



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA QUÍMICA, BIOTECNOLOGÍA Y
MATERIALES

**ESTRATEGIA DE CUANTIFICACIÓN DEL RANGO DE ESTABILIDAD DE
PARÁMETROS EN PROBLEMAS MINLP PARA EL DISEÑO DE REDES DE
AGUA**

TESIS PARA OPTAR AL GRADO DE DOCTOR EN CIENCIAS DE LA INGENIERÍA,
MENCION INGENIERÍA QUÍMICA Y BIOTECNOLOGÍA

FERNANDO ABEL ARENAS ARAYA

PROFESOR GUÍA:
FELIPE DÍAZ ALVARADO
PROFESOR CO-GUÍA:
JUAN ASENJO DE LEUZE DE LANCIZOLLE

MIEMBROS DE LA COMISIÓN:
ZIOMARA GERDTZEN HAKIM
LAUREANO JIMÉNEZ ESTELLER
MARÍA ELENA LIENQUEO CONTRERAS

Este trabajo ha sido financiado por: ANID-PFCHA/Doctorado Nacional/2018-21181490

SANTIAGO DE CHILE

2022

RESUMEN DE LA TESIS PARA OPTAR AL GRADO DE
DOCTOR EN CIENCIAS DE LA INGENIERÍA, MENCIÓN
INGENIERÍA QUÍMICA Y BIOTECNOLOGÍA
POR: FERNANDO ABEL ARENAS ARAYA
FECHA: 2022
PROF. GUÍA: FELIPE DÍAZ ALVARADO

ESTRATEGIA DE CUANTIFICACIÓN DEL RANGO DE ESTABILIDAD DE PARÁMETROS EN PROBLEMAS MINLP PARA EL DISEÑO DE REDES DE AGUA

Un problema de optimización de redes de agua puede ser formulado como un problema de programación no lineal mixta (MINLP), es decir, incluyendo variables continuas y enteras. Particularmente, el presente trabajo se enfoca en las variables binarias por la relación que tienen con las decisiones estratégicas de instalación de la red, configurando los costos de inversión y condicionando las decisiones operacionales. La resolución de este tipo de problemas requiere de la asignación de valores en parámetros que terminan configurando la solución mediante restricciones. Al asumir un valor de un parámetro no se tiene completa certeza sobre el valor escogido, especialmente aquellos sostenidos a partir de supuestos. Para analizar el efecto de asignar valores que no son precisos se plantea la siguiente pregunta: ¿cuánto es posible variar un parámetro manteniendo la solución óptima?

El efecto de variaciones en los parámetros se ha evaluado mayormente mediante herramientas de análisis post-optimal en problemas de programación lineal, por ejemplo, análisis de sensibilidad. En el caso de problemas MINLP, no se han reportado metodologías al respecto. A raíz de esta carencia, la novedad de este trabajo es el planteamiento de un algoritmo que responda eficientemente a la pregunta mencionada para problemas MINLP. El caso de estudio corresponde a las redes de agua, analizando el rango de validez de los parámetros para que no cambien las decisiones de instalación, denominado como rango de estabilidad. Se incluye como segunda novedad un índice de estabilidad, que permite jerarquizar los parámetros que deben ser analizados con mayor profundidad para asegurar que el óptimo obtenido en la etapa de diseño se mantenga.

La creación del algoritmo propuesto se basó en uno referencial generado para casos MILP. Este presenta un funcionamiento iterativo y se complementó con pasos que permiten: (i) acotar el rango de búsqueda; (ii) estimar un mejor valor inicial; (iii) reducir el tiempo de ejecución; (iv) determinar un índice de estabilidad. Esto permitió aplicar el algoritmo propuesto y comparar su desempeño con el algoritmo de referencia en tres casos MINLP de pequeña escala (entre 3 y 247 ecuaciones), determinando rangos de estabilidad más precisos y una reducción de tiempo de 35,0 %, 37,3 % y 96,6 % en los ejemplos 1, 2 y 3, respectivamente.

Finalmente, el algoritmo propuesto es validado mediante su aplicación en un problema de escala de ciudad, correspondiente a la red de agua de Santiago de Chile, obteniéndose que todos los valores evaluados al azar presentan el comportamiento esperado al interior y exterior del rango de estabilidad. Adicionalmente, el índice de estabilidad propuesto permitió reconocer que el parámetro crítico en la red analizada es el costo de electricidad. Este procedimiento y el índice pueden ser aplicados en otros contextos cuyas decisiones de mayor interés sean representadas mediante variables binarias.

*“En cualquier tarea se puede ganar o perder,
lo importante es la nobleza de los recursos utilizados.”*

Marcelo Bielsa

Agradecimientos

Quisiera comenzar agradeciendo a mi familia por el apoyo en este camino. Desde el cielo, de alguna u otra forma, mis padres siempre me han estado acompañando. Espero que se sientan orgullosos por el término de esta etapa y por los nuevos pasos que estoy dando. También mis hermanos, Karen, Claudia y Lalo, han sido un pilar fundamental apoyándome en mi día a día, para poder continuar haciendo lo que me gusta y apasiona.

El doctorado ha tenido diferentes etapas, con vivencias en la Universidad y con un período atípico de pandemia incluida, y en todo este tiempo mis amigos me ayudaron a que pudiera llevarlo adelante. En los primeros años las jornadas nómades de trabajo con la Mari me ayudaron mucho para buscar nuevas ideas y estar avanzando siempre. Las conversaciones en la oficina con Felipe, Dani, Javi, Jenny y tantas y tantos más me ayudaron a que fuera mucho más llevadero el trabajo, tanto con su apoyo intelectual como conversaciones para desconectarse y salir de una rutina que incluyó muchos fines de semana (uf!). Cómo no agradecerle también a la Catita por esos trayectos en micro al terminar el día y su paciencia y apoyo en los momentos más complejos, escuchando cada una de mis exageradas –para que fueran más divertidas– historias.

Ya en la época full tesis y encierro por la pandemia, las chelas, juegos de mesa y animé con el Seba fueron fundamentales para desconectarse de las largas jornadas de códigos y cosas ñoñas de la tesis. Las frecuentes videollamadas con la Andi me motivaron –y obligaron jaja– a trabajar en la tesis y que no quedara de lado. Agradezco también los panoramas con la Ceci, Nati, Pato y tantas personas más que hicieron que pudiera salir de la monotonía que involucraba el doctorado.

No todo en la U fue la tesis, pienso que estar cada vez más inmerso en Redes Beauchef me ayudó a ir aprendiendo y desarrollando nuevas habilidades, además de disfrutar de la compañía de las grandes personas que conformaban el equipo, especialmente de nuestro team tequila con la Cami y Magda. Vivir la experiencia de ser profesor y aprender de Willy también es algo que aprecio bastante al ir cerrando esta etapa. Muchas personas más fueron parte de este camino, que no alcanzo a individualizarlas acá pero que les estoy muy agradecido y, a mi manera, se los debo haber expresado durante estos años.

Mi etapa de seguir aprendiendo en la Universidad continuará, aunque desde otro rol – spoiler– así que espero seguir cerca de varias personas de mi día a día habitual en pregrado y postgrado, por lo que para muchas y muchos este cierre es un “continuará...”, y para las y los demás, nos veremos en alguno que otro panorama tranquilo que salga. Gracias a todas y todos por estar siempre.

Tabla de Contenido

1. Introducción	1
1.1. Antecedentes generales	2
1.1.1. Supuestos en Ingeniería	2
1.1.2. Motivación	2
1.1.3. Hipótesis	4
1.2. Objetivos y alcances	5
1.2.1. Objetivo general	5
1.2.2. Objetivos específicos	5
1.2.3. Alcances	5
1.3. Estado del arte	6
1.3.1. Análisis de sensibilidad	6
1.3.2. Métricas aplicadas para cuantificar el efecto de supuestos	7
1.3.3. Optimización	9
1.3.3.1. Optimización en Ingeniería de Procesos	9
1.3.3.2. Supuestos en Optimización	10
1.3.3.3. Análisis de sensibilidad en optimización	10
2. Caso de estudio: Redes de agua	14
2.1. Introducción	14
2.2. Optimización de redes de distribución de agua	16
3. Metodología	18
3.1. Algoritmo referencial	18
3.2. Determinación del rango de estabilidad	20
3.3. Cálculo del índice de estabilidad de parámetros	20
3.4. Recursos computacionales e implementación	21
3.5. Validación del algoritmo	22
4. Diagnóstico del algoritmo referencial	23
4.1. Ejemplo ilustrativo 1	24
4.1.1. Rango de estabilidad	25
4.1.2. Efecto del cambio de parámetros en las restricciones	27
4.1.3. Reflexiones parciales	31
4.2. Ejemplo ilustrativo 2	32
4.2.1. Planteamiento	32
4.2.1.1. Superestructura	32
4.2.1.2. Conjuntos	34
4.2.1.3. Parámetros	34

4.2.1.4.	Variables	34
4.2.1.5.	Restricciones	34
4.2.1.6.	Funciones objetivo	35
4.2.2.	Resultados del modelo de optimización	35
4.2.3.	Rango de estabilidad	36
4.2.4.	Configuraciones óptimas de la red en los extremos del rango de estabilidad	37
4.2.5.	Reflexiones parciales	39
4.3.	Ejemplo ilustrativo 3	40
4.3.1.	Planteamiento	40
4.3.1.1.	Superestructura	40
4.3.1.2.	Conjuntos	41
4.3.1.3.	Parámetros	41
4.3.1.4.	Variables	41
4.3.1.5.	Restricciones	42
4.3.1.6.	Función objetivo	42
4.3.1.7.	Ecuaciones adicionales	42
4.3.2.	Resultados del modelo de optimización	43
4.3.3.	Rango de estabilidad	43
4.3.4.	Configuraciones óptimas de la red en los extremos del rango de estabilidad	46
4.3.5.	Reflexiones parciales	47
4.4.	Reflexiones de los resultados preliminares	48
5.	Creación del nuevo algoritmo	50
5.1.	Características del nuevo algoritmo	50
5.2.	Descripción de las mejoras propuestas	53
5.2.1.	Iteraciones	53
5.2.1.1.	Funcionamiento iterativo	53
5.2.1.2.	Variables de interés	54
5.2.1.3.	Información previa de los parámetros	54
5.2.1.4.	Tiempo límite por iteración	54
5.2.2.	Adivinanza inicial	56
5.2.2.1.	Optimización de parámetros	56
5.2.3.	Finalización	59
5.2.3.1.	Índice de estabilidad	59
5.3.	Reflexiones de las ideas propuestas	60
5.4.	Algoritmo propuesto	61
6.	Comparación de resultados del algoritmo propuesto y el referencial	63
6.1.	Optimización de parámetros	63
6.2.	Determinación del rango de estabilidad	64
6.3.	Índice de estabilidad	66
6.4.	Discusiones	68
7.	Validación del algoritmo propuesto	69
7.1.	Ejemplo ilustrativo 4: Red de agua de Santiago	69
7.1.1.	Planteamiento	70
7.1.1.1.	Superestructura	70
7.1.1.2.	Parámetros	72

7.1.1.3. Variables	72
7.1.1.4. Restricciones	73
7.1.1.5. Funciones objetivo	74
7.1.2. Resultados del modelo de optimización	74
7.1.3. Rango de estabilidad	75
7.1.4. Índice de estabilidad	76
7.1.5. Reflexiones parciales	76
7.2. Validación	77
7.3. Reflexiones del algoritmo y la validación	80
8. Conclusión	83
Glosario	86
Bibliografía	87
Anexos	96
Anexo A. Situación hídrica de Chile frente al Cambio Climático	96
Anexo B. Antecedentes generales sobre optimización	99
B.1. Optimización matemática	99
B.2. Indicadores de sustentabilidad	101
B.3. Indicadores de sustentabilidad del agua	101
Anexo C. Ejemplo ilustrativo 1	103
C.1. Formulación	103
C.2. Solución óptima	103
C.3. Rango de estabilidad	104
C.3.1. Parámetro a	104
C.3.2. Parámetro b	106
C.4. Efecto del tiempo de resolución	108
C.4.1. Parámetro a	108
C.4.2. Parámetro b	109
C.5. Análisis de inecuaciones	110
C.6. Optimización de parámetros	113
C.7. Rango de estabilidad (algoritmo propuesto)	114
C.7.1. Parámetro a	114
C.7.2. Parámetro b	116
C.8. Índice de estabilidad	118
Anexo D. Ejemplo ilustrativo 2	119
D.1. Formulación	119
D.1.1. Conjuntos	119
D.1.2. Parámetros	119
D.1.3. Variables	120
D.1.4. Restricciones	120
D.1.5. Funciones objetivo	122
D.1.6. Supuestos del modelo	122

D.2.	Solución óptima	124
D.3.	Rango de estabilidad	125
D.3.1.	Parámetro D_{P_1}	125
D.3.2.	Parámetro D_{P_2}	128
D.3.3.	Parámetro $Cost_{T_1}$	131
D.3.4.	Parámetro $Cost_{T_2}$	134
D.4.	Efecto del tiempo de resolución	137
D.4.1.	Parámetro D_{P_1}	137
D.4.2.	Parámetro D_{P_2}	138
D.4.3.	Parámetro $Cost_{T_1}$	139
D.4.4.	Parámetro $Cost_{T_2}$	140
D.5.	Análisis de inecuaciones	141
D.6.	Optimización de parámetros	145
D.7.	Rango de estabilidad (algoritmo propuesto)	147
D.7.1.	Parámetro D_{P_1}	147
D.7.2.	Parámetro D_{P_2}	149
D.7.3.	Parámetro $Cost_{T_1}$	151
D.7.4.	Parámetro $Cost_{T_2}$	153
D.8.	Índice de estabilidad	155
Anexo E. Ejemplo ilustrativo 3		156
E.1.	Formulación	156
E.1.1.	Conjuntos	156
E.1.2.	Parámetros	156
E.1.3.	Variables	159
E.1.4.	Restricciones	160
E.1.5.	Función objetivo	164
E.1.6.	Ecuaciones adicionales	164
E.2.	Solución óptima	167
E.3.	Rango de estabilidad	170
E.3.1.	Parámetro $FPU_{P_1}^{in}$	170
E.3.2.	Parámetro $FPU_{P_2}^{in}$	172
E.3.3.	Parámetro r_{T_3}	175
E.3.4.	Parámetro IC_{T_1}	177
E.3.5.	Parámetro IC_{T_2}	180
E.3.6.	Parámetro IC_{T_3}	182
E.4.	Efecto del tiempo de resolución	184
E.4.1.	Parámetro $FPU_{P_1}^{in}$	184
E.4.2.	Parámetro $FPU_{P_2}^{in}$	187
E.4.3.	Parámetro r_{T_3}	190
E.4.4.	Parámetro IC_{T_1}	193
E.4.5.	Parámetro IC_{T_2}	196
E.4.6.	Parámetro IC_{T_3}	199
E.5.	Análisis de inecuaciones	202
E.6.	Optimización de parámetros	203
E.7.	Rango de estabilidad (algoritmo propuesto)	205
E.7.1.	Parámetro $FPU_{P_1}^{in}$	205

E.7.2.	Parámetro $FPU_{P_2}^{in}$	208
E.7.3.	Parámetro r_{T_3}	210
E.7.4.	Parámetro IC_{T_1}	212
E.7.5.	Parámetro IC_{T_2}	214
E.7.6.	Parámetro IC_{T_3}	216
E.8.	Índice de estabilidad	218
Anexo F. Ejemplo ilustrativo 4		219
F.1.	Formulación	219
F.1.1.	Conjuntos	219
F.1.1.1.	Subconjuntos	220
F.1.1.1.1.	Sub-subconjuntos	220
F.1.1.1.2.	Sub-sub-subconjuntos	221
F.1.1.2.	Otros tipos de conjuntos no mencionados previamente	222
F.1.1.3.	Sub-índices generales	222
F.1.1.4.	Sub-índices de plantas de tratamiento	223
F.1.2.	Parámetros	224
F.1.3.	Variables	225
F.1.4.	Restricciones	225
F.1.5.	Funciones objetivo	231
F.2.	Análisis de la formulación	232
F.2.1.	Características del problema	232
F.2.2.	Simplificaciones para la implementación del algoritmo	233
F.2.3.	Resumen del nuevo problema	234
F.2.4.	Parámetros de interés	234
F.3.	Solución óptima	235
F.4.	Optimización de parámetros	236
F.5.	Rango de estabilidad (algoritmo propuesto)	237
F.5.1.	Parámetro $DM_{\{RC,NW\}}$	237
F.5.2.	Parámetro $DM_{\{AC,SE\}}$	239
F.5.3.	Parámetro $DM_{\{IC,SE\}}$	241
F.5.4.	Parámetro E	243
F.5.5.	Parámetro M	245
F.6.	Índice de estabilidad	246
F.7.	Validación del algoritmo	247
F.7.1.	Parámetro $DM_{\{RC,NW\}}$	247
F.7.2.	Parámetro $DM_{\{AC,SE\}}$	248
F.7.3.	Parámetro $DM_{\{IC,SE\}}$	249
F.7.4.	Parámetro E	250
F.7.5.	Parámetro M	251
Anexo G. Publicación		252

Índice de Tablas

1.1.	Ejemplos de supuestos empleados en problemas de Ingeniería.	3
1.2.	Ejemplos de supuestos utilizados en problemas de optimización en Ingeniería. .	11
2.1.	Ejemplos de supuestos empleados en problemas de optimización de redes de agua.	17
4.1.	Características de los ejemplos utilizados para evaluar el desempeño del algoritmo referencial.	23
4.2.	Determinación del límite inferior del rango de estabilidad del parámetro a . . .	25
4.3.	Rango de estabilidad de los parámetros del ejemplo ilustrativo 2.	36
4.4.	Rango de estabilidad de los parámetros del ejemplo ilustrativo 3.	44
5.1.	Principales mejoras incorporadas en el algoritmo propuesto.	51
5.2.	Rango de estabilidad de los parámetros del ejemplo ilustrativo 2.	59
5.3.	Índice de estabilidad de los parámetros del ejemplo ilustrativo 2.	60
6.1.	Factibilidad de la asignación de la solución del problema reformulado como valor inicial para determinar el límite inferior o superior del rango de estabilidad en el algoritmo propuesto.	64
6.2.	Comparación de los resultados y tiempo utilizado por el algoritmo referencial y el propuesto.	65
6.3.	Índice de estabilidad de los parámetros de los ejemplos 1, 2 y 3.	67
7.1.	Número de nodos por tipo en la red de agua del ejemplo ilustrativo 4.	71
7.2.	Valores nominales de los parámetros de interés del ejemplo ilustrativo 4.	73
7.3.	Factibilidad de la asignación de la solución de los subproblemas (Min. y Max.) del ejemplo 4 como valor inicial en el algoritmo propuesto.	75
7.4.	Rango de estabilidad de los parámetros del ejemplo ilustrativo 4.	76
7.5.	Índice de estabilidad de los parámetros del ejemplo ilustrativo 4.	76
8.1.	Características y reducción del tiempo de implementación de los ejemplos utilizados para evaluar el desempeño de los algoritmos.	83
B.1.	Ejemplos de supuestos empleados en la elección de indicadores para problemas de Ingeniería.	102
C.1.	Determinación del límite inferior del rango de estabilidad del parámetro a . . .	104
C.2.	Determinación del límite superior del rango de estabilidad del parámetro a . .	105
C.3.	Determinación del límite inferior del rango de estabilidad del parámetro b . . .	106
C.4.	Determinación del límite superior del rango de estabilidad del parámetro b . .	107
C.5.	Solución óptima y restricciones del ejemplo ilustrativo 1 frente a variaciones en el parámetro a	110
C.6.	Solución óptima y restricciones del ejemplo ilustrativo 1 frente a variaciones en el parámetro b	112
C.7.	Parámetros optimizados del ejemplo ilustrativo 1.	113
C.8.	Determinación del límite inferior del rango de estabilidad del parámetro a mediante el algoritmo propuesto.	114

C.9.	Determinación del límite superior del rango de estabilidad del parámetro a mediante el algoritmo propuesto	115
C.10.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de a	115
C.11.	Determinación del límite inferior del rango de estabilidad del parámetro b mediante el algoritmo propuesto	116
C.12.	Determinación del límite superior del rango de estabilidad del parámetro b mediante el algoritmo propuesto	116
C.13.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de b	117
C.14.	Índice de estabilidad inferior y superior de los parámetros del ejemplo 1.	118
D.1.	Valores nominales de los parámetros del ejemplo ilustrativo 2.	120
D.2.	Supuestos empleados en el ejemplo ilustrativo 2.	123
D.3.	Resultados del ejemplo 2, empleando los valores nominales de los parámetros.	124
D.4.	Determinación del límite inferior del rango de estabilidad del parámetro D_{P_1}	125
D.5.	Valores de las variables en el límite inferior de D_{P_1}	126
D.6.	Determinación del límite superior del rango de estabilidad del parámetro D_{P_1}	126
D.7.	Valores de las variables en el límite superior de D_{P_1}	127
D.8.	Determinación del límite inferior del rango de estabilidad del parámetro D_{P_2}	128
D.9.	Valores de las variables en el límite inferior de D_{P_2}	128
D.10.	Determinación del límite superior del rango de estabilidad del parámetro D_{P_2}	129
D.11.	Valores de las variables en el límite superior de D_{P_2}	130
D.12.	Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_1}$	131
D.13.	Valores de las variables en el límite inferior de $Cost_{T_1}$	132
D.14.	Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_1}$	132
D.15.	Valores de las variables en el límite superior de $Cost_{T_1}$	133
D.16.	Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_2}$	134
D.17.	Valores de las variables en el límite inferior de $Cost_{T_2}$	135
D.18.	Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_2}$	135
D.19.	Valores de las variables en el límite superior de $Cost_{T_2}$	136
D.20.	Parámetros optimizados del ejemplo ilustrativo 2.	145
D.21.	Determinación del límite inferior del rango de estabilidad del parámetro D_{P_1} mediante el algoritmo propuesto.	147
D.22.	Determinación del límite superior del rango de estabilidad del parámetro D_{P_1} mediante el algoritmo propuesto.	148
D.23.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de D_{P_1}	148
D.24.	Determinación del límite inferior del rango de estabilidad del parámetro D_{P_2} mediante el algoritmo propuesto.	149
D.25.	Determinación del límite superior del rango de estabilidad del parámetro D_{P_2} mediante el algoritmo propuesto.	149
D.26.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de D_{P_2}	150
D.27.	Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_1}$ mediante el algoritmo propuesto.	151
D.28.	Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_1}$ mediante el algoritmo propuesto.	151

D.29.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $Cost_{T_1}$	152
D.30.	Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_2}$ mediante el algoritmo propuesto.	153
D.31.	Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_2}$ mediante el algoritmo propuesto.	154
D.32.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $Cost_{T_2}$	154
D.33.	Índice de estabilidad inferior y superior de los parámetros del ejemplo 2.	155
E.1.	Valores nominales de los parámetros operacionales del ejemplo ilustrativo 3.	158
E.2.	Valores nominales de parámetros económicos y otros del ejemplo ilustrativo 3.	158
E.3.	Flujos óptimos del ejemplo 3, empleando los valores nominales de los parámetros.	167
E.4.	Concentraciones óptimas del ejemplo ilustrativo 3, empleando los valores nominales de los parámetros.	168
E.5.	Instalaciones óptimas del ejemplo 3, empleando los valores nominales de los parámetros.	169
E.6.	Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$	170
E.7.	Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$	171
E.8.	Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$	172
E.9.	Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$	173
E.10.	Determinación del límite inferior del rango de estabilidad del parámetro r_{T_3}	175
E.11.	Determinación del límite superior del rango de estabilidad del parámetro r_{T_3}	176
E.12.	Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_1}	177
E.13.	Determinación del límite superior del rango de estabilidad del parámetro IC_{T_1}	178
E.14.	Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_2}	180
E.15.	Determinación del límite superior del rango de estabilidad del parámetro IC_{T_2}	181
E.16.	Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_3}	182
E.17.	Determinación del límite superior del rango de estabilidad del parámetro IC_{T_3}	183
E.18.	Parámetros optimizados del ejemplo ilustrativo 3.	203
E.19.	Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$ mediante el algoritmo propuesto.	205
E.20.	Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$ mediante el algoritmo propuesto.	206
E.21.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $FPU_{P_1}^{in}$	207
E.22.	Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$ mediante el algoritmo propuesto.	208
E.23.	Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$ mediante el algoritmo propuesto.	209
E.24.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $FPU_{P_2}^{in}$	209
E.25.	Determinación del límite inferior del rango de estabilidad del parámetro r_{T_3} mediante el algoritmo propuesto.	210
E.26.	Determinación del límite superior del rango de estabilidad del parámetro r_{T_3} mediante el algoritmo propuesto.	211
E.27.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de r_{T_3}	211

E.28.	Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_1} mediante el algoritmo propuesto.	212
E.29.	Determinación del límite superior del rango de estabilidad del parámetro IC_{T_1} mediante el algoritmo propuesto.	213
E.30.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de IC_{T_1}	213
E.31.	Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_2} mediante el algoritmo propuesto.	214
E.32.	Determinación del límite superior del rango de estabilidad del parámetro IC_{T_2} mediante el algoritmo propuesto.	214
E.33.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de IC_{T_2}	215
E.34.	Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_3} mediante el algoritmo propuesto.	216
E.35.	Determinación del límite superior del rango de estabilidad del parámetro IC_{T_3} mediante el algoritmo propuesto.	216
E.36.	Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de IC_{T_3}	217
E.37.	Índice de estabilidad inferior y superior de los parámetros del ejemplo 3.	218
F.1.	Consumo de agua por tipo y distrito en el ejemplo ilustrativo 4.	234
F.2.	Valores nominales de los parámetros de interés del ejemplo ilustrativo 4.	234
F.3.	Instalaciones óptimas del ejemplo ilustrativo 4 con los valores nominales de los parámetros.	235
F.4.	Parámetros optimizados del ejemplo ilustrativo 4.	236
F.5.	Determinación del límite inferior del rango de estabilidad del parámetro $DM_{\{RC,NW\}}$ mediante el algoritmo propuesto.	237
F.6.	Determinación del límite superior del rango de estabilidad del parámetro $DM_{\{RC,NW\}}$ mediante el algoritmo propuesto.	238
F.7.	Determinación del límite inferior del rango de estabilidad del parámetro $DM_{\{AC,SE\}}$ mediante el algoritmo propuesto.	239
F.8.	Determinación del límite superior del rango de estabilidad del parámetro $DM_{\{AC,SE\}}$ mediante el algoritmo propuesto.	240
F.9.	Determinación del límite inferior del rango de estabilidad del parámetro $DM_{\{IC,SE\}}$ mediante el algoritmo propuesto.	241
F.10.	Determinación del límite superior del rango de estabilidad del parámetro $DM_{\{IC,SE\}}$ mediante el algoritmo propuesto.	242
F.11.	Determinación del límite inferior del rango de estabilidad del parámetro E mediante el algoritmo propuesto.	243
F.12.	Determinación del límite superior del rango de estabilidad del parámetro E mediante el algoritmo propuesto.	244
F.13.	Determinación del límite inferior del rango de estabilidad del parámetro M mediante el algoritmo propuesto.	245
F.14.	Determinación del límite superior del rango de estabilidad del parámetro M mediante el algoritmo propuesto.	245
F.15.	Índice de estabilidad inferior y superior de los parámetros del ejemplo 4.	246

Índice de Ilustraciones

1.1.	Comparación entre el análisis OAT (a) y los resultados del SA correspondiente (b), en contraste con un análisis GSA (c) y sus índices de sensibilidad (d). Imagen obtenida de [37].	6
2.1.	Representación de una red de distribución y tratamiento de agua. Elaboración propia con información de [67].	15
3.1.	Algoritmo para obtener el rango de estabilidad de parámetros en problemas MILP [31].	19
3.2.	Comportamiento esperado de las variables binarias en relación al rango de estabilidad determinado. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	22
4.1.	Variación de la variable binaria y frente a cambios en el parámetro a . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	26
4.2.	Restricciones del ejemplo 1 frente a cambios del parámetro b	28
4.3.	Restricciones del ejemplo 1 frente a cambios del parámetro a	30
4.4.	Superestructura del ejemplo ilustrativo 2.	33
4.5.	Red óptima del ejemplo ilustrativo 2.	35
4.6.	Red óptima al disminuir D_{P_1} en el ejemplo ilustrativo 2.	37
4.7.	Red óptima al aumentar D_{P_2} en el ejemplo ilustrativo 2.	38
4.8.	Red óptima al disminuir D_{P_2} en el ejemplo ilustrativo 2.	38
4.9.	Superestructura del ejemplo ilustrativo 3.	41
4.10.	Red óptima del ejemplo ilustrativo 3.	43
5.1.	Algoritmo propuesto para obtener el rango de estabilidad de parámetros en problemas MINLP. Etapas verdes: Pasos mantenidos del algoritmo de Yi y Lu [31]. Etapas naranjas: Modificadas del algoritmo de Yi y Lu [31]. Etapas rojas: Nuevos pasos propuestos para la versión MINLP. Elaboración propia.	52
5.2.	Restricciones del ejemplo 1 reformulado frente a cambios en los parámetros.	58
5.3.	Algoritmo propuesto para obtener el rango de estabilidad de parámetros en problemas MINLP, aplicado al parámetro r_{T_3} . Etapas verdes: Pasos mantenidos del algoritmo de Yi y Lu [31]. Etapas naranjas: Modificadas del algoritmo de Yi y Lu [31]. Etapas rojas: Nuevos pasos propuestos para la versión MINLP. Elaboración propia.	62
7.1.	División de Santiago para el ejemplo ilustrativo 4. Donde NW: Noroeste, NE: Noreste, SW: Suroeste y SE: Sureste. Imagen obtenida de [75].	70

7.2.	Superestructura del ejemplo ilustrativo 4. El color de las líneas representa la calidad del agua (verde: potable, azul: dulce, naranja: riego, roja: descarga y gris: efluentes industriales). Pese a no ser representadas, se permiten las conexiones de IC con IWTP de diferentes distritos. La existencia de las líneas discontinuas dependerá de la calidad de salida del agua de las WWTP(n). Elaboración propia con información de [75].	72
7.3.	Red óptima del ejemplo ilustrativo 4. El color de las líneas representa la calidad del agua (verde: potable, azul: dulce, roja: descarga y gris: efluentes industriales). Las líneas verdes discontinuas señalan que las plantas WWTP(n) presentan una calidad de salida de agua potable.	75
7.4.	Rango de valores aleatorios generados para la validación del algoritmo. Se ejemplifica con un parámetro a , cuyo valor nominal es a^{nom} , el límite inferior es a^{inf} y el superior es a^{sup} . La longitud del rango de estabilidad (λ_a) se compone de la diferencia entre el valor nominal y el límite inferior (λ_a^{inf}) y la distancia entre el valor nominal y el límite superior (λ_a^{sup}).	77
7.5.	Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{RC,NW\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.	78
7.6.	Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{AC,SE\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.	79
A.1.	Distribución del consumo de agua superficial y subterránea en Chile. Imagen obtenida de [97].	96
C.1.	Variación de la variable binaria y frente a cambios en el parámetro b . Línea roja: límite inferior calculado.	107
C.2.	Número de iteraciones de a resueltas según el tiempo de ejecución.	108
C.3.	Número de iteraciones de b resueltas según el tiempo de ejecución.	109
C.4.	Variación del valor de las restricciones frente a cambios en el parámetro a . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	111
C.5.	Variación del valor de las restricciones frente a cambios en el parámetro b . Línea roja: límite inferior calculado.	112
D.1.	Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro D_{P_1} . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	127
D.2.	Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro D_{P_2} . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	130
D.3.	Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro $Cost_{T_1}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	133
D.4.	Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro $Cost_{T_2}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	136
D.5.	Número de iteraciones de D_{P_1} resueltas según el tiempo de ejecución.	137
D.6.	Número de iteraciones de D_{P_2} resueltas según el tiempo de ejecución.	138
D.7.	Número de iteraciones de $Cost_{T_1}$ resueltas según el tiempo de ejecución.	139
D.8.	Número de iteraciones de $Cost_{T_2}$ resueltas según el tiempo de ejecución.	140

D.9.	Variación del valor de las restricciones frente a cambios en el parámetro D_{P_1} . Línea vertical roja: límite inferior calculado. Línea vertical verde: límite superior calculado.	142
D.10.	Variación del valor de las restricciones frente a cambios en el parámetro D_{P_2} . Línea vertical roja: límite inferior calculado. Línea vertical verde: límite superior calculado.	142
D.11.	Variación del valor de las restricciones frente a cambios en el parámetro $Cost_{T_1}$. Línea vertical roja: límite inferior calculado. Línea vertical verde: límite superior calculado.	143
D.12.	Variación del valor de las restricciones frente a cambios en el parámetro $Cost_{T_2}$. Línea vertical roja: límite inferior calculado.	144
E.1.	Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_1}^{in}$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.	171
E.2.	Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.	174
E.3.	Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} . Línea roja: límite inferior calculado. Línea verde: límite superior calculado. . .	176
E.4.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} . Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.	179
E.5.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_2} . Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.	181
E.6.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_3} . Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.	183
E.7.	Número de iteraciones de $FPU_{P_1}^{in}$ resueltas según el tiempo de ejecución. . . .	184
E.8.	Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_1}^{in}$ con $\tau = 5$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.	185
E.9.	Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_1}^{in}$ con $\tau = 1$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.	186
E.10.	Número de iteraciones de $FPU_{P_2}^{in}$ resueltas según el tiempo de ejecución. . . .	187
E.11.	Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$ con $\tau = 20$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.	188
E.12.	Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$ con $\tau = 5$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.	189
E.13.	Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$ con $\tau = 1$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.	189
E.14.	Número de iteraciones de r_{T_3} resueltas según el tiempo de ejecución.	190

E.15.	Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} con $\tau = 100$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	191
E.16.	Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} con $\tau = 5$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	192
E.17.	Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} con $\tau = 1$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.	192
E.18.	Número de iteraciones de IC_{T_1} resueltas según el tiempo de ejecución.	193
E.19.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} con $\tau = 20$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.	194
E.20.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} con $\tau = 5$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.	195
E.21.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} con $\tau = 1$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.	195
E.22.	Número de iteraciones de IC_{T_2} resueltas según el tiempo de ejecución.	196
E.23.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_2} con $\tau = 5$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.	197
E.24.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_2} con $\tau = 1$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.	198
E.25.	Número de iteraciones de IC_{T_3} resueltas según el tiempo de ejecución.	199
E.26.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_3} con $\tau = 5$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.	200
E.27.	Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_3} con $\tau = 1$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.	201
F.1.	Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{RC,NW\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.	247
F.2.	Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{AC,SE\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.	248
F.3.	Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{IC,SE\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.	249
F.4.	Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro E . Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.	250

F.5. Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro M . Línea roja: límite inferior calculado. Línea naranja discontinua: valor nominal. 251

Capítulo 1

Introducción

La toma de decisiones se encuentra inmersa en diversos contextos, abarcando la medicina [1], ingeniería [2], deporte [3], educación [4], entre otros. Respecto a la ingeniería, ya sea en el corto o largo plazo, y a niveles técnicos o estratégicos, las decisiones usualmente se realizan con información incompleta. La incertidumbre asociada puede ocasionar cambios costosos en etapas posteriores al diseño, como ocurre con las modificaciones en la operación [5].

En los casos en que no hay certeza sobre un comportamiento o valor numérico requerido para el diseño en ingeniería, se adoptan supuestos con distintos efectos e incertidumbre [6]. Las aplicaciones de supuestos en ingeniería han tenido como propósito simplificar modelos matemáticos [7] [8], calcular indicadores asociados al problema [9], representar fenómenos físicos [10], entre otras alternativas que facilitan la toma de decisiones.

Las simplificaciones usadas en problemas de optimización de ingeniería tienen diversos impactos, como por ejemplo en la reducción del número de parámetros requeridos en el modelo correspondiente [7]. Considerando que la información disponible puede ser incompleta, se podrían estar realizando estimaciones imprecisas sobre los parámetros utilizados en los problemas de optimización, cobrando una gran relevancia los análisis post-optimal [11]. Precisamente, el objetivo de este trabajo es estudiar el efecto de los supuestos en estos casos y cuantificar las condiciones en que pueden ser utilizados sin modificar la solución.

En este primer capítulo se presentan los principales antecedentes asociados a la toma de decisiones, supuestos, optimización y análisis post-optimal en el diseño de la Ingeniería de Procesos, junto con los objetivos, alcances y novedad de este trabajo. Los demás contenidos de este documento se estructuran de la siguiente manera: el Capítulo 2 introducirá el caso de estudio, correspondiente a las redes de agua; la metodología para la elaboración de la estrategia de cuantificación del rango de estabilidad será descrita en el Capítulo 3; posteriormente, en el Capítulo 4 se determinará el rango de estabilidad para diversos ejemplos y se reconocerán potenciales mejoras; esta información permitirá que en el Capítulo 5 se proponga y describa la estrategia correspondiente al objetivo central de esta tesis; posteriormente, se aplicará la estrategia propuesta y los resultados se compararán con los obtenidos mediante el algoritmo de referencia en el Capítulo 6; habiendo creado y comparado la nueva estrategia, en el Capítulo 7 se aplicará el algoritmo propuesto en el caso de estudio principal y se validará; finalmente, las principales conclusiones del trabajo realizado se encuentran en el Capítulo 8.

1.1. Antecedentes generales

Las ingenieras e ingenieros deben resolver problemas lidiando con la realidad, presentándose diversas dificultades y soluciones. Uno de sus principales objetivos es obtener la mejor solución posible con los recursos disponibles [12]. En términos generales, las decisiones en ingeniería se pueden clasificar según el tipo de problema, alternativas, objetivos, además de considerar la incertidumbre asociada a cada uno [13]. Esta incertidumbre genera la necesidad de recurrir a los supuestos, que son analizados a continuación:

1.1.1. Supuestos en Ingeniería

Mamun y Hansson [14] elaboraron un *review* sobre supuestos y sus desafíos en el desarrollo de software. Consideraron como supuesto a una “*sentencia que se asume como verdadera y que pierde validez si se determina que es falsa*”. Los supuestos en ingeniería son un apoyo para la resolución de problemas, permitiendo disminuir la brecha entre el objetivo a alcanzar y el conocimiento o información que se tiene [15]. Algunos ejemplos de supuestos implementados en ingeniería se resumen en la Tabla 1.1, destacando la importancia del contexto, ya que métodos y herramientas pueden variar su validez en diferentes escenarios [16]. Los supuestos de la tabla mencionada ilustran que su aplicación aleja de la realidad al modelo respectivo. Las aplicaciones que están basadas en supuestos que no son válidos es probable que no logren los resultados deseados [16].

La cuantificación del efecto de los supuestos no ha sido investigada extensamente con tal denominación. Una de las formas en que ha sido abarcada corresponde a los análisis de sensibilidad o análisis post-optimal, que permiten conocer la importancia de los parámetros en las soluciones de distintos problemas, mediante la resolución de estos en reiteradas ocasiones con variaciones en los valores de cada parámetro, siendo costoso computacionalmente según el tamaño del problema. Pese a que no se declara directamente en todos los casos, los parámetros pueden surgir de supuestos adoptados, por lo que este tipo de análisis se considerará entre los antecedentes relevantes para la cuantificación del efecto de suponer y asignar valores a parámetros en un problema de optimización. En la sección 1.3 se profundizará sobre análisis de sensibilidad y algunas métricas que se han aplicado en este tipo de diagnósticos.

1.1.2. Motivación

Al desarrollar proyectos de ingeniería se emplean supuestos que poseen distintos efectos y riesgos [23]. La toma de decisiones presente en los proyectos, que en diversos casos se realiza con información incompleta, puede ocasionar que los resultados no sean los esperados, teniendo como meta aproximarse a la realidad. Desde la perspectiva de la sustentabilidad, los desafíos en los proyectos de ingeniería sugieren replantear cómo los procesos y productos industriales son diseñados, construidos, operados y evaluados [24]. En ese sentido, uno de los principales retos en ingeniería corresponde a solucionar los efectos del Cambio Climático. Entre estos se destaca la escasez de agua a nivel mundial a raíz del aumento en la temperatura y variaciones en el nivel de precipitaciones en distintas regiones, proyectándose que estos cambios continúen [25] [26].

Para poder afrontar la escasez de agua, comida y bienes en general, el consumo y producción deben ser diseñados con un enfoque sustentable [27]. En particular, la seguridad

Tabla 1.1: Ejemplos de supuestos empleados en problemas de Ingeniería.

Supuesto (Área de estudio)	Explicación	Ref.
Consumo inmediato de una especie (Corrosión)	Se asume que la corrosión es un proceso limitado por la difusión de oxígeno, por lo que se considera que el oxígeno es consumido inmediatamente al permear en el material analizado, presentando una concentración superficial nula. Si se asume otra etapa limitante, se obtienen otros efectos.	[17]
Comportamiento newtoniano de la sangre (Hemodinámica)	La sangre es un fluido no newtoniano, cuyo flujo se ha estudiado mediante modelos de fluidos newtonianos en los casos en que el diámetro del vaso sanguíneo es considerado mucho mayor que el de las partículas sólidas en la sangre.	[18] [19]
Permeabilidad de la fractura de un pozo es constante (Fracking)	Para disminuir la complejidad del problema, se considera que la permeabilidad de la fractura entre el pozo y la superficie no varía. Sin embargo, la permeabilidad suele tener valores más altos en el fondo del pozo y va disminuyendo al alejarse.	[20]
Reducción del número de componentes (Modelación de fluidos)	En un embalse es posible encontrar diversas especies químicas, en diferentes fases y composiciones. Para simplificar las ecuaciones a emplear, se han agrupado según distintos criterios, permitiendo reducir el número de especies a analizar.	[21]
Confiabilidad de datos recopilados (Caracterización de una población)	Para estudiar diferencias entre grupos de personas, generalmente la caracterización se realiza mediante encuestas. Los datos son completados por quienes se está estudiando, por lo que la fiabilidad y validez podría no ser completa.	[22]

del suministro de agua es un aspecto clave, debido a la vulnerabilidad de la disponibilidad de agua ante los continuos cambios en los patrones climáticos [26] y a la necesidad de este recurso en la vida cotidiana y en distintos procesos productivos [28].

El consumo de agua a nivel mundial ha aumentado y se espera que la tendencia continúe [29]. Adicionalmente, en Chile se proyecta una posible situación de estrés hídrico extremadamente alto para el año 2040 [30]. Mayores detalles de la situación hídrica del país y posibles medidas para enfrentar la escasez de agua se detallan en el Anexo A.

Para distribuir eficientemente el agua es posible analizar la interacción entre los distintos actores involucrados. La forma en que se relacionan los consumidores y las fuentes se puede representar mediante redes de distribución de agua, que serán descritas en profundidad en el Capítulo 2, ya que corresponden al caso de estudio de este trabajo.

A partir de los antecedentes anteriores, la motivación de este trabajo es mejorar la

modelación de redes de distribución y tratamiento de agua, fortaleciendo el diagnóstico de los supuestos adoptados. Específicamente, para los parámetros cuyo valor es sostenido en supuestos: ¿cuál es el rango de validez del parámetro para que la respuesta de un problema siga siendo óptima? (rango de estabilidad [31]).

La solución de un problema, en particular en los modelos de optimización de redes de agua, se compone de diversas decisiones. Estas pueden ser jerarquizadas en tres niveles: estratégicas, tácticas y operacionales [32]. Entre las decisiones posibles se destaca la instalación de plantas de tratamiento, considerándose del tipo estratégico al ser una decisión que se toma para un largo período de tiempo y, luego de su implementación, es difícil de revertir. Estas se pueden plantear como variables discretas, representando la instalación o no de la planta a la que esté asociada y que usualmente corresponden a variables binarias (variables 0 - 1) [33]. Ante esto, resulta de gran interés el análisis del rango de estabilidad con el que las variables binarias mantienen su valor, por el cambio de estructura de la red que se produce si el valor de estas variables se modifica. A lo largo del presente trabajo se analizarán estas variables en casos de diversas características.

Para problemas a gran escala sería conveniente tener una estimación del rango de estabilidad con el menor número de ejecuciones posible. Esta meta se basa en el tiempo que tarda en ser obtenida la solución de un problema a gran escala y también en el número de escenarios que se deberían evaluar según cuántos parámetros se analicen. Junto con ello, se destaca que evidenciar el uso de supuestos y sus implicancias también forma parte de la motivación de este trabajo. Supuestos que se realicen sin justificación podrían presentar un alto riesgo [34], lo que podría afectar la solución de un problema en ingeniería. De esta forma, se espera contribuir con este enfoque a un desarrollo ingenieril responsable.

En términos generales, la pregunta usual a resolver en los problemas de análisis del efecto de supuestos o de valores escogidos para los parámetros es: ¿cuánto se modifica una solución frente a un cambio en algún parámetro? Para el presente trabajo se efectúa una pregunta similar pero con otra perspectiva: ¿cuánto es posible modificar un parámetro para que no cambie la solución óptima? A continuación, se presentan la hipótesis del presente trabajo, continuando con los objetivos y el estado del arte sobre análisis de supuestos y estabilidad en el contexto de la optimización.

1.1.3. Hipótesis

Se propone que es factible determinar el rango de estabilidad de parámetros provenientes de supuestos en el diseño de una red de agua planteada como un problema de optimización MINLP. Específicamente, se considera que es posible crear un algoritmo que permita determinar este rango eficientemente, es decir, estimando un punto inicial que disminuya las iteraciones necesarias. Se presume factible la comparación de los rangos de estabilidad de distintos parámetros para priorizar aquellos que necesitan un análisis detallado sobre los valores asumidos en estos. Con ello, se busca apoyar a la detección de parámetros críticos que puedan generar que la solución obtenida no sea efectivamente óptima.

1.2. Objetivos y alcances

1.2.1. Objetivo general

Crear un algoritmo de apoyo a la toma de decisiones en ingeniería, enfocado en la cuantificación de la validez de los supuestos de un problema de optimización, mediante la determinación eficiente del rango de estabilidad de los parámetros a los que afectan en problemas de programación no lineal y mixta de redes de agua.

1.2.2. Objetivos específicos

- Analizar diferentes problemas MINLP de redes de agua, reconociendo los supuestos empleados en las diversas formulaciones y factores que permitan predecir cómo se comportan las variables binarias frente a cambios en los parámetros.
- Crear un indicador que permita comparar los rangos de estabilidad de los parámetros analizados y así priorizar aquellos que deban ser estudiados en mayor profundidad.
- Proponer un algoritmo que permita determinar el rango de estabilidad de parámetros, incorporando una adivinanza inicial eficiente y el indicador de estabilidad.
- Aplicar y validar el algoritmo propuesto en un problema de escala de ciudad basado en la red de agua de Santiago de Chile.

1.2.3. Alcances

El objetivo del algoritmo a crear es la determinación del rango de validez de supuestos que tengan efecto sobre parámetros del problema de optimización MINLP, interpretado como el rango de estabilidad de los parámetros, es decir, cuánto es posible modificarlos sin que cambie la solución del problema. Particularmente, se considerará que se mantiene la solución si es que el valor de las variables binarias obtenidas dentro de la optimización del problema determinístico no cambia, debido a la relación entre este tipo de variables y las decisiones de instalación de una red de agua (decisiones a mediano y largo plazo que influyen en los costos de inversión).

No serán incluidas las modificaciones o efectos que generen los supuestos sobre otros elementos del problema de optimización (variables, restricciones o funciones objetivo), como podría ser la inclusión o exclusión de restricciones, por ejemplo. Tampoco se analizarán variaciones de múltiples parámetros a la vez, solamente cambios individuales, dado que se está creando una aproximación inicial al análisis de este tipo en problemas MINLP.

La estrategia será un complemento para la formulación de problemas de redes de agua en los que no se conoce cómo se comportan los parámetros. Precisamente, se busca obtener el rango de estabilidad de los parámetros para que se puedan priorizar, mediante el indicador a crear, aquellos que deben ser estudiados más en detalle y asegurar que se mantengan en el rango determinado, ya que en caso contrario la red obtenida como solución no sería óptima.

1.3. Estado del arte

Los principales temas considerados para el presente trabajo y que se destacan en esta sección, corresponden al análisis de sensibilidad, debido al estudio de los efectos de modificaciones de parámetros; cuantificación del efecto de supuestos, ya que se analizarán cambios en los parámetros condicionados por estos; y, finalmente, optimización. Este último tópico se presenta en relación a los supuestos y el análisis de sensibilidad.

1.3.1. Análisis de sensibilidad

En los últimos años ha existido un aumento en el número de publicaciones científicas en que se utiliza el término *análisis de sensibilidad* (SA¹) [35]. En la mayoría de estas se efectúa un análisis de factores *One-At-a-Time* (OAT), en el que se varían los parámetros de entrada, uno a la vez, para describir las variaciones que ocurren en las variables de salida. En estos casos se calculan o estiman derivadas parciales del modelo en un punto específico [36].

Este tipo de análisis posee dos desventajas principales: en primer lugar, OAT solo permite explorar una pequeña porción del espacio en que pueden variar los parámetros, evidenciado aún más al tener un mayor número de parámetros a analizar, siendo más conveniente en casos en que se utilice una referencia y se modifique solamente un parámetro a la vez (ver Figura 1.1.(a)); en segundo lugar, no permite detectar interacciones entre los parámetros analizados, considerando que la modificación de dos o más parámetros a la vez podría generar distintos efectos que no son reflejados mediante un análisis OAT [37] (ver Figura 1.1.(b)).

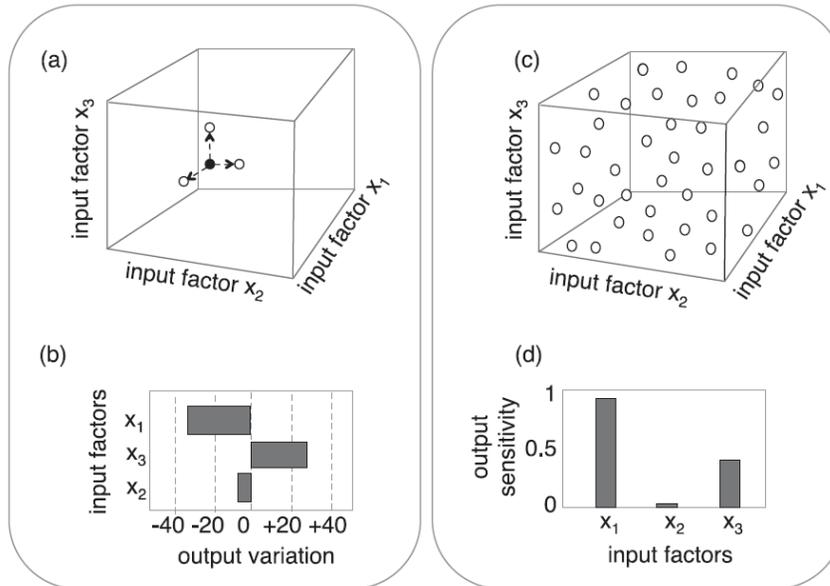


Figura 1.1: Comparación entre el análisis OAT (a) y los resultados del SA correspondiente (b), en contraste con un análisis GSA (c) y sus índices de sensibilidad (d). Imagen obtenida de [37].

Para lidiar con las desventajas anteriores, se ha planteado utilizar un análisis de sensibilidad global (GSA²), en el que todos los parámetros de entrada a estudiar se varían para

¹ Por su sigla en inglés, proveniente de *Sensitivity Analysis*.

² Por su sigla en inglés, proveniente de *Global Sensitivity Analysis*.

cubrir el espacio en que pueden desenvolverse (ver Figura 1.1.(c)). Los métodos cuantitativos de GSA usualmente generan un conjunto de índices de sensibilidad (ver Figura 1.1.(d)), que están asociados a las distintas combinaciones que se pueden generar [37]. GSA permite ordenar los parámetros basándose en su importancia o contribución en la incertidumbre de las variables, y en algunos casos cuantificar el impacto global que generan en estas [38]. Las definiciones y conceptos claves de un GSA se mencionan a continuación:

Si la relación entre un parámetro (a_i) y una variable (Y) es no lineal, pequeñas variaciones en el parámetro pueden provocar grandes cambios en la variable. Por otro lado, podría ocurrir que variaciones grandes en un parámetro (a_j) generen variaciones menores en la variable de salida (Y). En este caso, se considerará que Y es más sensible a a_i que a a_j . A partir de este tipo de comparaciones se puede lograr el objetivo de los análisis de sensibilidad global: obtener un ranking de importancia de los parámetros de entrada para el sistema. Los métodos que son comúnmente utilizados en GSA son los que están basados en la varianza, con los que se cuantifica la sensibilidad de la variable $Y = f(a_i)$ respecto a un parámetro a_i [39]. El índice de sensibilidad para el parámetro a_i definido para los métodos basados en la varianza es el siguiente:

$$S_i = \frac{Var_{a_i}[\mathbb{E}(Y|a_i)]}{Var(Y)} \quad (1.1)$$

Donde:

- a_i : Parámetro que influye en la variable de salida del problema. $i = 1, \dots, n$, siendo n el número total de parámetros considerados.
- Y : Variable del problema, siendo $Y = f(a_1, \dots, a_n)$.
- S_i : Índice de sensibilidad, que corresponde a la fracción de la variación de Y que se puede atribuir al parámetro a_i . Se obtiene comparando la varianza de la esperanza condicional $Var_{a_i}[\mathbb{E}(Y|a_i)]$ respecto a la varianza total $Var(Y)$.

El concepto de ranking de importancia será considerado para la priorización de parámetros según el rango de estabilidad de cada uno, correspondiente a uno de los objetivos del presente trabajo. Junto con el análisis de sensibilidad, es posible aplicar métricas como las siguientes:

1.3.2. Métricas aplicadas para cuantificar el efecto de supuestos

Error relativo

En el área de la Ingeniería Biomédica, Alastruey et al. [18] analizaron el efecto de supuestos adoptados para simular el movimiento de la sangre, teniendo como propósito comprender y minimizar el impacto que generan. Para ello, contrastan los datos clínicos adquiridos y los resultados obtenidos a través de los modelos que incorporan supuestos en sus parámetros, calculando el error en la estimación. Específicamente, estos valores se compararon a través del error relativo, calculado según las expresiones propuestas por Xiao et al. [40]:

$$\epsilon_{1,prom} = \frac{1}{N_t} \cdot \sum_{i=1}^{N_t} \left| \frac{x_i - x_i^*}{x_i^*} \right| \quad (1.2)$$

$$\epsilon_{2,prom} = \frac{1}{N_t} \cdot \sum_{i=1}^{N_t} \left| \frac{x_i - x_i^*}{\max_j(x_j^*)} \right| \quad (1.3)$$

$$\epsilon_{1,max} = \max_i \left| \frac{x_i - x_i^*}{x_i^*} \right| \quad (1.4)$$

$$\epsilon_{2,max} = \max_i \left| \frac{x_i - x_i^*}{\max_j(x_j^*)} \right| \quad (1.5)$$

Donde:

- N_t : Número total de puntos (datos) considerados para la comparación.
- x_i : Es el valor de la variable x en el punto i obtenida mediante el modelo que considera un set determinado de supuestos.
- x_i^* : Corresponde al valor de la variable x en el punto i obtenida a partir del modelo que considera supuestos diferentes a los de x_i .
- $\epsilon_{1,prom}$: Error relativo promedio.
- $\epsilon_{2,prom}$: Error relativo promedio, normalizado por el valor máximo de la variable con la que se está comparando. Este error se utiliza para evitar problemas numéricos en caso de que la variable en el denominador sea un número cercano a cero.
- $\epsilon_{1,max}$: Es la diferencia relativa máxima entre las dos variables comparadas.
- $\epsilon_{2,max}$: Con el mismo criterio que $\epsilon_{2,prom}$, se normaliza la diferencia relativa máxima.

Se destaca que la inclusión de la diferencia normalizada en la formulación del error relativo será una fuente de ideas para la elaboración del índice de estabilidad.

Intervalo de valores (range value)

Para el diseño de válvulas de compuertas submarinas, Liu et al. [41] analizaron la sensibilidad de la velocidad máxima del flujo que se transportaba y de la deformación máxima de la compuerta respecto a modificaciones en tres parámetros principales de diseño. Para establecer qué parámetro presenta una mayor influencia en cada variable, se calculan los valores máximos y mínimos generados, y así obtener la amplitud de cambios en las variables según la expresión de la ecuación 1.6. Específicamente, se establece que el parámetro que genere un mayor rango R_j es el que posee una mayor influencia en la variable de salida. Análogamente, el que posee un menor rango R_j es el que influye en menor medida.

$$R_j = \max_{i \in I} \{K_{ij}\} - \min_{i \in I} \{K_{ij}\} \quad (1.6)$$

Donde:

- K_{ij} : Valor promedio de la variable K obtenida mediante I modificaciones del parámetro j . En este caso, se analiza en cada iteración i todas las variaciones de los otros parámetros, las que son promediadas.
- R_j : Intervalo de valores generado mediante las modificaciones en el parámetro j .

El intervalo de valores es una de las métricas que podrían ser empleadas para evaluar cuánto se modifica una solución por efecto de un cambio en los parámetros. Esto permite evidenciar el contraste que se desea efectuar con la determinación del rango de estabilidad, ya que el nuevo enfoque está en cuánto es posible variar un parámetro sin que se modifique la solución. De todas formas, la noción de comparación entre amplitudes de rangos es una fuente creativa para la elaboración del índice de estabilidad.

Las métricas de los trabajos anteriores han permitido cuantificar el efecto de supuestos mediante comparaciones. Sin embargo, pueden requerir un gran número de datos o múltiples resoluciones del problema para generar los distintos escenarios. Contrastando estas opciones con el algoritmo que se propondrá, se destaca la diferencia de enfoques: los trabajos destacados se centran en cuánto afecta la modificación de los parámetros, mientras que el nuevo algoritmo se enfocará en cuánto es posible cambiar un parámetro sin que se vea afectada la solución óptima. De todas formas, los trabajos señalados son una fuente de inspiración para el análisis de parámetros y el índice de estabilidad a desarrollar.

En el ámbito de la optimización, los análisis *a posteriori*, como los explicados anteriormente, presentan una importante desventaja: en problemas a gran escala el tiempo de resolución puede ser elevado, por lo que necesitan alternativas que no requieran resolver el problema en numerosas ocasiones. Este desafío se ha abordado para determinados tipos de problemas de optimización, que serán explicados en la siguiente sección.

1.3.3. Optimización

Para comprender el efecto de los supuestos en los problemas de optimización y, específicamente, en la estimación de los parámetros involucrados, es necesario entender este tipo de problemas en el contexto de la Ingeniería de Procesos. El estado del arte de optimización descrito a continuación se enfoca en el estudio del efecto de parámetros en el diseño de procesos. Particularmente, se explicará el concepto de *rango de estabilidad*, bajo el que se centra este trabajo. Se presenta también una breve contextualización, mientras que los antecedentes generales sobre optimización se pueden consultar en el Anexo B.

1.3.3.1. Optimización en Ingeniería de Procesos

Los modelos de optimización se resuelven con la estrategia que mejor se adapte a las características de la función objetivo, restricciones y variables [42]. En particular, la naturaleza de los balances de masa en la Ingeniería de Procesos motiva el uso de, principalmente, estrategias que resuelven problemas no lineales [43]. Su objetivo es obtener la configuración óptima entre todas las posibilidades en la superestructura planteada.

Uno de los propósitos por el que se ha planteado la optimización de procesos es la obtención de las condiciones de operación óptimas [44]. Implementar una producción óptima puede reducir costos operacionales, aumentar los márgenes de mejora y mantenerse dentro de los estándares ambientales requeridos [45]. Las variables más comunes en este tipo de problemas son flujos y concentraciones [46]. Si existen modificaciones de layout, se pueden incorporar como variables del problema las ubicaciones y existencia de los equipos o plantas [47].

Entre los principales factores a considerar en el diseño de una planta de procesos está el layout, afectando los niveles de seguridad, condiciones económicas (principalmente los costos

de inversión) y optimización del espacio disponible [48]. Para la formulación matemática, usualmente se emplean variables binarias que pueden asociarse a decisiones sobre ubicaciones [48], conexiones entre equipos [49], selección de técnicas a emplear en los procesos [50], entre otras opciones.

Si se considera más de una función objetivo, el problema se vuelve multiobjetivo. Este tipo de problemas permite obtener la solución empleando distintos criterios, aunque presentan como limitación el costo computacional, que crece frente al número de funciones objetivo consideradas [51]. Para estos casos, se han aplicado análisis post-optimal desde una perspectiva de las soluciones de Pareto [52].

1.3.3.2. Supuestos en Optimización

En ingeniería, específicamente en la optimización de procesos, se han aplicado supuestos para facilitar el planteamiento y resolución de problemas. En la Tabla 1.2 se resumen aplicaciones de supuestos en optimización y su relación con los elementos de estos problemas.

1.3.3.3. Análisis de sensibilidad en optimización

De la Tabla 1.2 se destaca la existencia de supuestos que afectan parámetros de un problema de optimización. Como se mencionó en la sección 1.3.2, analizar el efecto de cambios en los parámetros de un problema de gran escala puede requerir un alto costo computacional.

Entre las preguntas principales asociadas a los efectos de cambios en parámetros de un problema están las siguientes [31]: ¿cuánto se modifica el valor de la solución si se cambia alguno de sus parámetros? y ¿cuánto es posible modificar un parámetro manteniendo el óptimo? Para responder a estas interrogantes es necesario explicar el siguiente término:

Rango o región de estabilidad

El concepto de estabilidad es relevante en el contexto de la optimización debido a posibles errores de precisión en datos iniciales, lejanía de los modelos matemáticos a la realidad, entre otros [53]. Esta propiedad se asocia a problemas cuya solución se mantiene pese a perturbaciones en los parámetros. Las variaciones que pueden adoptar los parámetros y que mantienen la optimalidad de la solución se conoce como *región de estabilidad* [54] o *rango de estabilidad* [55]. Se utilizarán indistintamente estos términos para referirse a la definición ya explicada. De todas formas, se considera posible hacer una distinción asociada al número de parámetros analizados. Si solo es uno (análisis OAT), se crearía un *rango* o intervalo, mientras que variaciones de más de un parámetro (GSA) generaría una *región*, dada la combinación de valores de los parámetros.

Habiendo introducido el concepto de rango de estabilidad, se analizarán las interrogantes mencionadas anteriormente para los diferentes tipos de problemas de optimización, cuyo nivel de respuesta difiere en cada uno:

Para problemas de programación lineal (LP), la mayoría de las estrategias de resolución están basadas en el *método simplex*. Se han creado distintos algoritmos y códigos para solucionar este tipo de problemas, que también permiten obtener información adicional mediante un análisis de sensibilidad o análisis post-optimal [56].

Tabla 1.2: Ejemplos de supuestos empleados en problemas de optimización en Ingeniería de Procesos y su relación con los elementos que lo componen. Donde P: parámetros, V: variables, R: restricciones y O: funciones objetivo.

Elemento				Supuesto	Explicación/ejemplificación	Ref.
P	V	R	O			
	X			Alimentación estequiométrica y sin inertes.	Si en la alimentación de un reactor con reciclo la fracción con la que se alimentan los reactantes es estequiométrica y no existen inertes, no es necesario incluir un flujo que represente la purga.	[57]
X				Mantenimiento de condiciones de entrada.	Se asume que las condiciones de operación (presión y temperatura) dentro del equipo están dadas por las establecidas en el flujo de entrada.	[57]
X				Eficiencia de operación es constante.	La eficiencia de un equipo deja de ser una variable, pasando a ser un parámetro, permitiendo simplificar el modelo.	[57]
		X		Mezcla isotérmica.	Es considerado que cada flujo en el mezclador se encuentra a la misma temperatura. Con esto se eliminan restricciones no lineales, aunque puede generar un resultado subóptimo.	[58]
X				Distribución de escenarios inciertos.	Ante la incertidumbre de algunas características en un problema de optimización, se asume una posible distribución para generar escenarios y resolverlo a partir de estos.	[59]
		X		Suministro ilimitado.	Para la resolución de problemas de optimización sin restricciones, se asume que es posible tener un acceso ilimitado al recurso que se está suministrando.	[60]

Jansen et al. [56] advierten que es necesario considerar dos métricas en conjunto para poder realizar conclusiones respecto a la sensibilidad de un problema LP: *precios sombra* (conocidos también como *precios duales* o *costos sombra*) y *rangos* (o *intervalos*) de los coeficientes del lado derecho (RHS³) de las restricciones o de la función objetivo⁴. Los *precios sombra* son interpretados como la variación del valor de la función objetivo cuando se efectúa un cambio

³ Por su sigla en inglés, proveniente de *right hand side*. Son los parámetros del vector de recursos de las restricciones.

⁴ Los coeficientes o parámetros de la función objetivo son conocidos también como vector de costos

en los coeficientes, siendo válida mientras éstos se encuentren dentro del rango permitido de los coeficientes del lado derecho. La utilización de una de estas métricas sin información sobre la otra podría ocasionar decisiones e interpretaciones erróneas [56]. De esta forma, las preguntas sobre los efectos de cambios de parámetros en problemas LP han sido analizadas y respondidas mediante el método y métricas mencionadas anteriormente.

En problemas de programación no lineal (NLP), el análisis de sensibilidad se ha estudiado según las condiciones necesarias y suficientes de optimalidad, siendo la base en estos problemas y la implementación de técnicas como el método de multiplicadores de Lagrange [61]. Las condiciones de suficiencia de segundo orden son indispensables para que las soluciones óptimas sean funciones diferenciables del parámetro analizado [62]. Particularmente, el trabajo presentado por Fiacco [63] recopila esfuerzos realizados sobre análisis de sensibilidad y estabilidad en problemas NLP. Teoremas presentados en este trabajo se basan en la continuidad y diferenciabilidad de las funciones involucradas, cuya extensión a problemas MINLP se complejiza por la naturaleza discontinua de las variables enteras y mixtas que componen las restricciones y/o funciones objetivo.

A diferencia del caso LP, en que se han generado trabajos y se encuentra establecida la forma de reconocer la región de estabilidad, para problemas MILP es más desafiante y se encuentra en evolución, cuya dificultad radica en el dominio discontinuo y raramente se ha utilizado en aplicaciones prácticas [31]. Una de las propuestas más recientes para el análisis de sensibilidad de problemas MILP corresponde al trabajo realizado por Yi y Lu [31]. En esta investigación se responden las dos preguntas destacadas anteriormente: cuánto cambia una solución frente a modificaciones en uno de los parámetros del problema y cuál es el rango o región de estabilidad de una solución óptima.

La estrategia propuesta por Yi y Lu [31] requiere el cálculo del óptimo del problema MILP. Una vez resuelto, se conoce la combinación de valores de las variables binarias. Posteriormente, se determina el rango de estabilidad de los parámetros mediante múltiples resoluciones del problema, esta vez empleando distintos valores en los parámetros. El algoritmo de búsqueda posee un paso que se adapta a medida que se aproxima al límite del rango de estabilidad. Como parte de la metodología del presente trabajo, en el Capítulo 3 es posible observar el diagrama con los pasos requeridos, ya que permitirá realizar comparaciones con el nuevo algoritmo que se creará.

La revisión bibliográfica efectuada determinó que no se han desarrollado estrategias para la obtención del rango o región de estabilidad de una solución en problemas MINLP. En términos generales, el estado del arte respecto a la determinación del rango de estabilidad de los parámetros permite reconocer la posibilidad de ampliar los trabajos desarrollados actualmente. Específicamente, podría ser extendida la propuesta de Yi y Lu [31], alcanzando casos MINLP, incorporando una forma más eficiente de determinar el rango de estabilidad de los parámetros. Particularmente, el funcionamiento iterativo podría ser modificado buscando una mejor estimación inicial al incluir características propias del modelo de optimización, que no han sido incorporadas hasta el momento (solo se utiliza el valor original del parámetro estudiado como punto inicial).

Además del desafío de la discontinuidad presente en problemas mixtos, los *solver*

empleados en problemas MINLP se ven afectados por las no linealidades de las expresiones involucradas [64]. Debido a la diversidad de consideraciones en el planteamiento, múltiples investigaciones se han elaborado para este tipo de problemas y se prevee que en la siguiente década se desarrollen y profundicen más trabajos en esta área [65]. Ante la evolución de los algoritmos para resolver los problemas MINLP, se propone que el primer acercamiento a la determinación del rango de estabilidad de parámetros en estos problemas se efectúe mediante un enfoque práctico, siendo complementado a futuro con nuevas investigaciones que se desarrollen al respecto. Con ello, se relacionará este tipo de análisis al contexto de la Ingeniería de Procesos.

Habiendo comprendido que los parámetros utilizados en el planteamiento de problemas de redes de agua generan distintos efectos en las variables, se estudiará cuánto se pueden modificar sin cambiar la solución, particularmente la asociada a las variables binarias. Al relacionarse este tipo de variables con instalaciones en la red, los cambios que producen los parámetros son estructurales y más difíciles y costosos de implementar en la realidad. En este contexto, se responderán las dos interrogantes planteadas previamente para profundizar el análisis de las decisiones en la red, buscando que aquellas decisiones que tienen un carácter de largo plazo se mantengan siendo óptimas.

Considerando los antecedentes mencionados, se buscará responder las dos interrogantes planteadas anteriormente para problemas MINLP. De esta forma, la novedad de este trabajo es la creación de una estrategia que permita determinar el rango de estabilidad de los parámetros empleados en un problema de optimización MINLP multiobjetivo de escala de ciudad, que considere las características de las ecuaciones presentes en el problema planteado para ser eficiente en cuanto a tiempo de resolución. Con ello, se podrá conocer bajo qué condiciones, en los parámetros provenientes de supuestos, el layout de la red de agua obtenida sigue siendo óptimo. La segunda novedad es la generación de un índice que compare los rangos de estabilidad, permitiendo establecer una priorización de los parámetros que deben ser analizados en mayor profundidad, y así lograr una mayor certeza sobre la resiliencia de la solución obtenida.

La creación de la estrategia mencionada y el índice de estabilidad corresponden a una de las primeras aproximaciones al análisis de sensibilidad de problemas de redes de agua formulados como MINLP. A partir del presente trabajo se podrían generar futuras extensiones que permitan estudiar otros modelos con este tipo de formulación.

Capítulo 2

Caso de estudio: Redes de agua

Una de las opciones para enfrentar la escasez de recursos hídricos es la optimización del uso de agua. Que todos los usuarios puedan contar con un suministro de agua depende de cuán eficiente es la red de distribución y tratamiento de agua (WDN⁵) que conecta las fuentes de agua con los consumidores. Estas redes están presentes en distintos procesos industriales y en escala de centros urbanos, teniendo como objetivo principal transferir agua a los nodos de consumo, que presentan distintas especificaciones [66]. Teniendo como motivación la necesidad de utilizar y distribuir eficientemente el agua, en este capítulo se introducirá la optimización de las redes de agua y los ejemplos que se usarán para explicar y aplicar el algoritmo que se propondrá en este trabajo.

2.1. Introducción

Los elementos básicos de una WDN corresponden a procesos o etapas que emplean agua, procesos de tratamiento, fuentes de agua, zonas de disposición, mezcladores y separadores de flujos. En algunos casos se incluyen zonas de almacenamiento de agua [67]. En la Figura 2.1 se representa una WDN según los elementos señalados anteriormente. Dada la creciente importancia de los recursos hídricos y materiales, la optimización ha surgido como un conjunto de técnicas que permiten un uso más eficiente de los recursos. [68].

Se destaca que el diseño de una red de agua o de algún otro tipo de proceso requiere tomar decisiones en distintos niveles. Desde el punto de vista operacional, se establecen los flujos y concentraciones óptimas; en cuanto a las decisiones estratégicas, se pueden efectuar sobre la instalación de plantas, equipos y sus respectivas conexiones, entre otras. Las decisiones estratégicas se realizan para un horizonte de tiempo mayor que las operacionales [69], debiendo efectuarse con un nivel de información suficiente para tener la seguridad de que la red propuesta es la mejor entre las distintas alternativas. Sin embargo, la información que se tiene durante la realización de un proyecto no necesariamente es completa [70].

Considerando la relevancia y permanencia en el tiempo de las decisiones estratégicas, los casos de estudio tendrán una formulación MINLP, examinando el comportamiento de las variables binarias, que pueden ser asociadas a modificaciones de layout. Los ejemplos específicos a analizar en este trabajo se describen a continuación.

⁵ Por su sigla en inglés, proveniente de *Water Distribution Network*. Particularmente, algunas redes poseen también etapas, equipos o plantas de tratamiento de agua.

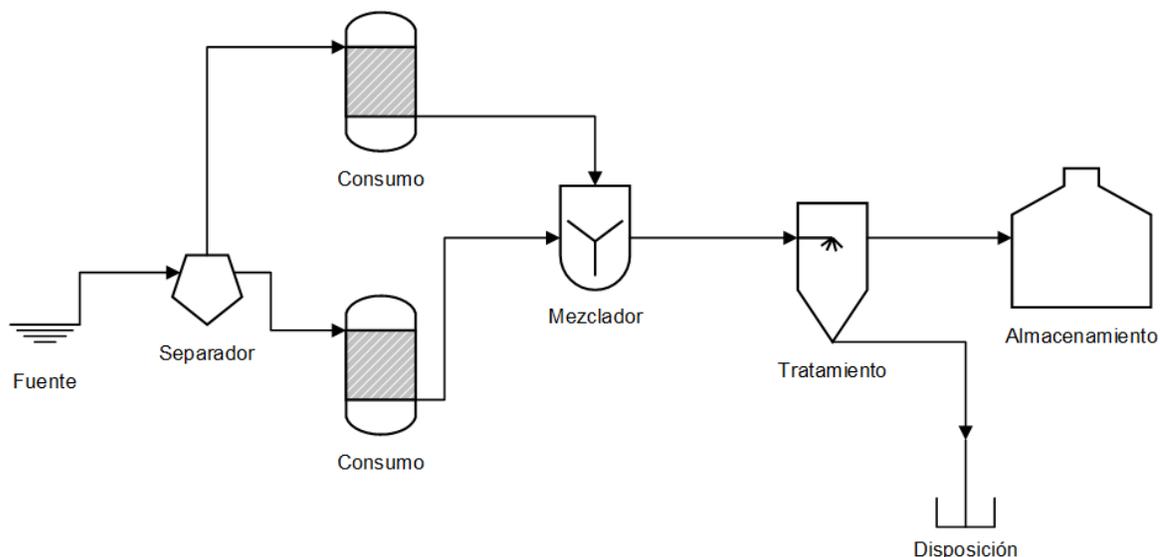


Figura 2.1: Representación de una red de distribución y tratamiento de agua. Elaboración propia con información de [67].

- Ejemplo ilustrativo 1: Es uno de los ejemplos de motivación de Kocis y Grossmann [33], siendo un problema MINLP a pequeña escala sin contexto específico [71]. El planteamiento del problema está disponible en el Anexo C.
- Ejemplo ilustrativo 2: Es un problema de red de agua de pequeña escala inspirado en los problemas planteados por Karuppiah y Grossmann [72]. Se formula como un problema MINLP en que se decide la instalación de plantas de tratamiento a través de variables binarias, considerando criterios ambientales y económicos. El planteamiento completo del problema se encuentra en el Anexo D.
- Ejemplo ilustrativo 3: Es un problema de red de agua con múltiples contaminantes, obtenido de Ahmetovic y Grossmann [73]. Las decisiones principales corresponden a la instalación de tuberías entre las plantas de procesos y tratamiento, representando las conexiones óptimas de la red que permitan disminuir los costos de instalación y operación. El planteamiento del problema está disponible en el Anexo E.
- Ejemplo ilustrativo 4: Es un problema de optimización multiobjetivo y de escala de ciudad, correspondiente a la red de agua de Santiago de Chile. Fue propuesto inicialmente por Gormaz-Cuevas et al. [74] como un problema MILP. Posteriormente, se planteó como un problema MINLP por Gormaz-Cuevas et al. [75], analizando diferentes simplificaciones para su resolución. Este último caso es el que se utilizará en la presente tesis, debido al tipo de formulación que se está analizando (MINLP). El propósito del problema es obtener los flujos, concentraciones y la ubicación de plantas (de recolección, tratamiento y distribución) que permitan minimizar el uso de agua desde fuentes naturales, el potencial de calentamiento global y el costo total de la red. El planteamiento del problema se encuentra en el Anexo F, siendo analizado y complementado con modificaciones para su inclusión en la presente tesis.

Los primeros tres ejemplos se emplearán en los Capítulos 4 y 6 para comparar el algoritmo referencial y el propuesto, mientras que el último se utilizará en el Capítulo 7 para validar y concluir la estrategia propuesta. Para comprender cada problema se presentan a continuación los principales antecedentes relacionados con optimización en el diseño de redes de agua.

2.2. Optimización de redes de distribución de agua

La optimización de WDN busca la mejor forma de transportar agua desde las fuentes a los usuarios, satisfaciendo sus requerimientos [76], y ha sido analizada por distintos autores. En 1980, Takama et al. [77] propusieron la base para el diseño óptimo de redes de agua, siendo los primeros en formular un problema de optimización de redes de distribución de agua que incluyera su uso y tratamiento en la refinería de petróleo [73][78]. Desde entonces se han elaborado distintos *reviews*, resumiéndose los principales a continuación:

En el año 2000, Bagajewicz [79] realizó una recopilación de información sobre trabajos asociados a la optimización de redes de agua en procesos, con énfasis en la programación matemática y las condiciones para alcanzar optimalidad global. Posteriormente, en 2009, Foo [80] analizó el estado del arte de las técnicas heurísticas para la síntesis de redes de agua basadas en el análisis Pinch, en problemas de flujo constante y con un contaminante.

Detectando la necesidad de incluir el estudio de redes de agua en procesos que no son continuos, en 2010 Jezowski [67] elaboró un tercer review para complementar los trabajos anteriores, agregando redes con funcionamiento batch. Una siguiente actualización al estado del arte fue efectuada en 2014, cuando Khor et al. [81] evaluaron la contribución general de los trabajos realizados hasta el momento sobre optimización de procesos, con enfoque en las redes de agua, destacando los hitos que han ocurrido en esta área y los desafíos futuros.

Entre los más recientes, se destaca en el año 2019 el review elaborado por Bhojwani et al. [82], que permite complementar los trabajos anteriores al estudiar distintas tecnologías que se pueden emplear en el tratamiento de agua. En este se consideran métricas económicas, ambientales y de rendimiento para cuantificar el comportamiento de las redes de distribución y tratamiento de agua con la variedad de alternativas recopiladas.

En los reviews mencionados anteriormente se destaca que, tras el trabajo propuesto por Takama et al. [77], transcurrieron varios años sin que se registraran publicaciones enfocadas en programación matemática para el planteamiento de este tipo de problemas [79]. Dentro de las más destacadas se encuentran las siguientes:

En 1997, Doyle y Smith [83] formularon un modelo de redes de agua como un problema de programación no lineal (NLP). Dada la dificultad de resolución de los casos NLP, presentaron tanto un enfoque lineal como no lineal. De forma complementaria, en 1998, Alva-Argáez et al. [84] plantearon un modelo MINLP y una estrategia para descomponerlo en una secuencia de problemas MILP y aproximarse a una solución óptima.

Continuando con el desafío de superar la complejidad de resolución que tienen los problemas no convexos [42] [84], en 2006 Karuppiah y Grossmann [72] propusieron una estrategia en la que los problemas de uso y tratamiento de agua se resolvían en conjunto. En esta publicación presentaron un algoritmo de *branch and contract* que establece cotas superiores e inferiores para efectuar una relajación convexa (envoltura convexa de McCormick [85]) y así resolver el problema en la búsqueda del óptimo global.

Ahmetovic y Grossmann [73] en 2010 utilizaron una extensión y generalización de la

superestructura propuesta por Karuppiah y Grossmann [72], añadiendo cotas propuestas por ellos para resolver problemas de optimización NLP y MINLP. Presentaron una estrategia de dos pasos para resolver problemas MINLP a gran escala, resolviendo la versión relajada y utilizando esos resultados en una siguiente ejecución del problema.

Supuestos

El diseño y optimización de redes de agua requiere la implementación y justificación de supuestos, que se emplean con el objetivo de simplificar la geometría, planteamiento y/o resolución del problema estudiado. Por ejemplo, a nivel mundial se han utilizado modelos hidráulicos de WDN basados en el estado estacionario para respaldar las decisiones óptimas de diseño y gestión [86]. En la Tabla 2.1 se resumen algunos supuestos empleados en la optimización de redes de agua y su relación con los elementos de un problema de este tipo.

Tabla 2.1: Ejemplos de supuestos empleados en problemas de optimización de redes de agua y los elementos en que influyen. Donde P: parámetros, V: variables, R: restricciones y O: funciones objetivo.

Elemento				Supuesto	Explicación	Ref.
P	V	R	O			
	X			Mezcla perfecta	Se asume una mezcla instantánea y perfecta de contaminantes biológicos y químicos en las tuberías y conexiones.	[87]
X				Flujo pistón	Se desprecia la dispersión longitudinal a lo largo del eje de desplazamiento.	[87]
X				Demanda fija	Se asume demanda de agua constante durante el análisis del problema, siendo independiente de la presión.	[88]
		X		Costo de mantención	Se estima que el costo de mantención de una WDN corresponde a un porcentaje del costo de inversión.	[89]
X				Estado estacionario	Se asume que el flujo másico total que ingresa a un nodo es igual al flujo másico total de salida. Es decir, no hay acumulación.	[90]

Finalmente, se destaca que en el caso específico del diseño óptimo de redes de agua, no se encontraron aplicaciones de un análisis cuantitativo del efecto de supuestos. De esta forma, se considera oportuno explorar alternativas que permitan determinar cómo afecta la inclusión de supuestos en el resultado de un problema de redes de agua, particularmente en la asignación de valores en parámetros de aquellos formulados como un problema MINLP.

Capítulo 3

Metodología

En el algoritmo que se creará, la región de estabilidad es calculada para aquellos parámetros cuyos valores se encuentran asignados a partir de supuestos. Inicialmente, la búsqueda bibliográfica permitió determinar que no existen herramientas desarrolladas para este propósito en aplicaciones prácticas de problemas MINLP. La mejor alternativa encontrada para calcular el rango de estabilidad es la propuesta por Yi y Lu [31], que determina este rango para parámetros en problemas MILP. Dado que no han sido creados trabajos para casos MINLP, el algoritmo referencial MILP se ha escogido para una comparación inicial y fuente de inspiración. Su funcionamiento se explica a continuación:

3.1. Algoritmo referencial

El algoritmo de Yi y Lu [31] evalúa distintos valores de los parámetros según pasos de búsqueda y tolerancia. Los pasos corresponden a la diferencia entre un valor del parámetro y el siguiente, mientras que la tolerancia es el valor mínimo que puede adoptar el paso. Esta información permite ejecutar el problema con un nuevo valor del parámetro, definido a partir del paso inicial. Continuará la búsqueda del límite del rango de estabilidad (inferior o superior, dependiendo de si se está disminuyendo o aumentando el parámetro, respectivamente) con el mismo paso hasta producir un cambio en las variables binarias. Esto significa que el valor evaluado está fuera del rango de estabilidad, por lo que se regresa al valor anterior y se avanza desde ese punto con un paso de menor tamaño. Se repite el proceso hasta que el paso alcanza la tolerancia, obteniéndose el límite buscado, ya que no es posible continuar iterando sin generar un cambio en la solución. Los elementos presentes en el algoritmo se explican a continuación, mientras que los pasos que lo componen se observan en la Figura 3.1.

Sea $Y = f(x, a)$ una función que depende de la variable x y del vector de parámetros $a = (a_1, \dots, a_n)$. Para determinar el rango en el que la variable x no cambia de valor óptimo (x_0) se realizan los pasos de la Figura 3.1, estableciéndose inicialmente un contador de iteraciones por cada largo de paso, denotado por i e inicializado en 0, además de un paso Δl y una tolerancia t . Estos dos últimos parámetros se definen de la siguiente forma [31]:

- Largo del paso de búsqueda (Δl): Es el número más pequeño que posea el mismo orden de magnitud del parámetro analizado.
- Tolerancia (t): 4/10000 veces el valor nominal del parámetro de entrada a estudiar. Este valor se sugiere a partir de distintas pruebas de ensayo y error.

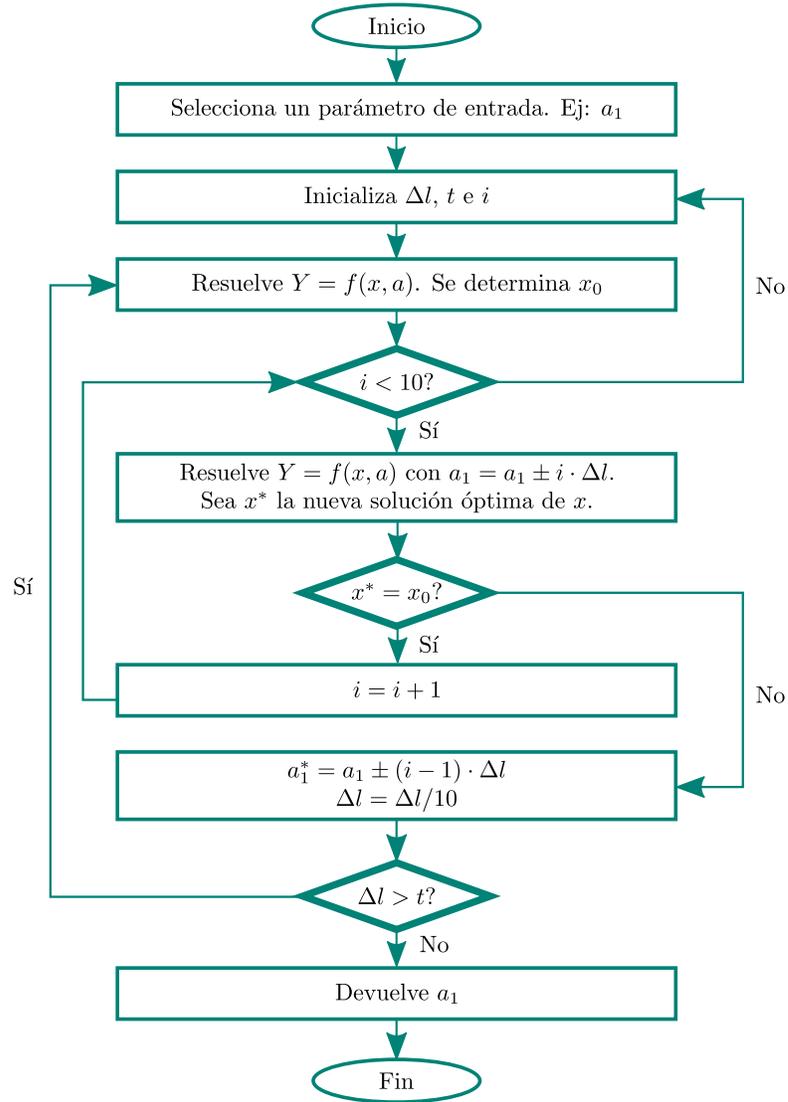


Figura 3.1: Algoritmo para obtener el rango de estabilidad de parámetros en problemas MILP [31].

El algoritmo propuesto por Yi y Lu [31] se utilizará para obtener una respuesta referencial, pero podría no ser lo suficientemente eficiente en cuanto a costo computacional para determinar el rango de estabilidad en problemas MINLP. Mediante su implementación en tres ejemplos, analizados en el Capítulo 4, se realizará un diagnóstico del desempeño del algoritmo de referencia para revelar el origen de posibles dificultades que podría tener al momento de ser empleado en casos MINLP, siendo una fuente creativa para el nuevo algoritmo que se presenta en el Capítulo 5. Entre las dificultades que podrían ser significativas se destaca la no convexidad propia de estos problemas y el tiempo requerido por las múltiples resoluciones.

El nuevo algoritmo incluirá pasos adicionales a los presentes en el trabajo de Yi y Lu [31], buscando ser eficiente para casos MINLP. Con estos nuevos pasos se determinará una nueva adivinanza inicial para acelerar la búsqueda, además de acotar el tiempo de resolución de cada iteración y el rango de búsqueda con información del contexto del problema. Junto con ello, se obtendrá una comparación entre los rangos de cada parámetro analizado. La metodología y herramientas utilizadas son descritas a continuación:

3.2. Determinación del rango de estabilidad

El rango de estabilidad es definido sobre el comportamiento de las variables binarias, mientras que cambios en otras variables son permitidos. Esta priorización se basa en la relación entre las variables binarias y las decisiones estratégicas, correspondientes a decisiones que son difíciles o complejas de cambiar si la solución implementada no fuese la mejor. Por lo tanto, es esencial confirmar que las decisiones son correctas incluso si los parámetros cambian su valor. Entonces, mientras las variables binarias mantengan su valor original, se considera que el valor del parámetro se encuentra dentro de su región de estabilidad.

El efecto de los parámetros en la solución se reconoce mediante evaluaciones iterativas modificando un parámetro a la vez. Las iteraciones corresponden a la resolución del problema de optimización para diferentes valores de los parámetros. Estas iteraciones continúan hasta que un cambio en las variables binarias sea detectado, inspirándose este procedimiento en el algoritmo de Yi y Lu [31]. El paso inicial de búsqueda y la forma de calcular la tolerancia es la misma, aunque el punto inicial de las iteraciones cambiará en la nueva propuesta. Para ello, se resuelve un subproblema que fija la solución de las variables binarias y considera como variable el parámetro en estudio. La formulación de este subproblema se discutirá en el Capítulo 5, mientras que sus resultados se analizan y emplean en el Capítulo 6.

La resolución original del problema permite establecer un tiempo de ejecución referencial para las iteraciones y el subproblema. Este tiempo referencial se incluye en el algoritmo propuesto para limitar el tiempo de cada iteración a la escala y tiempo del caso analizado. En consecuencia, su inclusión permite reducir el tiempo de implementación del algoritmo.

El rango de búsqueda es acotado por información previa que se conozca de estos. Por ejemplo, algunas cotas físicas y lógicas se incorporan para analizar solamente el rango posible de valores de los parámetros en el contexto de cada problema y evitar iteraciones en valores infactibles, consiguiendo reducir el tiempo utilizado por el algoritmo.

3.3. Cálculo del índice de estabilidad de parámetros

Una comparación de los rangos de estabilidad de los parámetros es necesaria para reconocer el parámetro que presenta un mayor riesgo de ocasionar una solución óptima diferente. Sin embargo, la comparación no puede realizarse directamente con los valores del rango de estabilidad, debido a posibles órdenes de magnitud diferentes de los distintos parámetros. Por lo tanto, se sugiere normalizar aquellos valores para evitar interpretaciones erróneas.

Un índice es propuesto para comparar la estabilidad de los parámetros cuantificando el riesgo que presentan de cambiar la solución. La idea principal es calcular el margen de error utilizando el valor nominal y los extremos del rango de estabilidad. Este índice será calculado como la diferencia normalizada entre el valor nominal y el límite más cercano. Específicamente, permite conocer cuánto podría cambiar un parámetro sin generar una solución óptima diferente. Por lo tanto, un mayor índice de estabilidad representa un mayor margen de error en la solución y resiliencia para la solución óptima.

3.4. Recursos computacionales e implementación

La resolución de cada problema de optimización se efectuará mediante el software GAMS [91] y el solver BARON [92], especializado para problemas MINLP. Para actualizar el valor de los parámetros según el comportamiento de las variables se utilizará el software Python [93]. Ambos programas se implementan en un computador con procesador Intel Core I7-7700 HQ con 16 GB de memoria RAM y cuya interacción se describe a continuación:

Python se utilizará para declarar el modelo y los valores nominales de los parámetros, además de ejecutar GAMS con esta información. Una vez obtenidos los resultados, se actualizarán los valores de los parámetros en Python y se obtendrán nuevas soluciones. Dependiendo del cambio o permanencia en los valores de las variables binarias, se ejecutará nuevamente el modelo con diferentes valores de los parámetros. La ventaja de utilizar Python en este proceso corresponde al manejo de las soluciones, la programación de las condiciones que se requieren para las múltiples ejecuciones y la interpretación gráfica de las soluciones.

Uno de los factores a considerar y ya destacado previamente es el tiempo de resolución. El tiempo máximo por defecto en GAMS es de 1.000 segundos⁶, que podría alcanzarse dependiendo del tamaño y otras características del problema. Para poder realizar la comparación de tiempos entre el algoritmo referencial y el propuesto, se mantendrá este límite para el caso referencial y para la resolución original en el nuevo algoritmo. En el resto del procedimiento del nuevo algoritmo se utilizará un nuevo límite de tiempo, estimado a partir del tiempo de ejecución del caso original. Las distintas condiciones que se pueden obtener en la solución debido a la influencia del tiempo límite y características del problema se resumirán mediante la siguiente denominación:

1. *Mantiene*: La solución obtenida es óptima y concuerda con los valores originales de las variables binarias.
2. *Cambia*: La solución obtenida es óptima y no todos los valores de las variables binarias concuerdan con el caso original.
3. *Interrumpido*: La solución obtenida es óptima, aunque se interrumpió el proceso de búsqueda por alcanzar el límite de tiempo. Para conocer si se mantiene o cambia la solución óptima, se deben observar los valores obtenidos.
4. *Entera*: El proceso de búsqueda alcanzó el tiempo límite y la solución obtenida corresponde al mejor valor (que minimiza o maximiza) que incluya soluciones enteras, dado el tipo de variables involucradas (binarias). Al igual que el caso anterior, esto se puede generar tanto para valores que mantienen o cambian la solución.
5. *Infactible*: El valor evaluado para el parámetro genera que la solución sea infactible.

Se destaca que el solver escogido es BARON considerando las características de los cuatro problemas a utilizar en la tesis. Particularmente, los autores del caso de estudio principal consideraron aquel solver. De todas formas, para otros casos en que fuese conveniente utilizar un solver diferente, el algoritmo se adaptará, ya que la decisión está relacionada con el propio problema y el algoritmo empleará el mismo solver del problema original en cada iteración.

⁶ En algunos casos las resoluciones podrían superar levemente este valor, considerándose un tiempo referencial aproximado en el software.

3.5. Validación del algoritmo

La validación del algoritmo se realizará a partir de su aplicación en el caso de estudio principal en el Capítulo 7, mediante dos aspectos principales: predicción y demostración. La predicción se efectuará determinando el rango de estabilidad con el algoritmo a proponer, mientras que la demostración se realizará evaluando la solución del problema para 100 valores de los parámetros, generados al azar. Los escenarios que se producirán son los siguientes:

1. El valor evaluado es menor al límite inferior del rango de estabilidad.
2. El valor evaluado es mayor al límite superior del rango de estabilidad.
3. El valor evaluado se encuentra entre el límite inferior y superior del rango mencionado.

En los casos 1 y 2, la solución debería presentar valores en las variables binarias distintos a los obtenidos con los valores nominales de los parámetros. Podrían existir casos que generen la misma solución, pero separados del rango de estabilidad por valores que cambiaron la solución, es decir, no forman parte de un rango continuo en torno al valor nominal. Por el contrario, en el caso 3 la solución deberá presentar la misma configuración respecto a las variables binarias que en el escenario original. Gráficamente, se aspira a que las evaluaciones para diferentes valores de parámetros presenten el comportamiento observado en la Figura 3.2, donde se ha convenido que si la solución mantiene el valor de las variables binarias, adopta el valor 1 en el gráfico. En caso contrario, se expresa con un valor de 0. Particularmente, se espera que los casos en que las evaluaciones se señalan como *Interrumpido* y *Entera* se minimicen, debido a una búsqueda eficiente del rango.

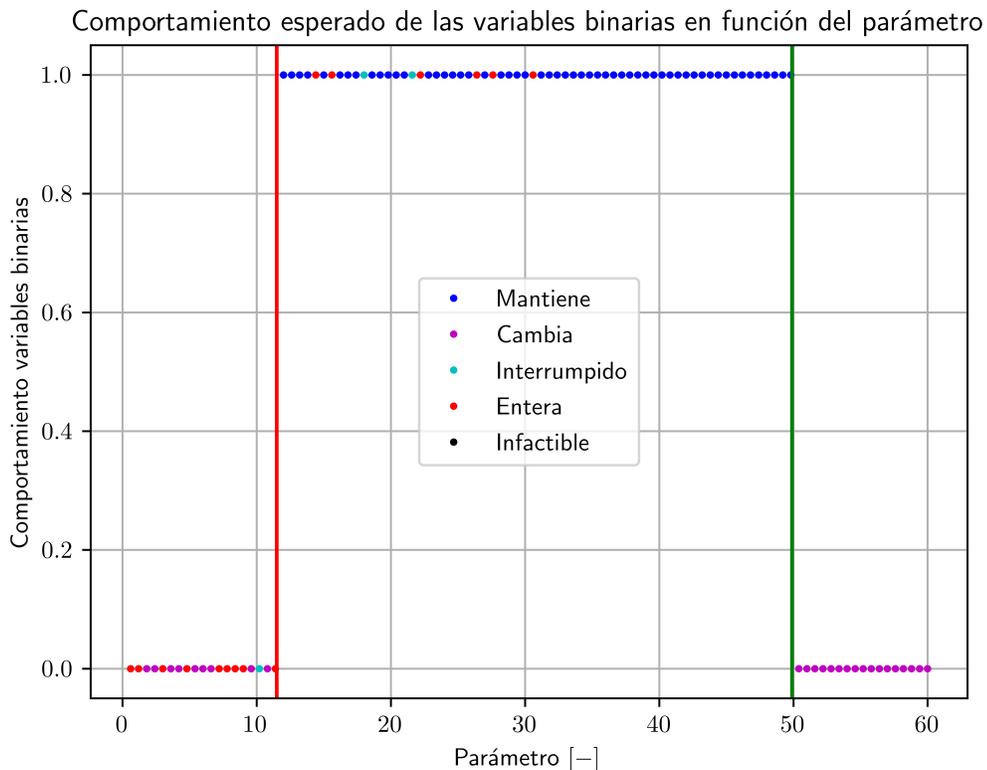


Figura 3.2: Comportamiento esperado de las variables binarias en relación al rango de estabilidad determinado. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

Capítulo 4

Diagnóstico del algoritmo referencial

El propósito de este capítulo es analizar el desempeño del algoritmo de Yi y Lu [31] en tres ejemplos. Al presentar cada uno se destacarán las características que poseen y los cambios que requiere el algoritmo referencial para tener un funcionamiento más eficiente para las condiciones descritas. El desarrollo y las reflexiones de cada caso permitirán comprender la necesidad de las modificaciones anunciadas, para luego ser incluidas en el algoritmo propuesto en el Capítulo 5.

Los ejemplos escogidos presentan diversas características que generarán diferentes desafíos para el algoritmo referencial. Las principales radican en las decisiones asociadas a las variables binarias y el tamaño de los problemas, resumidas en la Tabla 4.1. Se espera que aquellos con un mayor número de variables binarias y ecuaciones generen mayores dificultades para que el algoritmo determine el rango de estabilidad, representadas en mayor tiempo o número de iteraciones para alcanzar cada límite (inferior y superior) con la precisión requerida.

Tabla 4.1: Características de los ejemplos utilizados para evaluar el desempeño del algoritmo referencial.

Ej.	Descripción	Nº var. binarias	Nº ecuaciones
1	Problema de pequeña escala, sin contexto específico. Se utiliza, principalmente, para ejemplificar cómo se emplea el algoritmo referencial.	1	3
2	Problema de red de agua, cuyas variables binarias se utilizan para representar la decisión de instalar dos plantas de tratamiento.	2	28
3	Problema de red de agua que presenta plantas ya instaladas y las decisiones asociadas a variables binarias corresponden a las posibles conexiones entre los nodos de la red.	29	247

Previamente, en el Capítulo 3, se ha relatado cómo se calcula el rango de estabilidad mediante el algoritmo de Yi y Lu [31]. A continuación, este procedimiento será ejemplificado a través de su implementación en un problema de pequeña escala, permitiendo comprender y representar los pasos a seguir en los ejemplos ilustrativos 2 y 3 de esta sección.

4.1. Ejemplo ilustrativo 1

El planteamiento propuesto por Kocis y Grossmann [33] corresponde a un problema MINLP de pequeña escala sin contexto específico. Inicialmente, al no poseer un contexto particular, no se reconocen supuestos y es utilizado en esta sección con el fin de explicar cómo se comporta la solución del problema frente a modificaciones en sus parámetros. Particularmente, para ejemplificar cómo se determina el rango de estabilidad de los parámetros mediante el algoritmo referencial.

Respecto a una posible mejora, el diagrama del funcionamiento del algoritmo de Yi y Lu [31] señala que las iteraciones se efectúan a partir del valor nominal del parámetro estudiado, tanto en la búsqueda del límite inferior como el superior. Comprendiendo que el iniciar la búsqueda desde un valor más cercano al límite a determinar permitiría reducir el número de iteraciones necesarias, ¿sería posible establecer una mejor estimación inicial? Las ideas en que se basará esta potencial mejora se plantearán a partir de la estructura de este primer ejemplo y de los resultados obtenidos mediante el algoritmo referencial.

La estructura y formulación matemática del problema se encuentra en el Anexo C.1, mientras que se ha adaptado una versión para el presente trabajo según la expresión 4.1. La diferencia entre la formulación original y la adaptación radica en los parámetros a y b . En el caso original estas expresiones poseen los valores de 1,25 y 1,6, respectivamente. En esta sección se denominan como a y b para distinguir los parámetros a analizar y caracterizar cómo se modifica la solución del problema frente a cambios en estos. La solución de este problema es $z = 2$, obtenida con $x = 0,5$ e $y = 1$, destacada en el Anexo C.2. Resolviendo este ejemplo para distintos valores iniciales de las variables se obtiene el mismo óptimo global [71].

$$\left\{ \begin{array}{ll} \min & z = 2x + y \\ & x, y \\ \text{s.a.} & \\ & \mathbf{a} - x^2 - y \leq 0 \\ & x + y \leq \mathbf{b} \\ & x \geq 0 \\ & y \in \{0, 1\} \end{array} \right. \quad (4.1)$$

Si en este caso algunos de los parámetros fueran asumidos a partir de supuestos y la variable binaria y se relacionara con la instalación de alguna planta, se desearía conocer cuánto podrían variar los parámetros para que la solución obtenida en la variable binaria fuese la misma. Entonces, para los parámetros destacados anteriormente la pregunta por dilucidar es: ¿cuál es el rango de valores de a y b con el que la variable binaria y continúa teniendo un valor igual a 1?

El algoritmo existente para responder la interrogante anterior en casos MILP es el propuesto por Yi y Lu [31]. Pese a que el problema descrito es MINLP, al no existir otro algoritmo adaptado para este tipo de problemas, se aplicará el algoritmo referencial para conocer su funcionamiento y desempeño en este tipo de casos y así explorar los cambios

requeridos para una implementación eficiente en modelos mixtos y no lineales.

4.1.1. Rango de estabilidad

La aplicación de esta metodología se inicia con la determinación de los siguientes parámetros, para luego ser utilizados en los pasos señalados en el Capítulo 3:

- Largo del paso de búsqueda (Δl): Es el número más pequeño que posea el mismo orden de magnitud del parámetro analizado. Para el caso de a y b corresponde a 1.
- Tolerancia (t): $4/10000$ veces el valor nominal del parámetro de entrada a estudiar. Para el caso de a es $0,0005$ y para el parámetro b la tolerancia es $0,00064$.

Los parámetros anteriores se complementan con los resultados obtenidos originalmente en las variables de interés. En este caso, la variable binaria y adopta un valor de $y_0 = 1$ como solución del problema, previo a las modificaciones. Este valor será comparado en cada iteración con y' , correspondiente a la solución determinada para la variable y con cada valor del parámetro en estudio. El mencionado valor del parámetro a (b) será actualizado a a' (b'), según el cumplimiento de la condición de que no cambie la variable binaria.

La utilización de estos parámetros para la búsqueda del límite inferior de a con el algoritmo de Yi y Lu [31] genera las iteraciones expresadas en la Tabla 4.2, donde se destaca el número de iteraciones (i) en cada ciclo, el valor evaluado en cada iteración (a), el valor del paso (Δl) y su comparación con la tolerancia correspondiente. También se incluye el tiempo empleado en cada iteración y el total utilizado para determinar el límite inferior del rango de estabilidad.

Tabla 4.2: Determinación del límite inferior del rango de estabilidad del parámetro a , con b constante.

i	a	$y' = y_0?$	a'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,25	Sí	–	–	0,26
1	0,25	Sí	–	–	0,26
2	-0,75	No	0,25	$0,1 > 0,0005$	0,08
Ciclo 2: $\Delta l = 0,1$					
0	0,25	Sí	–	–	0,24
1	0,15	No	0,25	$0,01 > 0,0005$	0,27
Ciclo 3: $\Delta l = 0,01$					
0	0,25	Sí	–	–	0,27
1	0,24	No	0,25	$0,001 > 0,0005$	0,27
Ciclo 4: $\Delta l = 0,001$					
0	0,25	Sí	–	–	0,26
1	0,249	No	0,25	$0,0001 < 0,0005$	0,30
Tiempo total: 2,21[s]					

De la Tabla 4.2 se observa que, variando solamente el parámetro a (b se mantiene constante), el valor de la variable binaria y se mantiene mientras $a \geq 0,25$. Para valores

inferiores, y tiene un valor de 0. La detención de búsqueda del rango de estabilidad ocurre tras evaluar el valor de $a = 0,249$, debido a que el algoritmo buscaría un nuevo valor desde 0,25, pero con un paso igual a 0,0001, que sería inferior a la tolerancia establecida.

El procedimiento para obtener el límite superior del rango de estabilidad es similar al realizado para el inferior, aunque los pasos se efectúan para incrementar el parámetro. El tiempo total para analizar el efecto del aumento del parámetro a es de 2,06[s], determinándose que la cota superior del rango de estabilidad corresponde a 1,36, según se reporta en las evaluaciones del Anexo C.3. Por lo tanto, el rango de estabilidad de a es el siguiente:

$$0,25 \leq a \leq 1,36 \quad (4.2)$$

Con el propósito de corroborar el rango obtenido a través del algoritmo, se resolvió el problema utilizando 100 valores arbitrarios del parámetro a y equidistantes, cuyo efecto en la variable binaria se refleja en la Figura 4.1. Tal como se esperaba, los valores que se encuentran fuera del rango de estabilidad calculado provocan un cambio en la variable binaria, coincidente con la explicación realizada en el Capítulo 3.

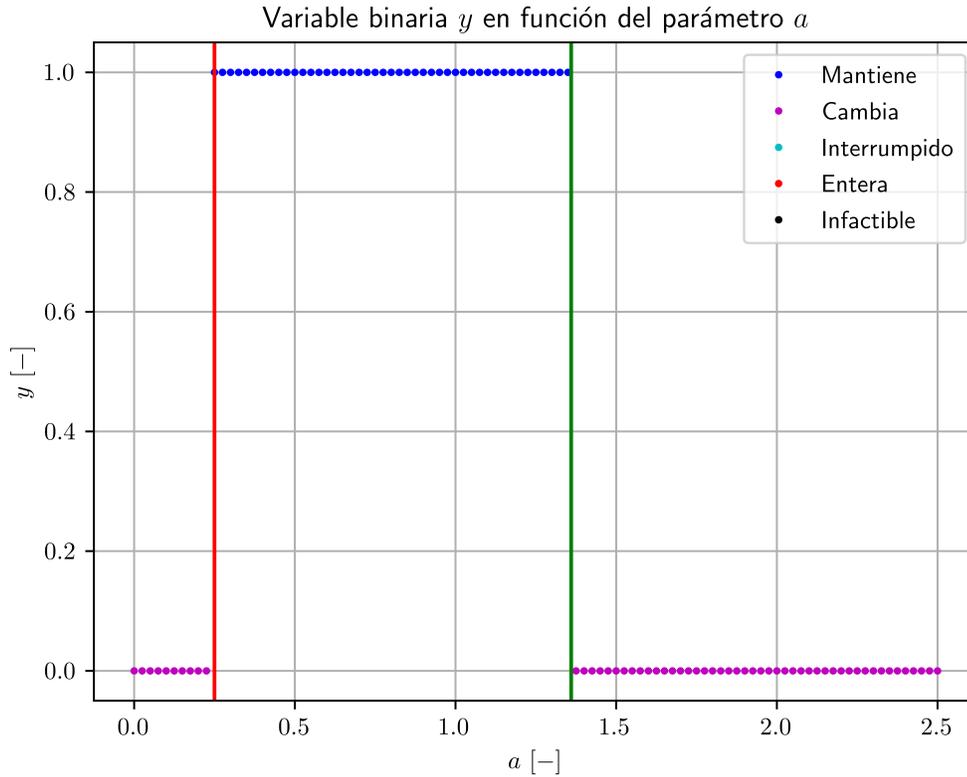


Figura 4.1: Variación de la variable binaria y frente a cambios en el parámetro a . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

Para el caso del parámetro b , se realizó el mismo procedimiento que para a . El detalle de cada iteración se encuentra en el Anexo C.3, donde se señala que el menor valor asignado a este parámetro para que la variable binaria no cambie es 1,5 y el tiempo de ejecución corresponde a 1,98[s]. Respecto al límite superior, no existe un cambio en la variable binaria para los valores evaluados. Ante esta situación, el algoritmo de Yi y Lu [31] sugiere escoger nuevamente otros

valores de Δl , t y reiniciar i , con el propósito de continuar la búsqueda del límite superior del rango de estabilidad. El tiempo reportado para el límite superior es de 2,74[s] y no considera nuevas actualizaciones, siendo registrado para su posterior comparación con el algoritmo a proponer, excluyendo el tiempo de nuevas evaluaciones arbitrarias que dificulten el contraste de resultados.

Para examinar el comportamiento de la solución con valores mayores a los alcanzados en las iteraciones iniciales, se realizaron nuevas resoluciones del problema y se reconoció que para valores superiores del parámetro no se generaron cambios en la variable binaria, sugiriéndose que no está acotado superiormente el rango de estabilidad de b . De esta forma, el rango es el presentado en la ecuación 4.3 y la representación gráfica se encuentra en el Anexo C.3, donde se comprueba que el rango calculado concuerda con lo obtenido para múltiples evaluaciones arbitrarias del parámetro en estudio, similar a la evaluación realizada para a .

$$1,5 \leq b \quad (4.3)$$

4.1.2. Efecto del cambio de parámetros en las restricciones

Habiéndose determinado el rango de estabilidad de a y b , a continuación se analizará el efecto que producen los cambios de los valores de ambos parámetros en las restricciones 4.4 y 4.5, correspondientes a las expresiones en que se encuentran directamente involucrados los parámetros de interés. Con este estudio se pretende establecer las bases que permitan proponer una metodología para determinar una mejor adivinanza inicial para la búsqueda de los extremos del rango de estabilidad.

$$a - x^2 - y \leq 0 \quad (4.4)$$

$$x + y \leq b \quad (4.5)$$

Las expresiones anteriores son graficadas en la Figura 4.2 para cuatro casos diferentes, destacándose en estos los siguientes puntos de interés: El punto C representa la intersección de la restricción 4.4 y la recta $y = 1$, correspondiente a la solución óptima del problema original; el punto D es la intersección de la restricción 4.5 y la recta $y = 1$, por lo que la solución óptima se podría mantener en el trazo \overline{CD} ; E es la intersección de la restricción 4.4 y la recta $y = 0$; F es la intersección de la restricción 4.5 y la recta $y = 0$. Debido a que la variable y es binaria, las posibles soluciones solamente se encuentran en los trazos \overline{CD} y \overline{EF} , y se ha marcado de color gris la zona intermedia solamente para favorecer la visualización de las dimensiones de los trazos mencionados, correspondientes a la región factible. Los cuatro gráficos que conforman la imagen consideran distintos valores del parámetro b , siendo descritos a continuación:

- (a) $b = 1,6$: Es el valor nominal del parámetro en estudio y la solución óptima se representa en el punto C , estando activa la restricción 4.4 y no activa la expresión 4.5.
- (b) $b = 1,5$: Al realizar una disminución del parámetro b hasta el valor de 1,5, C y D corresponden al mismo valor. De esta forma, la restricción 4.5 pasa a estar activa y la solución óptima se mantiene.
- (c) $b = 1,3$: Cuando b es inferior a 1,5 ya no es factible una solución que mantenga el valor de la variable $y = 1$. Así, la nueva solución óptima se encuentra en el punto E .

(d) $b = 1,119$: Es el valor más pequeño de b con el que la solución es factible. En este caso los puntos E y F se encuentran en el mismo lugar, teniéndose las dos ecuaciones destacadas anteriormente como activas, aunque la solución $y = 0$ es diferente a la original.

Restricciones del ejemplo 1 frente a cambios del parámetro b

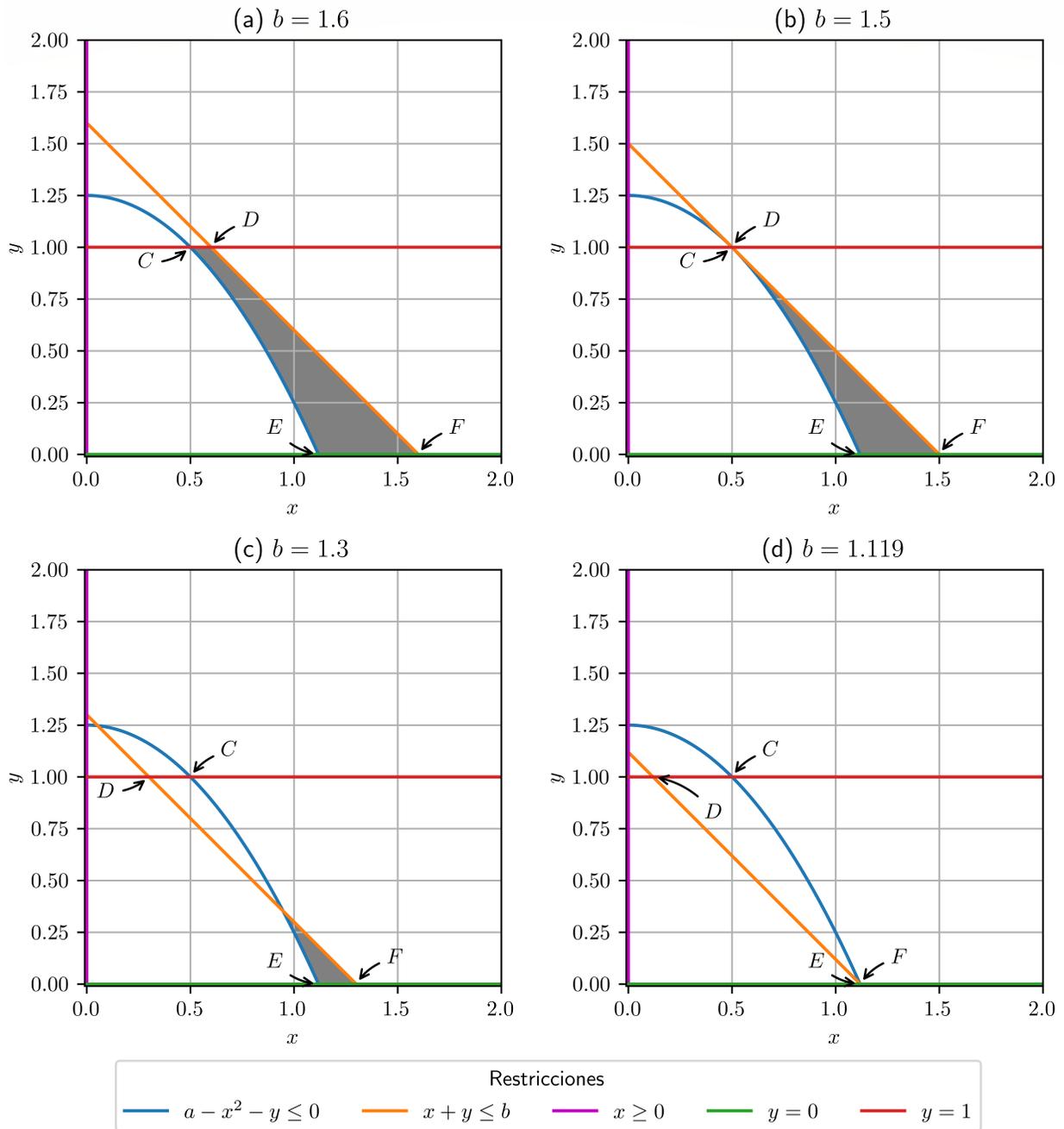


Figura 4.2: Restricciones del ejemplo 1 frente a cambios del parámetro b .

En la Figura 4.2.(b) se observa que al pasar a estar activa la restricción 4.5 se alcanza el límite inferior del rango de estabilidad de b calculado previamente por el algoritmo de Yi y Lu [31]. Los casos en que b aumenta desde el valor original no son analizados ya que la solución óptima continuaría siendo la observada en el punto C.

Para complementar lo analizado en el caso del parámetro b , en la Figura 4.3 se observa cómo se comportan las restricciones y la solución para los cambios en el parámetro a , el que se ha caracterizado por presentar un rango de estabilidad acotado superior e inferiormente. Para este caso los puntos C , D , E y F representan las mismas intersecciones que lo señalado para la Figura 4.2, destacando el punto C como el óptimo en el problema con los valores nominales de los parámetros. Cada escenario generado al variar a se describe a continuación:

- (a) $a = 1,25$: Corresponde al valor nominal del parámetro analizado. Para este caso, la solución óptima es la representada en el punto C , estando activa la restricción 4.4 y no activa la expresión 4.5.
- (b) $a = 1,36$: Al realizar un aumento del parámetro a hasta el valor de 1,36, C y D corresponden al mismo punto. De esta forma, la restricción 4.5 pasa a estar activa y el valor de la variable binaria y en el óptimo se mantiene.
- (c) $a = 2,56$: Si se continúa aumentando el valor del parámetro a hasta 2,56, ya no es posible que la solución mantenga el valor de la variable binaria (pasa a ser $y = 0$). Particularmente, el valor máximo que permite obtener una solución factible del problema corresponde a 2,56, encontrándose E y F en el mismo punto y las dos restricciones de interés pasan a estar activas. Valores superiores de a generarían que la solución fuese infactible.
- (d) $a = 0,25$: Considerando una disminución de a , se determina que el menor valor de a es 0,25 para que la solución considere el valor de la variable binaria obtenido en el caso original ($y = 1$). Debido a la forma de la función objetivo $z = 2x + y$, valores inferiores de a originarían que fuese preferible una solución con $y = 0$.

Para las variaciones en el parámetro a , los gráficos de la Figura 4.3 muestran que al alcanzar el límite superior del rango de estabilidad una restricción que no estaba activa pasa a estarlo. Esto coincide con lo ocurrido para el parámetro b . Sin embargo, para el límite inferior no es posible notar el mismo comportamiento, por lo que si se desea explotar esta noción debe ser analizada en mayor profundidad su aplicabilidad. Esto permite destacar que las eventuales mejoras a proponer no necesariamente funcionarán en la búsqueda de ambos límites para un mismo parámetro. Es decir, podrían existir mejoras que favorezcan la búsqueda de un solo límite y que para el otro se requiera implementar una idea diferente.

Restricciones del ejemplo 1 frente a cambios del parámetro a

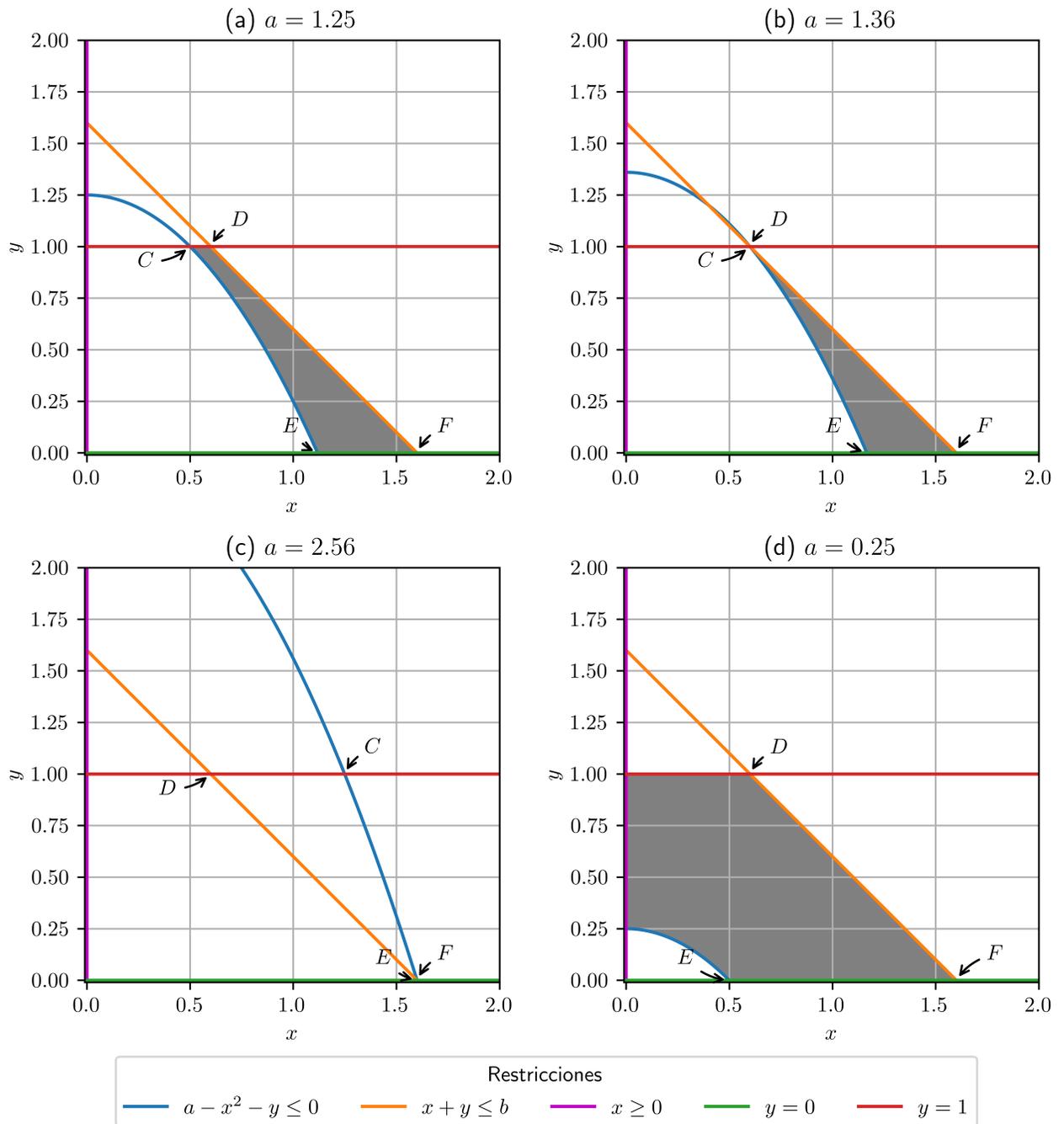


Figura 4.3: Restricciones del ejemplo 1 frente a cambios del parámetro a .

4.1.3. Reflexiones parciales

Entre los resultados obtenidos se destaca que el parámetro b no presenta un límite superior para los valores analizados, por lo que el algoritmo sugiere actualizar los parámetros de búsqueda, siendo importante analizar la necesidad de continuar investigando el límite. Para el presente ejemplo, al ser de pequeña escala, es posible observar que un aumento en el parámetro b no generará un cambio en la variable binaria, lo que podría ser dificultoso de inferir en problemas de mayor tamaño o con otra formulación de las restricciones. La particularidad y conocimiento del problema permite concluir que no existe un límite superior.

Desde otra perspectiva, la decisión de proseguir buscando el límite se relaciona con la necesidad de encontrar un límite o la consideración de que ya ha sido suficiente con descubrir que hasta un determinado valor no cambió la variable binaria. Este tipo de análisis se basa en la información y requerimientos del usuario, ya que es quien define la necesidad de estudiar un rango más amplio. Se tendrá presente esta reflexión en los siguientes casos a analizar. Para este primer ejemplo no se requiere continuar, pero en los posteriores se profundizará sobre los criterios de detención de la búsqueda, evidenciando la necesidad de alguna modificación al respecto.

Habiendo determinado las expresiones 4.2 y 4.3 mediante distintas resoluciones del problema, es relevante recordar el objetivo central de este trabajo y la siguiente pregunta a resolver: ¿Es posible obtener las ecuaciones 4.2 y 4.3 mediante un método más eficiente y que requiera un menor número de ejecuciones del problema de optimización?

En el algoritmo referencial, el inicio de las iteraciones se efectúa a partir del valor nominal de cada parámetro, razón por la que sería conveniente iniciar la búsqueda desde un valor más cercano a cada límite del rango de estabilidad y así requerir un menor número de iteraciones. La interpretación gráfica del efecto del cambio en los parámetros ha permitido notar que la modificación en la variable binaria se produce (en algunos casos) luego de que una restricción que no estaba activa pasa a estarlo. Entonces, ¿sería posible sistematizar la determinación del rango de estabilidad a partir de los cambios en las restricciones de desigualdad? De no ser posible calcular directamente los rangos de esta forma, ¿se podría aprovechar la información que otorga la estructura de las restricciones para generar una mejor estimación inicial de búsqueda del rango de estabilidad? ¿existirá la forma de generalizar esta idea para problemas de mayor escala y aplicar alguna metodología que permita estimar rápidamente valores que generen un cambio en las restricciones activas del problema? En el Capítulo 5 se presentarán los pasos necesarios para que se aplique esta idea en un nuevo algoritmo para determinar eficientemente el rango de estabilidad de parámetros, específicamente aportando a la definición de una mejor adivinanza inicial de búsqueda para alguno de los límites del rango.

Para reconocer otras mejoras a implementar en el Capítulo 5, se continuará analizando el desempeño del algoritmo referencial en otros ejemplos con diferentes características que permitan obtener mayor información para la posterior comparación de algoritmos. En particular, a continuación se calculará el rango de estabilidad para el segundo ejemplo escogido, que presenta un mayor número de variables binarias y restricciones que el primer ejemplo analizado.

4.2. Ejemplo ilustrativo 2

El segundo ejemplo a estudiar, y el primero asociado a redes de agua, corresponde a uno inspirado por los casos de síntesis de sistemas integrados de agua de Karuppiah y Grossmann [72]. Para este caso se consideran dos variables binarias, asociadas a la decisión de instalación de plantas de tratamiento. También presenta un mayor número de restricciones que el primer ejemplo, por lo que se espera que sea más exigente en costo computacional la implementación del algoritmo referencial. Este ejemplo permitirá, adicionalmente, explicitar los supuestos presentes en el contexto de problemas de redes de agua y ejemplificar cómo se relacionan con el algoritmo y el proceso de determinación del rango de estabilidad.

Respecto a la posible mejora a detectar en este ejemplo, las iteraciones que se realizarán permitirán notar que es conveniente utilizar información previa de los parámetros para enfocar el rango de búsqueda. Particularmente, se contrastarán los resultados matemáticos con la interpretación física o lógica de estos y así establecer la búsqueda dentro de valores que sean de interés por su representación en la realidad.

A grandes rasgos, el problema consta de dos plantas de procesos, cuya demanda total es de 90 [t/h] de agua, correspondiente al caso base. Para suplirla es posible obtener agua desde una fuente natural, flujos de agua recirculada tras el paso por plantas de tratamiento o desde otras plantas de procesos. El límite de batería en este caso considera desde el ingreso de agua proveniente de una fuente natural hasta que se envía a un sumidero post tratamiento.

Para este problema se asumirá que la demanda de agua de las plantas es constante y con un valor determinado, mismo supuesto que se considerará para el costo de instalación de las plantas. La superestructura y los elementos principales de la formulación del modelo se analizarán a continuación, mientras que el modelo completo se encuentra en el Anexo D.1.

4.2.1. Planteamiento

4.2.1.1. Superestructura

Los participantes presentes en el problema son los siguientes:

- Fuente de agua natural (N)
- Plantas de procesos (P_1 y P_2)
- Plantas de tratamiento de agua (T_1 y T_2)
- Sumidero (S)

Estos elementos se encuentran conectados a través de una red representada por nodos y arcos, cuya superestructura se observa en la Figura 4.4. Los colores de los flujos están asociados al nodo del que provienen y se asumirá que cada color representa una concentración diferente.

Para el planteamiento de este problema y su correspondiente superestructura se está suponiendo que las plantas de procesos pueden compartir flujos, además de obtener el agua proveniente de la fuente natural o de plantas de tratamiento. Adicionalmente, se considera

que las plantas de tratamiento no procesan agua obtenida desde otra planta de tratamiento.

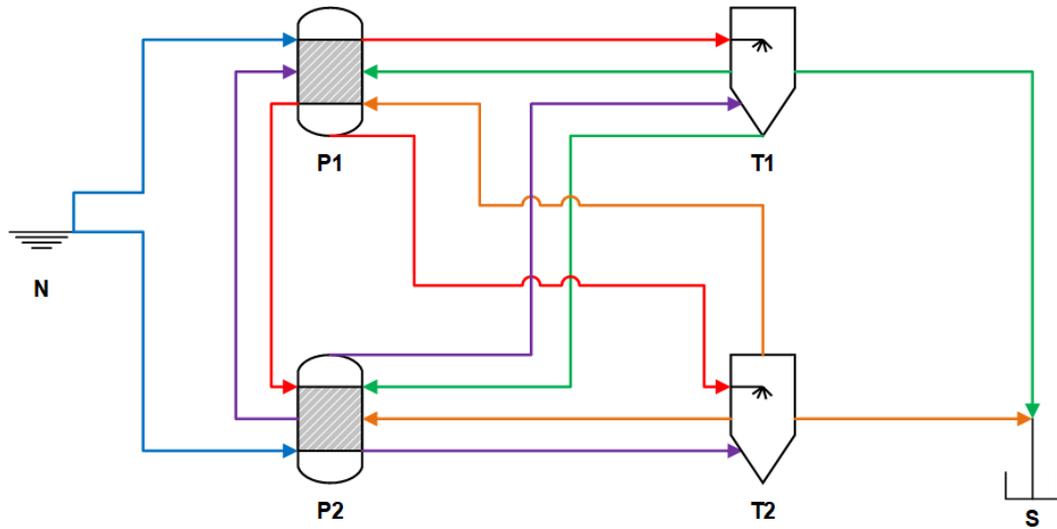


Figura 4.4: Superestructura del ejemplo ilustrativo 2.

Como se ha mencionado anteriormente, el objetivo principal de este trabajo corresponde al análisis de supuestos que presentan efectos en los parámetros del problema, cuya interpretación matemática tiene relación con el rango de estabilidad. El listado completo de supuestos de este problema se encuentran en la Tabla D.2 del Anexo D.1, destacando el elemento del problema con el que se relacionan. Aquellos que afectan a parámetros del problema son considerados de interés para el presente estudio, mientras que los otros se explicitan y están disponibles para discutirlos si fuese pertinente, considerando que una de las motivaciones de este trabajo tiene relación con evidenciar la presencia de supuestos. A continuación, se analizan los supuestos relacionados con parámetros del problema:

Demanda de agua constante y conocida

Considerar que la demanda de agua de las plantas de procesos no varía podría generar resultados alejados de la realidad, ya que existen distintos factores por los que podría fluctuar el consumo que tienen estas plantas. Entre estos están los requerimientos de agua fluctuantes a partir de cambios en la demanda del producto principal de la planta o cambios en los requerimientos de agua de refrigeración a partir de las variaciones de temperatura externas.

Costo de instalación constante y conocido

A diferencia de la demanda de agua, el costo de instalación podría ser constante en el corto y mediano plazo, ya que los elementos necesarios para realizarla podrían tener variaciones pequeñas o mantenerse sin cambios en su precio. Sin embargo, la estimación del costo de instalación de una planta de tratamiento podría presentar errores o no alcanzar la suficiente precisión, debido a los múltiples factores, decisiones y elementos involucrados. Adicionalmente, la magnitud de los costos de instalación puede generar que fluctuaciones en el valor escogido generen un gran impacto en la solución.

Flujo de contaminante es constante

Al igual que la demanda de agua, el flujo de contaminante podría variar. Este podría estar

asociado a aquella demanda, ya que se relacionan con la producción de la planta de procesos. Sin embargo, la diferencia entre ambos está en el orden de magnitud, ya que los flujos de contaminantes son inferiores en cuanto a órdenes de magnitud respecto a la demanda de agua, por lo que una variación en este parámetro podría tener un efecto inferior.

Considerando que este ejemplo es ilustrativo, se escogerán dos parámetros a estudiar, ya que el procedimiento para el tercero (o más parámetros si correspondieran) sería similar. Particularmente, se escogen la demanda de agua y el costo de instalación por el efecto que podrían generar sus cambios, en contraste con el menor orden de magnitud y eventual efecto de la no variación del flujo de contaminante. Esta decisión se sustenta en que, entre las diversas fuentes de incertidumbre presentes en los problemas de ingeniería, la demanda y los costos se han destacado como parte de las más estudiadas [94].

4.2.1.2. Conjuntos

- $I = \{P_1, P_2\}$: Conjunto de plantas de procesos.
- $J = \{T_1, T_2\}$: Conjunto de plantas de tratamiento.
- $K = \{N, P_1, P_2, T_1, T_2\}$: Conjunto de nodos que poseen flujos de salida.
- $L = \{P_1, P_2, T_1, T_2, S\}$: Conjunto de nodos que poseen flujos de entrada.

4.2.1.3. Parámetros

A continuación, se listan los parámetros asociados a los supuestos mencionados, mientras que la lista completa y sus valores nominales se encuentran en el Anexo D.1. Se destaca que los valores registrados son nominales (asignados en el problema original), debido a que algunos de estos serán modificados para determinar su rango de estabilidad.

- D_i : Demanda de agua de la planta de procesos $i \in I$.
- $Cost_j$: Costo de instalación de la planta de tratamiento $j \in J$.

4.2.1.4. Variables

- $F_{k,l}$: Flujo proveniente del nodo $k \in K$ que ingresa al nodo $l \in L$, con $k \neq l$.
- C_k : Concentración de salida del nodo $k \in K \setminus \{N\}$.
- y_j : Variable binaria asociada a la existencia ($y_j = 1$) o ausencia ($y_j = 0$) de la planta de tratamiento $j \in J$.

4.2.1.5. Restricciones

Las restricciones planteadas en este problema se pueden clasificar de la siguiente forma:

Balances de masa

Los balances de masa considerados serán el balance global del proceso, los balances por planta y por especies. Debido a la diferencia en orden de magnitud de los flujos másicos y las concentraciones de contaminante [*ppm*], se asume que el efecto del flujo de contaminantes no interfiere en los balances globales (del proceso y por planta) [72].

Operacionales

Se debe cumplir con la demanda de agua de cada planta de procesos, estableciéndose una concentración máxima de contaminante. Se incluye una restricción asociada a la dependencia de la existencia de flujos de plantas que se han instalado, es decir, no pueden existir flujos provenientes o dirigidos a plantas que no forman parte de la solución del problema. Adicionalmente, se establece una capacidad máxima para las plantas de tratamiento.

4.2.1.6. Funciones objetivo

La función objetivo del problema considera dos objetivos, uno de enfoque ambiental y otro económico, incluidos mediante la formulación de sumas ponderadas. El primer objetivo se basa en la minimización del uso de agua proveniente de la fuente natural (N), mientras que el segundo se enfoca en la minimización del costo de instalación de las plantas de tratamiento (T_1 y T_2).

La solución del modelo descrito se destaca a continuación:

4.2.2. Resultados del modelo de optimización

Resolviendo el problema de optimización con los valores nominales de los parámetros se determina que las variables binarias, consideradas como las de mayor interés para el diseño óptimo de la red de agua, adoptaron los valores de $y_{T_1} = 1$ e $y_{T_2} = 0$. La solución completa, incluyendo el resto de las variables, se encuentra disponible en el Anexo D.2. A partir de la información anterior, solamente se instala la planta de tratamiento T_1 y se observa también que esta solo envía agua tratada a P_1 . La solución obtenida se encuentra representada gráficamente en la Figura 4.5.

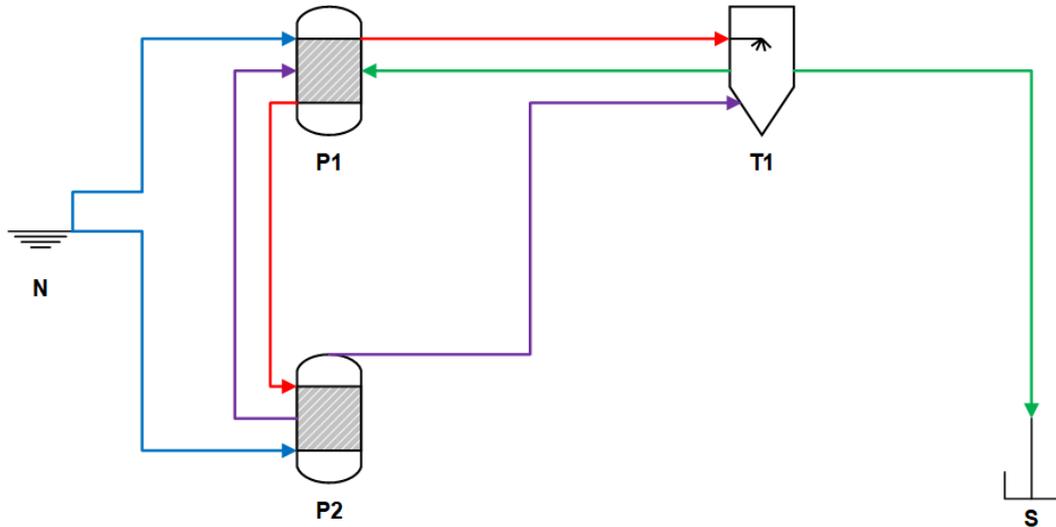


Figura 4.5: Red óptima del ejemplo ilustrativo 2.

La pregunta central para este problema es: ¿cuál es el rango de valores que pueden adoptar los parámetros (D_{P_1} , D_{P_2} , $Cost_{T_1}$ y $Cost_{T_2}$) para que los valores de y_{T_1} e y_{T_2} se mantengan? La respuesta a esta interrogante se resolverá en la siguiente sección.

4.2.3. Rango de estabilidad

Utilizando la estrategia propuesta por Yi y Lu [31] se obtuvieron los resultados que se encuentran en el Anexo D.3, destacándose las distintas iteraciones realizadas y el valor de las variables en los extremos de los rangos de estabilidad. Los principales resultados se resumen en la Tabla 4.3, incluyendo el valor nominal (original) del parámetro, el rango de estabilidad y el tiempo requerido para determinarlo. Estos resultados son analizados a continuación.

Tabla 4.3: Rango de estabilidad de los parámetros del ejemplo ilustrativo 2, obtenidos mediante el algoritmo referencial.

Parámetro [UN]	Nominal	Rango de estabilidad	Tiempo [s]
D_{P_1} [t/h]	50	$0 \leq D_{P_1} \leq 57,6$	30,34
D_{P_2} [t/h]	40	$0 \leq D_{P_2} \leq 81,9$	33,56
$Cost_{T_1}$ [USD\$]	1.000.000	$-1.100.000 \leq Cost_{T_1} \leq 1.018.000$	30,01
$Cost_{T_2}$ [USD\$]	1.100.000	$1.081.000 \leq Cost_{T_2}$	36,36

Resultados obtenidos al modificar D_{P_1}

Se determinó que el menor valor de D_{P_1} para que las variables se mantengan es cero. Particularmente, las iteraciones evaluadas para valores negativos ocasionaban que la solución fuese infactible, deteniéndose la búsqueda por alcanzar la tolerancia establecida en $D_{P_1} = 0$. Respecto al límite superior, la detención se produjo por la misma razón, esta vez en el valor 57,6. En cuanto a la interpretación gráfica, esta se encuentra en el Anexo D.3, notándose que el comportamiento de las variables binarias concuerda con lo esperado.

Resultados obtenidos al modificar D_{P_2}

Tal como ocurrió para el parámetro D_{P_1} , la búsqueda del límite inferior de D_{P_2} evaluó valores negativos y finalizó en cero por alcanzar la tolerancia definida. En cuanto al límite superior, la búsqueda se detiene en 81,9 y el algoritmo sugiere actualizar los parámetros de búsqueda para proseguir su implementación. Las iteraciones efectuadas permiten concluir que la solución se encuentra entre 81,9 y 82,0, y la continuación de las iteraciones hubiese requerido pasos de magnitud inferior a la tolerancia definida, por lo que no se efectúan. Se destaca que la representación gráfica en el Anexo D.3 muestra un comportamiento como el esperado.

Resultados obtenidos al modificar $Cost_{T_1}$

A diferencia de los parámetros asociados a la demanda (D_{P_1} y D_{P_2}), las evaluaciones de valores negativos de $Cost_{T_1}$ no producen infactibilidades, si no que se generan cambios en las variables binarias para los casos en que $Cost_{T_1}$ sea inferior a $-1.100.000$. Considerando el contexto del problema, aquel parámetro no puede ser negativo. En ese sentido, el rango que representa una interpretación real sería el expresado ecuación 4.6, donde se señala también el límite superior, que se ha determinado tras alcanzar la tolerancia establecida.

$$0 \leq Cost_{T_1} \leq 1.018.000 \quad (4.6)$$

El resultado para el límite superior se encuentra dentro de lo esperado, considerando que debería estar entre el valor original (USD\$1.000.000) y el costo de instalación de la planta T_2 (USD\$1.100.000). Si fuese más costoso que T_2 , se optaría por instalar aquella planta. La

representación visual concuerda con lo esperado, según se aprecia en el Anexo D.3.

Resultados obtenidos al modificar $Cost_{T_2}$

La búsqueda del límite inferior del rango de estabilidad de $Cost_{T_2}$ finalizó en 1.081.000, donde el algoritmo sugería actualizar los parámetros de búsqueda. Sin embargo, ya se conocía por las iteraciones previas que el caso $Cost_{T_2} = 1.080.000$ genera un cambio en las variables binarias, por lo que el límite se encuentra entre 1.080.000 y 1.081.000. No es necesario continuar la búsqueda porque una actualización del paso para iterar entre estos valores sería inferior a la tolerancia establecida. Tal como podría esperarse, al ser la planta de mayor costo, su precio puede disminuir hasta un valor intermedio entre su costo original ($USD\$1.100.000$) y $Cost_{T_1}$ ($USD\$1.000.000$), sin modificar el valor original de las variables binarias. El resto de variables del problema tampoco cambian.

Respecto al límite superior, las iteraciones efectuadas finalizan en 10.100.000 por alcanzar el número máximo de iteraciones sin detectar cambios, donde el algoritmo sugiere actualizar los parámetros de búsqueda para continuar. Sin embargo, la estructura de las restricciones sugiere que el rango no está acotado superiormente. Que no esté acotado superiormente el rango de estabilidad se debe a que, como $Cost_{T_2} > Cost_{T_1}$, al decidir instalar la planta T_1 en el escenario original, un aumento en $Cost_{T_2}$ no cambiará la opción, ya que en ese caso la opción de instalar T_1 continúa presentando un mayor incentivo. El comportamiento descrito se observa en el Anexo D.3, concordando con lo esperado.

4.2.4. Configuraciones óptimas de la red en los extremos del rango de estabilidad

Hasta el momento se ha destacado que la decisión de instalación de plantas se representa mediante variables binarias. Sin embargo, ¿qué sucede con la instalación de las conexiones? A continuación, se presentan las configuraciones obtenidas en los extremos del rango de estabilidad de los parámetros analizados en este problema, enfatizando en cómo cambian los flujos de la red, pese a que las variables binarias (instalaciones de planta) se mantienen:

Al aumentar D_{P_1} se obtiene la misma configuración de la red que en el caso óptimo original, ilustrada en la Figura 4.5. Por el contrario, la disminución de D_{P_1} hasta un valor nulo genera un cambio en la red, eliminándose los flujos desde y hacia P_1 , y apareciendo F_{T_1, P_2} , según se observa en la Figura 4.6.



Figura 4.6: Red óptima al disminuir D_{P_1} en el ejemplo ilustrativo 2.

Para el caso de D_{P_2} , al aumentar su valor se presenta la siguiente diferencia respecto al resultado original: no existe un flujo desde la planta T_1 a P_1 (F_{T_1, P_1}), pero sí desde aquella planta de tratamiento a P_2 (F_{T_1, P_2}), tal como se aprecia en la Figura 4.7.

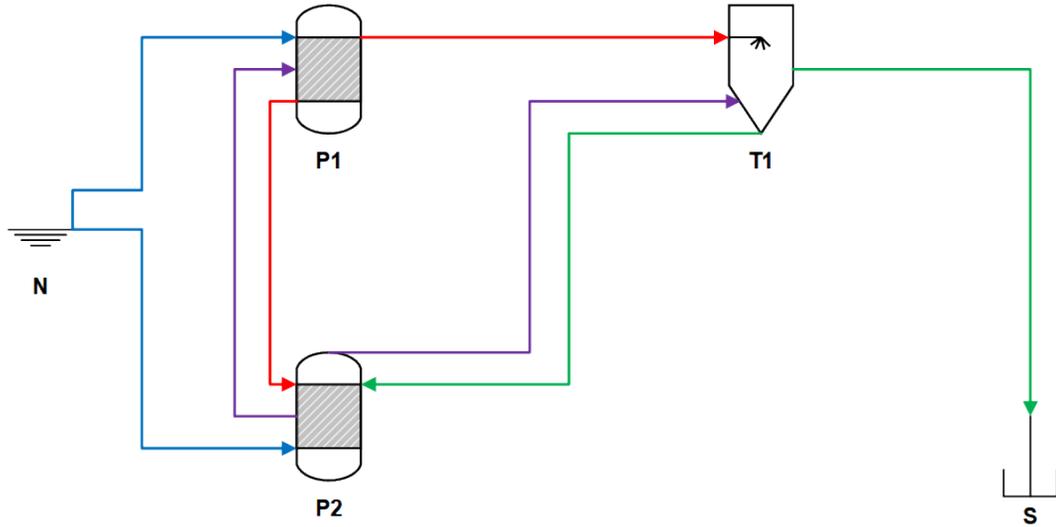


Figura 4.7: Red óptima al aumentar D_{P_2} en el ejemplo ilustrativo 2.

La disminución de D_{P_2} origina un efecto similar a la disminución de D_{P_1} , ya que se eliminan las conexiones desde y hacia P_2 al tener una demanda nula. Esta es la única diferencia con el caso original, según se observa en la Figura 4.8:

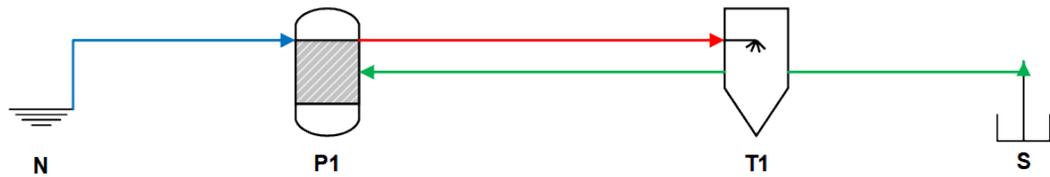


Figura 4.8: Red óptima al disminuir D_{P_2} en el ejemplo ilustrativo 2.

En cuanto a las variaciones de $Cost_{T_1}$ y $Cost_{T_2}$, la red óptima obtenida al evaluar los extremos del rango de estabilidad de ambos parámetros posee las mismas conexiones y valores de los flujos que en el caso original, tal como se determinó en el Anexo D.3. Estos resultados se pueden relacionar con que solo se está considerando el costo de instalación de plantas. En caso de que se analizaran los costos de operación, que podrían relacionarse con los flujos, sería posible que se modifiquen las conexiones de la red.

Previamente se ha mencionado la importancia y la duración que tienen algunas decisiones sobre otras, como ocurre con la comparación entre estratégicas y operacionales. Para este segundo ejemplo ilustrativo se observa que, pese a mantener los valores de las variables binarias, en algunos casos se modifican las conexiones entre los nodos de la red, por lo que podría ser necesario distinguir dos categorías de decisiones de instalación: instalación de plantas y de conexiones en la red.

Pese a que inicialmente el trabajo se centra en la decisión de instalación de plantas, se considera que para la estrategia a desarrollar es conveniente establecer previamente qué se está decidiendo y qué se desea mantener, considerando la interpretación del rango de estabilidad. Esta generalización se tendrá presente al crear el algoritmo en el Capítulo 5. También serán consideradas otras ideas para la elaboración del algoritmo propuesto, provenientes de las reflexiones señaladas a continuación.

4.2.5. Reflexiones parciales

Se aplicó el algoritmo de Yi y Lu [31] para un problema de optimización de redes de agua de pequeña escala, logrando calcular el rango de estabilidad para los parámetros en estudio. Como se esperaba, los tiempos requeridos por cada parámetro fueron superiores a los del ejemplo 1, ya que presenta un mayor número de variables y restricciones. La resolución del ejemplo 2 permitió reconocer las siguientes posibilidades de mejora:

Los parámetros estudiados correspondieron a demandas de agua (D_{P_1} y D_{P_2}) y costos de instalación ($Cost_{T_1}$ y $Cost_{T_2}$), los que no pueden ser negativos. Sin embargo, el algoritmo evaluó valores inferiores a cero para D_{P_1} , D_{P_2} y $Cost_{T_1}$. En los primeros dos casos la solución era infactible para valores inferiores a cero, por lo que no formaron parte del rango de estabilidad. Sin embargo, se podría disminuir el tiempo de ejecución si se descartan desde un comienzo y no se realizaran aquellas evaluaciones. En cuando a $Cost_{T_1}$, el algoritmo determinó un límite inferior negativo. Ante los resultados mencionados, se propone que el nuevo algoritmo a desarrollar establezca cotas relacionadas con la interpretación en la realidad que tienen los parámetros en estudio, considerando que la herramienta a proponer será un apoyo a la toma de decisiones en contextos reales de redes de agua.

Existieron también casos en que el algoritmo sugería actualizar los parámetros de búsqueda por alcanzar el número máximo de iteraciones para un mismo paso ($i < 10$), pese a que ya se conocía información sobre valores más alejados (generaban un cambio en las variables binarias). En estos casos sería conveniente que se aproveche aquella información y se automatice la búsqueda del límite para que el criterio de detención sea el haber alcanzado la tolerancia definida previamente, sin la necesidad de una interacción adicional del usuario para agilizar el proceso. Por otro lado, se destaca la conveniencia de requerir una actualización externa en los casos en que no se ha determinado una infactibilidad o cambio en las variables binarias al alcanzar un determinado número de iteraciones, ya que sin esa información el algoritmo hubiese continuado la búsqueda del límite superior de $Cost_{T_2}$ sin detenerse, debido a que no detectaría cambios para valores superiores a los evaluados.

En cuanto al tipo de variables analizadas, se reconoció que dentro del rango de estabilidad las instalaciones de plantas se mantienen, aunque las conexiones pueden sufrir modificaciones. Esto se debe a que los flujos que unen plantas no están siendo representados por variables binarias, correspondientes a las que el algoritmo analiza. Esta modificación corresponde a un cambio en el layout, por lo que se considera útil establecer al comienzo de la ejecución del algoritmo qué variables se desea mantener, generalizando el proceso para aquellos escenarios en que algunas instalaciones se consideren con mayor relevancia que otras.

Las ideas que permitió reconocer el ejemplo ilustrativo 2 serán incluidas en el nuevo algoritmo a desarrollar en el Capítulo 5. Para complementarlas, en la siguiente sección se estudiará el ejemplo ilustrativo 3, cuyas variables de interés representan la instalación de conexiones entre nodos.

4.3. Ejemplo ilustrativo 3

El tercer ejemplo corresponde a la optimización de una red de agua con tres unidades de tratamiento y dos de procesos, cuya formulación MINLP fue planteada por Ahmetovic y Grossmann [73]. Esta red posee múltiples contaminantes y presenta la posibilidad de que sus flujos sean tratados con anterioridad al ingreso a las unidades de proceso, existiendo también un tercer tratamiento para su reutilización o disposición al ambiente. Las variables binarias de este ejemplo representan la instalación de las conexiones entre los nodos de la red, difiriendo con lo analizado hasta el momento (instalación de plantas).

Este problema es de una escala mayor a los dos que se han analizado previamente, esperándose que se requiera un mayor tiempo para determinar el rango de estabilidad de cada parámetro. Precisamente, una posible mejora detectada con este ejemplo es la inclusión de un tiempo límite por iteración, para así obtener el rango de estabilidad en un período de tiempo eficiente. La evidencia de esta situación se realizará a partir de los resultados obtenidos, siendo discutidos al finalizar el análisis de este ejemplo.

El modelo completo se encuentra planteado en el Anexo E.1, presentando diversas consideraciones, siendo destacadas las principales en las siguientes secciones. Entre los parámetros utilizados, serán estudiados en este documento el flujo demandado por cada unidad de proceso, el porcentaje de remoción de una de las plantas de tratamiento (las otras dos no consideran porcentajes, si no que tienen una concentración final definida) y el costo de instalación de las tres unidades de tratamiento.

4.3.1. Planteamiento

4.3.1.1. Superestructura

Los nodos que componen la red son los siguientes:

- Fuente de agua (N)
- Unidades de procesos (P_1 y P_2)
- Unidades de tratamiento (T_1 , T_2 y T_3)
- Mezcladores de flujos de las unidades de procesos (MPU_1 y MPU_2)
- Mezcladores de flujos de las unidades de tratamiento (MTU_1 , MTU_2 y MTU_3)
- Mezclador de flujo final para la descarga al sumidero (MF)
- Separadores de flujos de las unidades de proceso (SPU_1 y SPU_2)
- Separadores de flujos de las unidades de tratamiento (STU_1 , STU_2 y STU_3)
- Separador de flujo de la fuente inicial de agua (SI)
- Sumidero (S)

A partir de las diversas opciones mencionadas por Ahmetovic y Grossmann [73], se ha considerado para este problema que es posible compartir flujos entre procesos o enviarlos a alguna de las etapas de tratamiento (T_2 y T_3), descartándose que luego de los procesos se conecten con el pretratamiento T_1 . También se ha eliminado la opción de reciclos locales de cada nodo. Con ello, las posibles conexiones quedan representadas en la superestructura de la Figura 4.9. En esta imagen las flechas de color negro corresponden a instalaciones realizadas por defecto, considerando la relación entre mezcladores y separadores con cada unidad de proceso, de tratamiento, la fuente y el sumidero. Las flechas de los demás colores son las decisiones de instalación del problema, siendo incluidas de diferente color según el separador de flujo de origen para facilitar su visualización.

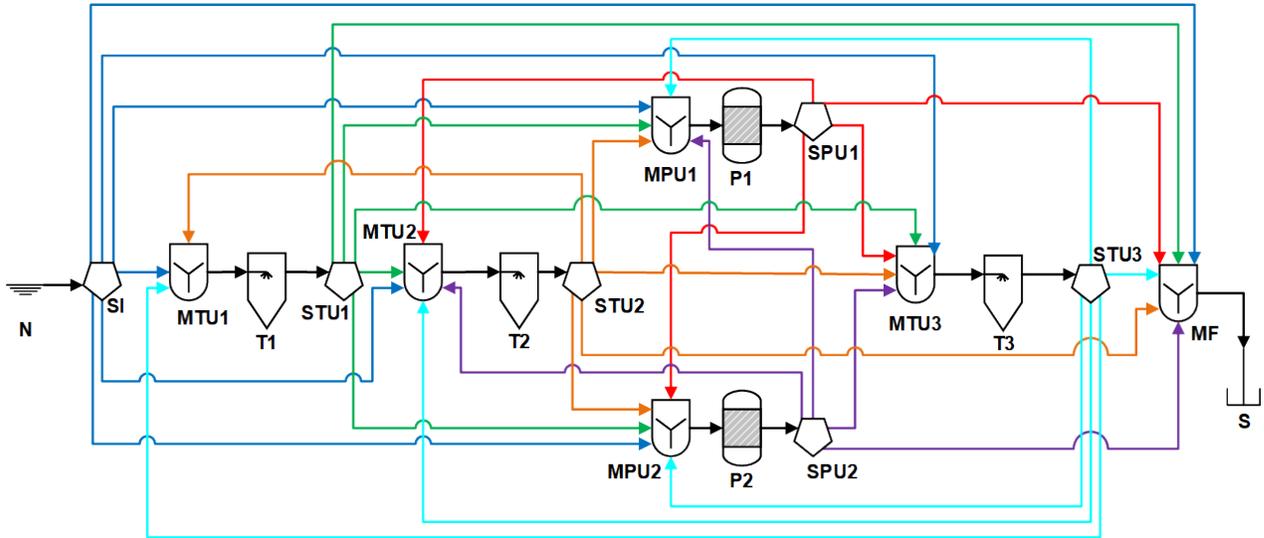


Figura 4.9: Superestructura del ejemplo ilustrativo 3.

4.3.1.2. Conjuntos

- $J = \{A, B, C, D\}$: Conjunto de contaminantes.
- $PU = \{P_1, P_2\}$: Conjunto de unidades de procesos.
- $TU = \{T_1, T_2, T_3\}$: Conjunto de unidades de tratamiento.

4.3.1.3. Parámetros

Los diversos parámetros presentes en el problema se encuentran en el Anexo E.1, destacándose a continuación los principales para el análisis del rango de estabilidad:

- FPU_p^{in} : Flujo de entrada al proceso $p \in PU$.
- r_{T_3} : Porcentaje de remoción de contaminantes en la unidad de tratamiento $T_3 \in TU$.
- IC_t : Coeficiente de costo de inversión de la unidad de tratamiento $t \in TU$.

4.3.1.4. Variables

Las variables del problema se dividen en continuas y binarias. Por un lado, los flujos y concentraciones corresponden a variables continuas y positivas, mientras que las binarias representan la existencia de conexiones entre los nodos de la red. A continuación, se destacan las variables que se buscará mantener su valor original en el cálculo del rango de estabilidad:

- y_{FIF} : Existencia de conexión entre la fuente y el mezclador de flujo final.
- y_{FIP_p} : Existencia de conexión entre la fuente y la unidad de proceso $p \in PU$.
- y_{FIT_t} : Existencia de conexión entre la fuente y la unidad de tratamiento $t \in TU$.
- $y_{FP_{p,p'}}$: Existencia de conexión entre el proceso $p \in PU$ y el proceso $p' \in PU$.
- y_{FPO_p} : Existencia de conexión entre el proceso $p \in PU$ y el mezclador de flujo final.
- $y_{FPT_{p,t}}$: Existencia de conexión entre el proceso $p \in PU$ y el tratamiento $t \in TU$.
- $y_{FT_{t,t'}}$: Existencia de conexión entre el tratamiento $t \in TU$ y el tratamiento $t' \in TU$.
- y_{FTO_t} : Existencia de conexión entre el tratamiento $t \in TU$ y el mezclador de flujo final.
- $y_{FTP_{t,p}}$: Existencia de conexión entre el tratamiento $t \in TU$ y el proceso $p \in PU$.

4.3.1.5. Restricciones

En términos generales, las restricciones de este problema son descritas a continuación, mientras que sus expresiones completas se encuentran en el Anexo E.1:

Balances de masa

Los balances de masa se plantean en estado estacionario. Entre estos, se ha incluido el balance global según lo recomendado por Karuppiah y Grossmann [72]. Se añaden también los balances por equipos y por especies. Debido a los órdenes de magnitud de los contaminantes [ppm], se ha desestimado su efecto en los balances globales de la red y los equipos.

Operacionales

Principalmente, las restricciones impuestas para el funcionamiento de las unidades de procesos y tratamientos se enfocan en la calidad de entrada de sus flujos. Además, existe una limitación sobre la concentración permitida para la descarga al sumidero (ambiente).

Lógicas

Físicamente, para transportar agua es necesaria una tubería que conecte los nodos, por lo que se añadieron expresiones que relacionen las variables binarias con los flujos. Junto con ello, Ahmetovic y Grossmann [73] propusieron la inclusión de restricciones basadas en las características físicas de la red y relaciones lógicas, permitiendo que la búsqueda del óptimo se realice en una región más acotada, acelerando la resolución del problema.

4.3.1.6. Función objetivo

Este problema es monoobjetivo y busca minimizar el costo anualizado de la red, basándose en el costo de agua de la fuente, la instalación y operación de las unidades de tratamiento, la inversión en instalación de tuberías y el costo del bombeo de agua en estas.

4.3.1.7. Ecuaciones adicionales

Una de las novedades de la investigación de Ahmetovic y Grossmann [73] es la inclusión de expresiones lógicas y provenientes de las características físicas del problema, definiendo cotas para las variables del problema, las que se encuentran en el Anexo E.1, enfatizando en que sus resultados son incorporados como parámetros del problema de optimización. Es decir, corresponde a información que se calcula previamente a la ejecución del problema.

4.3.2. Resultados del modelo de optimización

El resultado de la optimización del modelo planteado al emplear los valores nominales de los parámetros se representa gráficamente en la Figura 4.10, mientras que la solución completa se puede consultar en el Anexo E.2. Se destaca que, al asumir que la concentración de contaminante en los flujos de salida de cada separador es la misma, los colores de las conexiones en la Figura 4.10 representan flujos con una misma concentración.

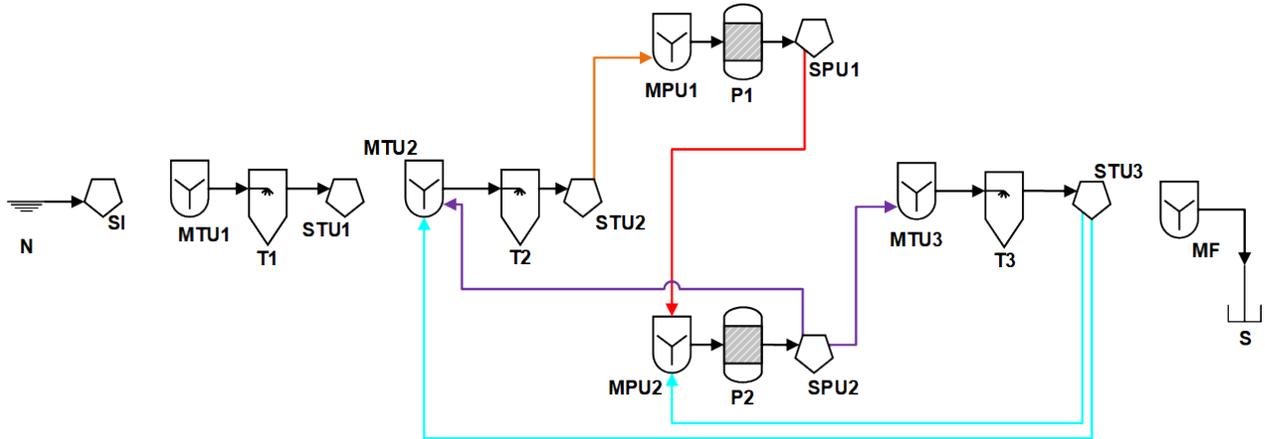


Figura 4.10: Red óptima del ejemplo ilustrativo 3.

En la red óptima se ha decidido instalar tuberías que enlazan las unidades de procesos P_1 y P_2 , y los tratamientos T_2 y T_3 . No existen conexiones con T_1 , ni tampoco se extrae agua de la fuente o se dispone en el sumidero. Entonces, las variables binarias cuyo valor óptimo es igual a 1 son las siguientes: $y_{FP_{P_1,P_2}}$, $y_{FPT_{P_2,T_2}}$, $y_{FPT_{P_2,T_3}}$, $y_{FT_{T_3,T_2}}$, $y_{FTP_{T_2,P_1}}$ e $y_{FTP_{T_3,P_2}}$, mientras que el valor óptimo de las demás variables binarias es cero. Estas conexiones y los caudales obtenidos son aproximadamente iguales a los reportados por Ahmetovic y Grossmann [73] (la función objetivo difiere en un 0,16 %, por la tolerancia ϵ explicada en el Anexo E.1).

Tal como en los ejemplos ilustrativos anteriores, la pregunta a responder es: ¿cuál es el rango de valores de los parámetros ($FPU_{P_1}^{in}$, $FPU_{P_2}^{in}$, r_{T_3} , IC_{T_1} , IC_{T_2} e IC_{T_3}) para que las variables binarias mantengan su valor? Esta pregunta se responde a continuación.

4.3.3. Rango de estabilidad

El algoritmo de Yi y Lu [31] es aplicado en este ejemplo, obteniendo los resultados del Anexo E.3, que presenta las diversas iteraciones realizadas para determinar los límites del rango de estabilidad de los parámetros en estudio, mientras que los rangos obtenidos y el tiempo requerido para cada caso se resumen en la Tabla 4.4. Se destaca que se han considerado tres parámetros de la categoría operacional y tres económicos. Particularmente, los parámetros operacionales afectan a las cotas adicionales planteadas, según se expresa en el Anexo E.1. Debido a esto, al modificarse el valor de los mencionados parámetros, las ecuaciones en que están involucrados se resuelven nuevamente y se obtienen otras cotas para el problema, ingresadas como parámetros en cada iteración.

En términos generales, de la Tabla 4.4 se observa que el tiempo requerido para calcular el rango de estabilidad aumentó en dos órdenes de magnitud en relación al ejemplo 2, pese

Tabla 4.4: Rango de estabilidad de los parámetros del ejemplo ilustrativo 3, obtenidos mediante el algoritmo referencial.

Parámetro [UN]	Nominal	Rango de estabilidad	Tiempo [s]
$FPU_{P_1}^{in}$ [t/h]	40	$11,5 \leq FPU_{P_1}^{in} \leq 49,0$	7.259,02
$FPU_{P_2}^{in}$ [t/h]	50	$41,0 \leq FPU_{P_2}^{in} \leq 55,3$	1.783,94
r_{T_3} [%]	95	$80,8 \leq r_{T_3} \leq 97,4$	146,06
IC_{T_1} [] ^a	10.000	$4.150 \leq IC_{T_1} \leq 100.000$	3.173,24
IC_{T_2} [] ^a	25.000	$-65.000 \leq IC_{T_2} \leq 115.000$	1.039,41
IC_{T_3} [] ^a	30.000	$-60.000 \leq IC_{T_3} \leq 120.000$	32,02

^a El costo anualizado de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [)]. Ver Anexo E.1

a que el tiempo necesario en el caso original del ejemplo 3 es aproximadamente el doble del anterior (ejemplo 2: 0,90[s] y ejemplo 3: 1,71[s]). La razón principal es la complejidad del tercer ejemplo, existiendo valores que se evaluaron en determinadas iteraciones y que la solución no fue encontrada hasta alcanzar el tiempo límite por defecto de GAMS (1.000[s]), obteniendo soluciones en que se interrumpió la búsqueda, situación explicada en el Capítulo 3. A continuación, se discuten las particularidades de lo obtenido al analizar cada parámetro:

Resultados obtenidos al modificar $FPU_{P_1}^{in}$

Durante las múltiples resoluciones del problema de optimización para encontrar el límite inferior de $FPU_{P_1}^{in}$, el algoritmo alcanzó el tiempo límite incluido en su resolución en GAMS (1.000 [s]) para cinco casos, según se detalla en el Anexo E.3. El valor determinado fue de 11,5 y se obtuvo tras alcanzar la tolerancia establecida. Por otro lado, en la búsqueda del límite superior el algoritmo se detuvo en el valor de 49,0, ya que alcanzó el límite de iteraciones para un mismo paso ($i < 10$). Sin embargo, previamente se evaluó el valor de 50,0, generándose un cambio en las variables binarias, por lo que el límite se encuentra efectivamente entre 49,0 y 50,0. Dado que la tolerancia no ha sido alcanzada, en este caso se podría automatizar la búsqueda y que el algoritmo continúe iterando entre ambos valores señalados. Por lo pronto, se ha escogido como límite el mencionado por el algoritmo, ya que solo es posible asegurar que no existen cambios en las variables de interés hasta el valor de 49,0.

Resultados obtenidos al modificar $FPU_{P_2}^{in}$

La búsqueda del límite inferior del rango de estabilidad de $FPU_{P_2}^{in}$ finalizó en el valor de 41,0 por alcanzar el número máximo de iteraciones para un mismo paso ($i < 10$), aunque para este valor no se produce un cambio en las variables binarias. Las iteraciones presentes en el Anexo E.3 permiten concluir que el límite se encontraría entre 40,0 y 41,0, por lo que es una oportunidad de mejora para alcanzar la precisión determinada por la tolerancia establecida. Específicamente, las evaluaciones graficadas en el Anexo E.3 corroboran la existencia del límite entre ambos valores. En contraste, la búsqueda del límite superior finaliza por alcanzar la tolerancia definida. Particularmente, para este límite el tiempo máximo de ejecución no fue alcanzado, lográndose iteraciones que requerían individualmente menos del triple de tiempo del caso original. El alto tiempo requerido para obtener el rango de estabilidad y que se señala en la Tabla 4.4 se debe a que en una de las evaluaciones necesarias en la búsqueda del límite inferior se alcanzó el tiempo máximo, mientras que otra se aproximó a aquel tiempo.

Resultados obtenidos al modificar r_{T_3}

El límite inferior de r_{T_3} es 80,8, determinado luego de alcanzar la tolerancia establecida, siendo el mismo criterio por el que finalizó la búsqueda del límite superior en 97,4. Se destaca que no fue alcanzado el tiempo límite, aunque se detectó otro aspecto por mejorar: Entre los valores analizados en la búsqueda del límite superior, el algoritmo evalúa el valor de 105,0 por el tamaño del paso inicial. Para este valor la solución es infactible, tal como se esperaba, ya que el porcentaje de remoción r_{T_3} debiese definirse entre 0 y 100. Esta carencia en cuanto a la interpretación del parámetro es un aspecto a mejorar y que se tendrá presente para el nuevo algoritmo a desarrollar.

Resultados obtenidos al modificar IC_{T_1}

El límite inferior de IC_{T_1} se obtuvo luego de alcanzar la tolerancia definida, presentándose un valor evaluado que originaba que se utilizara el tiempo máximo de resolución. Por el contrario, para el límite superior el algoritmo se detiene por el límite en el número de iteraciones por paso ($i < 10$), sugiriendo actualizar los parámetros de búsqueda. Considerando que no se ha detectado un cambio en las variables binarias hasta el valor de 100.000, es posible plantear la siguiente pregunta: ¿es necesario continuar buscando un valor que genere un cambio en las variables binarias? La no automatización del algoritmo ocasiona que el usuario deba plantearse esta interrogante previo a definir nuevos parámetros de búsqueda, siendo necesario evaluar la pertinencia de proseguir el proceso, basándose en si realmente es posible que el parámetro aumente a 10 veces su valor nominal y en la información del problema. Específicamente, en la solución original no hay conexiones asociadas a la unidad de tratamiento T_1 , por lo que un aumento en sus costos no genera un incentivo para que existan, si no que se espera lo contrario.

La reflexión sobre la pertinencia de continuar la búsqueda de los límites del rango de estabilidad cuando se tiene la intuición de que no existirá una cota o para aquellos casos en que se ha superado lo que se ha investigado como variación máxima, será parte del nuevo algoritmo a formular. Por el momento se establecerá como límite del rango de estabilidad el último valor ejecutado, para así no incorporar valores arbitrarios a evaluar y poder efectuar una comparación con los nuevos pasos que tendrá el algoritmo a proponer.

Resultados obtenidos al modificar IC_{T_2}

La determinación del límite inferior del rango de estabilidad se realizó hasta alcanzar el valor de -65.000 , sugiriéndose actualizar los parámetros de búsqueda. Al igual que para el parámetro r_{T_3} , el algoritmo evalúa valores fuera de lo esperado, ya que el costo de inversión no debiese ser negativo. Esta consideración será incluida en el nuevo algoritmo a desarrollar, facilitando la incorporación de información previa en el análisis correspondiente.

En cuanto al límite superior, la búsqueda concluye en el valor de 115.000, sugiriendo actualizar la tolerancia y el paso de búsqueda. Nuevamente es posible plantearse la necesidad de continuar buscando el límite para casos en que se ha sido evaluado un cambio significativo en los valores de los parámetros. Para este parámetro, se ha aumentado hasta un 460%. Según la información anterior, no es posible asegurar que no exista un cambio por sobre el valor de 115.000, pero sí la interpretación del parámetro sugiere que su menor valor es cero (no puede ser negativo), por lo que el rango de estabilidad se puede acotar al menos inferiormente y quedaría representado por la expresión 4.7:

$$0 \leq IC_{T_2} \leq 115.000 \quad (4.7)$$

A diferencia de IC_{T_1} , en este caso existen conexiones con la unidad de tratamiento T_2 , por lo que podría ocurrir que para un valor elevado de su costo de inversión se produzca un cambio en las variables, a menos que alguna configuración de restricciones ocasione que las conexiones existan independiente de su costo, lo que no se puede concluir en el rango analizado. De todas formas, en caso de que se conozca previamente que las conexiones se mantendrán pese al valor del parámetro, no sería necesario calcular el rango de estabilidad.

Resultados obtenidos al modificar IC_{T_3}

El cálculo del límite inferior del rango de estabilidad de IC_{T_3} se efectuó hasta el valor de -60.000 . Nuevamente la búsqueda debería finalizar sin disminuir a valores negativos, por lo que la inclusión de una cota adicional permitiría reducir el número de iteraciones necesarias. Para el resultado actual, la detención se produjo tras no detectar un cambio en las variables binarias y sugerir que se modifique el paso y la tolerancia definida. Al igual que para el límite inferior, en la búsqueda del superior tampoco se detectan cambios en las variables binarias, por lo que se alcanza a analizar hasta el valor de 120.000 y, posteriormente, el algoritmo sugiere actualizar el paso y tolerancia de búsqueda. En caso de que se estime que el valor del parámetro pueda aumentar por sobre el 400% , sería necesario continuar la búsqueda. A partir de los resultados anteriores, y según la interpretación del parámetro de costos, el rango de estabilidad se puede acotar a la expresión 4.8:

$$0 \leq IC_{T_3} \leq 120.000 \quad (4.8)$$

Se destaca que se ha acotado el límite inferior, pero para el superior se conoce que el valor de 120.000 mantiene la solución original, por lo que el límite efectivo se encuentra en un valor mayor, pero al menos hasta 120.000 es posible asegurar que no existen cambios en las variables binarias. Específicamente, y en contraste a los costos de T_1 y T_2 , las múltiples efectuadas en la representación gráfica del Anexo E.3 muestran que si se continuaba la búsqueda, se reconocería el límite efectivo del rango, lo que demuestra que no es posible asegurar en problemas complejos que el rango sea no acotado si es que no se ha alcanzado a notar un cambio en las variables solamente en los valores evaluados al emplear el algoritmo.

4.3.4. Configuraciones óptimas de la red en los extremos del rango de estabilidad

En este tercer ejemplo las variables binarias representan la existencia de conexiones entre nodos, razón por la que un cambio en estas ocasiona una modificación de la red. Es decir, dentro del rango de estabilidad la red óptima será la misma que en la solución original. Esto contrasta con lo observado en el ejemplo 2, cuyas variables binarias representan la instalación de plantas, por lo que mantener el valor de estas variables no asegura que la red sigue igual. Específicamente, en el ejemplo 2 se observaron diferentes configuraciones que permiten unir las plantas instaladas, dentro del rango de estabilidad de los parámetros analizados.

Notando que los cambios en los parámetros pueden ocasionar modificaciones en la red de agua, incluso dentro del rango de estabilidad, se observa la dependencia del algoritmo con la formulación del problema. Particularmente, con la elección de las decisiones representadas mediante variables binarias. Esta situación será analizada en mayor profundidad en las

reflexiones finales del presente capítulo, siendo introducida en esta sección para facilitar la comprensión de la diferencia entre los ejemplos 2 y 3. Específicamente, para explicar la ausencia de comparaciones en configuraciones en los extremos del rango de estabilidad en el ejemplo 3, ya que en todos los casos la red es igual a la original.

4.3.5. Reflexiones parciales

Complementando las reflexiones parciales de los ejemplos anteriores, el problema actual permite reconocer consideraciones relevantes, como es la interpretación lógica o física de los parámetros. Particularmente, en este ejemplo se evaluaron valores negativos para costos, aspecto ya discutido en el ejemplo 2 y que corresponde a una oportunidad de mejora para el desarrollo del nuevo algoritmo.

Otro de los aspectos a profundizar es la pertinencia de continuar la búsqueda de los límites del rango de estabilidad cuando se ha alcanzado el número máximo de iteraciones por paso ($i < 10$). Observando que en algunos casos del presente ejemplo se ha detectado cambios en las variables binarias en determinadas iteraciones, se logra acotar el rango de valores donde se encuentra el límite buscado. En aquellos casos sería conveniente continuar la búsqueda, en contraste con otros en que podría no existir el límite y se emplee un alto tiempo antes de finalizarla. El uso de información previa también podría favorecer la decisión de proseguir la búsqueda, ya que para valores que tengan una diferencia mayor a la que realmente pueda establecerse no sería necesaria su evaluación.

Como el presente ejemplo es de mayor tamaño y complejidad que los anteriores, el tiempo requerido para determinar el rango de estabilidad de cada parámetro fue superior a lo registrado previamente. Específicamente, existieron evaluaciones en que se alcanzó el tiempo máximo de resolución incluido por defecto en GAMS (1.000[s]), siendo el principal factor presente en los altos tiempos requeridos para determinar los rangos de estabilidad. Ante esto, sería conveniente analizar cómo acelerar la búsqueda de los rangos de estabilidad, tanto en el número de iteraciones como en el tiempo que requiere cada una de estas.

Uno de los efectos de alcanzar el tiempo máximo de resolución es la presencia de soluciones enteras, es decir, casos en que no se ha alcanzado a obtener una solución óptima global pero sí la mejor solución factible que presente valores enteros, dado el tipo de problema analizado. Estas soluciones pueden mantener o cambiar los valores de las variables de interés, por lo que aquellos casos en que se mantienen han sido incorporados y aceptados dentro del rango de estabilidad. Observando las iteraciones realizadas en el presente ejemplo, se reconoce que los tiempos de cada iteración varían, presentándose algunos más elevados y que si el tiempo máximo de resolución se hubiese acotado, también formarían parte de las soluciones *interrumpidas* o *enteras*. Por el contrario, haber extendido el tiempo de resolución puede ocasionar que casos con soluciones de esa característica logren alcanzar la solución óptima, aunque no siendo tan eficiente en términos de tiempo. Entonces, uno de los desafíos para el nuevo algoritmo será la definición del tiempo máximo conveniente de resolución por iteración, el que podría estar relacionado con el tiempo que requiere el problema original (nominal).

A continuación, se presentarán las principales reflexiones de los tres ejemplos analizados en este capítulo, destacando las oportunidades de mejora detectadas y las ideas centrales para la elaboración de un nuevo algoritmo para determinar eficientemente el rango de estabilidad.

4.4. Reflexiones de los resultados preliminares

La implementación del algoritmo referencial en los tres ejemplos anteriores permitió calcular el rango de estabilidad de los parámetros más relevantes, destacando aquellos cuyo valor nominal podría estar sostenido por supuestos o estimaciones. Esto se realizó mediante un procedimiento iterativo que puede ser complementado para generar un nuevo algoritmo. Entre los aspectos a añadir en el algoritmo que se propondrá se considera: (i) incluir pasos que permitan determinar una mejor adivinanza inicial con el propósito de disminuir las iteraciones necesarias para obtener el rango de estabilidad; (ii) definir un tiempo límite por iteración para reducir el tiempo total de implementación; (iii) establecer cotas asociadas a la interpretación lógica o física de los parámetros, permitiendo que los valores analizados sean aquellos que se relacionen con el contexto de cada problema y se evite evaluar valores que no sean de interés.

Las ideas principales para la estimación de una mejor adivinanza inicial tienen relación con el análisis de la estructura del problema, particularmente con las restricciones presentes y que incorporan a los parámetros de interés. Debido a que esto estaría condicionado a las formulaciones específicas de cada problema y que se complejizaría el análisis para modelos de gran escala, el desafío es implementar alguna metodología general que permita estimar eficientemente un mejor punto de partida para el proceso iterativo.

En relación al tiempo requerido para determinar el rango de estabilidad, es necesario establecer cotas para restringir el tiempo máximo por ejecución, ya que en caso contrario podría emplearse un período elevado en obtenerse, no siendo eficiente la determinación del rango buscado. Este aspecto presenta una mayor relevancia en los casos con mayor complejidad, debido a que podría necesitarse un tiempo del orden de miles de segundos, como ocurrió en el ejemplo 3, alcanzándose el tiempo máximo de resolución establecido por defecto en GAMS. Para definir el tiempo máximo por iteración se propone que esté condicionado a cada problema, por lo que se estudiarán en el próximo capítulo algunas características propias de cada formulación y resolución que permitan estimarlo, comprendiendo que la definición de un tiempo único que aplique para cualquier problema no adoptaría las particularidades y requerimientos de tiempo de cada modelo.

Otra de las alternativas de mejora que se detectó en el presente capítulo para disminuir el número de iteraciones y que se estudiará en el próximo corresponde a la inclusión de cotas adicionales para las iteraciones de los valores de los parámetros. Es necesario que se genere la posibilidad de incluir cotas previo a la ejecución del algoritmo, que estén basadas en características físicas o información previa que se tenga del parámetro en estudio, inspirándose en la propuesta de Ahmetovic y Grossmann [73]. Los autores mencionados incluyeron cotas para acelerar la resolución de un problema de optimización, mientras que para el algoritmo se utilizarán para restringir el rango de búsqueda a aquellos valores que sean de interés por su interpretación en la realidad. Adicionalmente, se destaca que aquellos problemas que añadan cotas adicionales para su resolución, como el tercer ejemplo estudiado y proveniente de Ahmetovic y Grossmann [73], generan dificultades para el algoritmo por las actualizaciones realizadas a los parámetros previo a cada ejecución. Particularmente, en algunas evaluaciones las cotas podrían generar que la modificación de la región factible produzca un elevado tiempo de resolución.

La información previa sobre los parámetros también se puede enfocar desde el conocimiento de los cambios posibles de sus valores, restringiendo aún más lo logrado con la interpretación física y lógica de estos. Por ejemplo, si durante la ejecución del algoritmo en algún caso se sugiere actualizar el paso y tolerancia de búsqueda, será conveniente considerar la información y conocimientos que se tengan de la interpretación del parámetro, ya que si el rango determinado hasta el momento es suficientemente amplio para lo requerido en el problema y contexto asociado, no es necesario continuar la búsqueda con un nuevo paso y tolerancia. Específicamente, para problemas del tamaño de los ejemplos 1 y 2 se podría concluir que alguno no esté acotado a partir de la tendencia del comportamiento de la solución en las diferentes iteraciones. Por el contrario, casos con una complejidad como el tercero podrían no permitir conclusiones al respecto directamente, recordando que uno de los costos de instalación (IC_{T_3}) poseía un límite superior no detectado por el algoritmo tras haber finalizado por el número de iteraciones en un valor menor al límite real.

Tanto la obtención del rango de estabilidad como su correcta interpretación son relevantes para la toma de decisiones en el contexto de la Ingeniería. Es por ello que, para poder priorizar los análisis en los parámetros y los supuestos empleados, se plantea la necesidad de establecer una herramienta que permita comparar los rangos obtenidos para diversos parámetros, que pueden presentar diferentes órdenes de magnitud, generando dificultades para ser estudiados entre sí directamente. Específicamente, esto se realizará a partir de un índice que se propondrá y describirá en el siguiente capítulo.

Finalmente, las reflexiones parciales realizadas en los ejemplos anteriores permitieron concluir que el mantener el valor de las variables binarias no implica directamente que el layout óptimo no cambia, ya que esto depende de qué decisiones del problema se han representado mediante este tipo de variables. Es por ello que en el capítulo siguiente se incorporará en el nuevo algoritmo la definición de las variables a analizar, previa decisión del usuario. Se continúa considerando que las instalaciones de plantas son las principales decisiones por la duración e implicancias que poseen, basadas en la siguiente reflexión: Si una planta que estaba en el layout óptimo deja de estarlo por un cambio en uno de los parámetros del problema, entonces los flujos que ingresaban y salían de esa planta tampoco pueden existir. Por el contrario, si un cambio en los parámetros ocasiona que alguno de esos flujos desaparezca, no necesariamente la planta dejaría de instalarse. Esta idea será parte de las consideraciones que se realizarán para escoger qué variables poseen una mayor trascendencia en el problema y cuyo valor es prioridad mantener.

Las reflexiones e ideas destacadas en esta sección serán profundizadas y formalizadas en el Capítulo 5, mientras que su implementación conjunta será efectuada en el Capítulo 6.

Capítulo 5

Creación del nuevo algoritmo

Luego de reconocer oportunidades de mejora del algoritmo de Yi y Lu [31], en el presente capítulo se elaborará una nueva estrategia para calcular el rango de estabilidad de parámetros, esta vez enfocada en problemas MINLP. Inicialmente, se justificará la necesidad de las mejoras y cómo se relacionan entre sí, para posteriormente describirlas en detalle y discutir las, finalizando el capítulo con una explicación detallada del funcionamiento del nuevo algoritmo.

5.1. Características del nuevo algoritmo

Un algoritmo iterativo para determinar la región de estabilidad de parámetros en problemas formulados como MINLP es desarrollado a partir de la versión MILP de Yi y Lu [31]. El objetivo primordial de la nueva versión es acelerar la búsqueda de los límites del rango de estabilidad, debido a que los problemas no lineales son más desafiantes que los lineales. Las principales características de la versión propuesta en contraste con el algoritmo referencial de Yi y Lu [31] son las siguientes: (i) el rango de búsqueda está acotado por información previa de los parámetros; (ii) el valor inicial de las iteraciones es estimado para ser más cercano a cada límite del rango de estabilidad; (iii) un tiempo límite es incluido en base al tiempo de resolución del problema original; (iv) junto con el rango de estabilidad, la información de salida incluye un índice para comparar los rangos de estabilidad de diferentes parámetros. La justificación de cada una de estas mejoras es presentada en la Tabla 5.1.

La relación de las mejoras mencionadas se realiza mediante el algoritmo propuesto en la Figura 5.1, donde se han incluido los pasos provenientes del algoritmo de Yi y Lu [31], junto con aquellos con cambios menores y los nuevos pasos. La formalización de los elementos requeridos en cada paso se detallará en la sección 5.2, cuyas expresiones matemáticas se contextualizarán mediante el uso de uno de los parámetros de los ejemplos previos.

El procedimiento propuesto se resume de la siguiente manera: El algoritmo comienza con la selección del parámetro y variables de interés (por defecto son todas las variables binarias). Con el valor nominal del parámetro son calculados los parámetros de búsqueda (tolerancia y paso inicial). Resolviendo el problema original se obtiene el valor óptimo de las variables binarias que se espera mantener. Estos resultados son implementados en un subproblema para determinar una posible nueva adivinanza inicial (*parámetro optimizado*, explicado en la sección 5.2.2). Este valor es evaluado en el problema original y pueden ocurrir dos situaciones: (i) mantiene la solución original, por lo que se acepta como nuevo valor inicial o (ii) cambia

la solución, por lo que se descarta y las iteraciones comienzan en el valor nominal. Estas ejecuciones continuarán hasta detenerse por alcanzar la tolerancia o el número máximo de iteraciones por paso. El procedimiento completo considera evaluar solo valores dentro del rango referencial generado con información previamente incorporada por el usuario.

Tabla 5.1: Principales mejoras incorporadas en el algoritmo propuesto.

Mejora	Justificación
(i) El rango de búsqueda está restringido por información que es conocida con anterioridad	Los resultados respecto al rango de estabilidad deben ser contrastados con la realidad. Por ejemplo, el algoritmo no debería evaluar valores negativos para flujos del proceso por la infactibilidad de adoptar aquellos valores realmente. Para los parámetros en que el usuario conoce información previamente sobre su naturaleza o infactibilidad, es posible incorporar aquella información al comienzo de algoritmo. Con ello, la búsqueda es acotada al rango de interés y se requiere un menor tiempo de implementación.
(ii) Un nuevo valor inicial de las iteraciones es estimado	Comenzar la búsqueda en un valor cercano a los límites del rango de estabilidad puede reducir el número de iteraciones requeridas para encontrarlos. El algoritmo referencial inicia desde el valor nominal, pero se propone utilizar un mejor punto inicial para acelerar la búsqueda. Debido a que la determinación del rango de estabilidad incluye dos límites (superior e inferior), es necesario estimar dos valores iniciales, independientes entre sí. Cada uno es obtenido minimizando o maximizando un subproblema con valores fijos de las variables binarias, detallado más adelante y cuyo procedimiento se ha denominado <i>optimización de parámetros</i> .
(iii) El tiempo por iteración es acotado	El tiempo de cada iteración debe tener un orden de magnitud similar al tiempo empleado en resolver el problema originalmente. Si una iteración individual requiere más tiempo, las múltiples iteraciones involucradas podrían necesitar un tiempo considerable. Para acelerar el algoritmo, en el software de optimización es posible editar el tiempo de ejecución por defecto y así acortar las resoluciones del proceso iterativo. Un caso en el que se podría acelerar la búsqueda es el ejemplo 3, ya que algunas iteraciones requerían un tiempo del orden de 1.000 [s].
(iv) Se propone un índice de estabilidad que compara rangos de estabilidad de parámetros	Comparar directamente la longitud del rango de estabilidad de diferentes parámetros podría generar conclusiones erradas, debido a los diferentes órdenes de magnitud que podrían tener cada uno. Por lo tanto, la comparación requiere una normalización y es incluida en un índice propuesto para este fin. Adicionalmente, el riesgo de cambiar el valor de las variables binarias está condicionado por la cercanía del valor nominal con los límites del rango de estabilidad. Es por ello que el índice a proponer permitirá contrastar los rangos de estabilidad considerando la distancia relativa del valor nominal a aquellos límites.

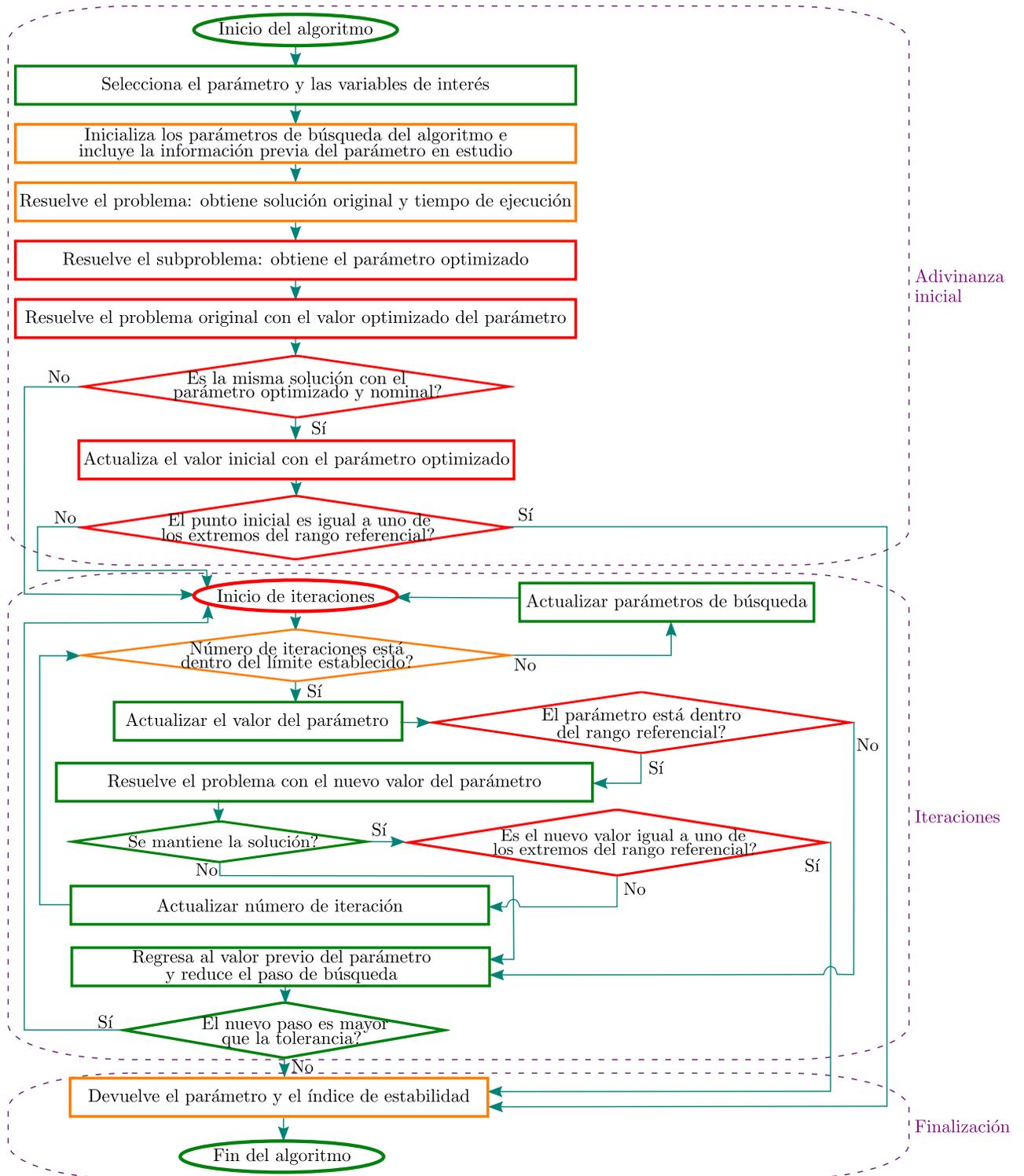


Figura 5.1: Algoritmo propuesto para obtener el rango de estabilidad de parámetros en problemas MINLP. Etapas verdes: Pasos mantenidos del algoritmo de Yi y Lu [31]. Etapas naranjas: Modificadas del algoritmo de Yi y Lu [31]. Etapas rojas: Nuevos pasos propuestos para la versión MINLP. Elaboración propia.

Para comprender el funcionamiento y cómo se implementarían las mejoras mencionadas, a continuación se describen y explicitan los elementos principales de cada una.

5.2. Descripción de las mejoras propuestas

Los elementos del algoritmo a proponer se pueden clasificar según las etapas en que intervienen, empleando la división de la Figura 5.1: *Iteraciones*, relacionado con las características que permiten la búsqueda iterativa del rango de estabilidad; *Adivinanza inicial*, enfocadas en la búsqueda y determinación de un mejor punto de partida del funcionamiento iterativo, con el propósito de disminuir el tiempo empleado en el proceso; *Finalización*, donde se propone un índice que permita comparar los rangos de estabilidad y jerarquizarlos. Cada elemento se detalla a continuación.

5.2.1. Iteraciones

5.2.1.1. Funcionamiento iterativo

Rescatando lo propuesto en el algoritmo referencial, la búsqueda del límite del rango de estabilidad se efectuará mediante múltiples iteraciones, evaluando dónde se mantienen las variables binarias y desde qué valores se producen cambios, aunque aplicando modificaciones propuestas para automatizar y reducir la interacción con el usuario si aún no se ha alcanzado la tolerancia establecida. Los elementos necesarios para estas evaluaciones forman parte de lo descrito por Yi y Lu [31] en el algoritmo mencionado en el Capítulo 3, listándose a continuación:

- Largo del paso de búsqueda (Δl): Es el número más pequeño que posea el mismo orden de magnitud del parámetro analizado.
- Tolerancia (t): 4/10000 veces el valor nominal del parámetro de entrada a estudiar. Este valor se sugiere a partir de distintas pruebas de ensayo y error.
- Contador de iteraciones (i): Inicializado en 0 y cuenta el número de iteraciones en cada paso para evitar iterar indefinidamente si no se encuentra un cambio en la solución.

Los puntos anteriores forman parte del algoritmo referencial, aunque se ha propuesto que el valor de i sea actualizado incluyendo el valor 10 (anteriormente se evaluaba hasta $i < 10$), permitiendo una solución práctica para que en los casos en que se produzca un cambio de los valores de variables binarias para $i = 10$ se actualice automáticamente el paso y continúe la búsqueda. De esta forma, se evita la interacción con el usuario si es que no se ha alcanzado la tolerancia definida inicialmente. Un ejemplo de esta situación corresponde al parámetro $FPU_{P_1}^{in}$ del ejemplo 3 en que se determinó que el algoritmo itera hasta el valor $FPU_{P_1}^{in} = 49,0$ y pide actualizar los parámetros de búsqueda (Δl , t e i) si se desea continuar buscando, pese a que iteraciones previas determinaron que existe un cambio en las variables binarias en $FPU_{P_1}^{in} = 50,0$. Comprendiendo que el límite se encuentra entre 49,0 y 50,0, y que no se ha alcanzado la tolerancia ($t = 0,016$), se debiese continuar buscando automáticamente un valor más preciso.

Se destaca que la sugerencia anterior beneficia a los casos cuyo décimo valor de i produce un cambio en las variables binarias. Si no ha ocurrido en las iteraciones realizadas, seguirá siendo el usuario quien defina si es necesario continuar buscando, ya que podría ser o no requerido determinar valores más lejanos a los evaluados. La vinculación con la realidad y las características del problema podrían orientar al usuario respecto a cuánto más iterar.

5.2.1.2. Variables de interés

El contraste del tipo de decisiones de instalación entre el ejemplo 2 y 3 permite reconocer que existen diversas interpretaciones en las redes de agua para las variables binarias. Comprendiendo que algunas podrían ser más relevantes para el contexto, como la instalación de plantas por sobre las conexiones, por el nivel de inversión y duración de la decisión, el usuario podría querer priorizar qué variables binarias mantener. Es por ello que el nuevo algoritmo incluirá una etapa al comienzo en que se definan las variables binarias de interés. Por defecto, se buscará mantener todas, tal como en los ejemplos anteriores y en el caso de estudio principal a analizar en el ejemplo 4, salvo que exista algún criterio propio del problema o del usuario que genere una decisión diferente.

5.2.1.3. Información previa de los parámetros

El análisis del algoritmo referencial mostró que en algunos casos se podrían estar evaluando valores que, desde un comienzo, se podrían descartar por no ser posibles física o lógicamente. Por ejemplo, en el problema 2 se evaluaron valores negativos para demandas y costos, comprendiéndose que no serían adoptados por los parámetros en la realidad. Ante este tipo de situaciones, se propone establecer límites en la búsqueda asociados a la información que se conozca del problema, teniendo como inspiración la mejora en el rendimiento de la solución de problemas de optimización propuesta por Ahmetovic y Grossmann [73], quienes incorporaron restricciones adicionales basadas en el conocimiento previo del problema y así disminuir el tiempo de resolución de problemas de optimización MINLP. En caso de existir aquellos límites, la búsqueda del rango de estabilidad se acotaría según la expresión 5.1:

$$a_{ref}^{min} \leq a \leq a_{ref}^{max} \quad (5.1)$$

Donde:

- a : Parámetro al que se le calculará el rango de estabilidad.
- a_{ref}^{min} : Valor mínimo que puede alcanzar el parámetro a , incluido a partir de información referencial que posea el usuario.
- a_{ref}^{max} : Valor máximo que puede adoptar el parámetro a , incluido a partir de información referencial que posea el usuario.

Es importante destacar que la definición de los límites anteriores permite extender el uso a otra situación: ¿hasta qué valor es necesario analizar la estabilidad? Si el usuario determina que no es relevante conocer qué ocurre con el valor de las variables binarias por sobre algún valor dado, puede establecerlo y analizar el rango hasta aquel punto. A priori, no se aseguraría la determinación del rango completo, pero permite una flexibilidad en cuanto al tiempo de resolución, ya que se estudiará solo un rango de valores de interés para quien lo requiera.

5.2.1.4. Tiempo límite por iteración

Considerando que el algoritmo debe ser eficiente en la búsqueda del rango de estabilidad, se propone establecer un límite de tiempo de ejecución para cada evaluación diferente al incluido por defecto en GAMS (1.000[s]), basado en el tiempo que tardó en resolverse el problema con los valores nominales. Este límite se calcularía de la siguiente forma:

$$t^{lim} = \tau \cdot t^{nom} \quad (5.2)$$

Donde:

- t^{lim} : Tiempo límite de resolución para cada iteración del algoritmo.
- t^{nom} : Tiempo que tardó el problema con los valores nominales de los parámetros.
- τ : Ponderador para ajustar el tiempo máximo de resolución para cada iteración.

La inclusión del tiempo requerido en el problema con los valores nominales permite considerar las particularidades de cada modelo, ya que se espera que a mayor complejidad, requieran un mayor tiempo por iteración. El desafío es, entonces, determinar el valor de τ , ya sea general para todos los casos o que también se adapte a cada problema.

En los Anexos C.4, D.4 y E.4 se analizaron los tiempos requeridos en múltiples iteraciones de los ejemplos 1, 2 y 3, respectivamente. El estudio de estos casos permitió reconocer que para los ejemplos 1 y 2 es posible resolver cada iteración (alrededor de 100 por cada parámetro) en un tiempo inferior a 5 veces el original. Particularmente, para el ejemplo 1 se obtenía un mayor porcentaje de soluciones con un tiempo igual o inferior al original, en contraste con el ejemplo 2, donde se presentaban más casos que requerían un tiempo superior. En el ejemplo 3 también se obtenía la mayoría de las evaluaciones en un tiempo inferior a 5 veces el original, aunque este ejemplo es más complejo, presentándose casos que requerían más tiempo. Específicamente, en todos los parámetros de este ejemplo existieron evaluaciones que necesitan un ponderador τ superior a 100 para obtener la solución sin que se interrumpiera la ejecución del modelo. Entonces, si se decidiera aumentar a un valor de τ mayor a 5, sería necesario que ese incremento sea por sobre el valor de 100 para que nuevas iteraciones obtengan una solución óptima sin que se interrumpa la resolución.

A partir de las consideraciones anteriores, ¿cuál debería ser el valor de τ ? Los resultados señalan que se necesita un mayor tiempo para casos más complejos, aunque este debe limitarse para obtener una solución en un período razonable, ya que la demora individual de las iteraciones repercutirá en el tiempo total del algoritmo. Habiéndose destacado la desventaja de no acotar el tiempo de solución, ¿qué ocurriría si se acota con un valor muy pequeño? El inconveniente de aquella limitación en el tiempo radica en que la mejor solución que encuentre en ese período podría involucrar cambios en variables binarias, pudiendo no corresponder a la solución óptima que se obtiene con tiempos mayores. Para los ejemplos 1 y 2 bastaría con usar $\tau = 5$ y todas las soluciones alcanzarían el óptimo.

Respecto al ejemplo 3, las múltiples resoluciones del Anexo E.4 para diferentes valores de τ permitieron notar que los resultados se mantienen si se emplea $\tau = 5$, aunque se presentan casos en que la solución es *Entera* o posee el carácter de *Interrumpido*. El aumento del tiempo de resolución para algunos parámetros de este ejemplo permite reducir el número de iteraciones con aquellas características del modelo y solver, aunque no generan un efecto en los valores de las variables binarias que conforman la solución. Por el contrario, se detecta que en el caso $\tau = 1$ se producen cambios en los valores de las variables binarias para 3 de los 6 parámetros estudiados, añadiéndose también un parámetro en que el tiempo de resolución con $\tau = 1$ no permite calcular la solución en todas las evaluaciones.

Considerando los resultados anteriores, se propone emplear $\tau = 5$ por defecto para el nuevo algoritmo, pudiendo ser editado por el usuario en caso de que tuviera más tiempo para

resolver cada iteración. Se reconoce que la utilización de este valor de τ corresponde a lo analizado con los tres ejemplos anteriores, por lo que una posible extensión y profundización del trabajo podría efectuarse para decidir mediante algunas características del problema (número de restricciones, variables binarias, entre otras) el mejor valor de aquel ponderador, siendo necesarios más casos para obtener una tendencia más clara. De todas formas, se destaca que la expresión propuesta para t^{lim} incorpora el tiempo original de cada problema, lo que permitirá considerar, a grandes rasgos, particularidades de cada modelo.

5.2.2. Adivinanza inicial

Los resultados y reflexiones del ejemplo 1 permitieron reconocer que existía una relación entre los extremos del rango de estabilidad y las restricciones activas del problema, ya que se presentan puntos críticos de análisis al pasar a estar activa una restricción que no lo estaba. Para encontrarlos, es necesario analizar las inecuaciones involucradas, lo que fue estudiado en mayor profundidad para los ejemplos 1 y 2 en los Anexos C.5 y D.5, respectivamente.

Comprendiendo que esta idea no es posible de implementar en aquellos casos en que las restricciones están activas para los valores nominales de los parámetros o son de igualdad originalmente, su aplicación se limitaría a estudiar las inecuaciones en que esté inmerso el parámetro de interés, según se concluyó en el Anexo E.5 respecto a las restricciones y su relación con los parámetros del ejemplo 3. De todas formas, es necesario destacar que las expresiones presentes en los ejemplos 1 y 2 no presentan una gran complejidad para su análisis y han sido escogidas precisamente por esta característica para facilitar la comprensión del efecto de los parámetros en la solución y cómo funcionaría una nueva metodología a proponer.

Las limitaciones relatadas anteriormente no permiten garantizar que en problemas más complejos sea factible este *análisis de inecuaciones* efectuado en los Anexos C.5 y D.5. Junto con ello, se destaca que se ha fijado la solución completa (variables x e y), siendo que el interés está en mantener las variables binarias en su valor original, estableciéndose la posibilidad de ser modificada la de naturaleza continua. Pese a las limitaciones presentes, el *análisis de inecuaciones* corresponde una base para la búsqueda de algún método que generalice la determinación de una mejor adivinanza inicial para el proceso iterativo del algoritmo a proponer. En particular, sería conveniente no solo modificar el parámetro en estudio si no que permitir que las variables continuas puedan también adaptarse. Comprendiendo que no siempre será posible establecer un sistema de inecuaciones que determine fácilmente una solución para esta idea propuesta, se relata a continuación una alternativa para obtener una mejor estimación inicial, correspondiente a una extensión de la búsqueda anterior, pero resolviendo un nuevo problema de optimización.

5.2.2.1. Optimización de parámetros

A partir de la dificultad del *análisis de inecuaciones* en casos con múltiples expresiones o de gran complejidad, se propone obtener la estimación inicial de búsqueda del rango de estabilidad mediante un subproblema del caso original, cuyo fundamento y metodología para ser implementado se explica a través del ejemplo 1 de esta tesis:

Como se desea conocer el valor máximo y mínimo que puede adoptar un parámetro para que la variable binaria no cambie su valor original, se propone analizar el caso en que el parámetro en estudio fuese variable y que se establezca como parámetro el valor de la variable

binaria. Esta reformulación del problema permitiría conocer el rango de valores del parámetro que podría incluir en su solución el valor original de la variable binaria. Se destaca que solamente refleja una región factible, por lo que no se puede asegurar que la implementación de todos los valores de la región determinada en el problema original generen que la solución óptima mantenga el valor nominal de la variable binaria. Es decir, se logra acotar el rango de búsqueda, pero no calcular directamente el rango de estabilidad.

Matemáticamente, el subproblema propuesto en caso de analizar el parámetro b del ejemplo 1 se plantearía según la expresión 5.3. En este modelo se ha fijado el valor de la variable binaria y también el parámetro a , dado que se está modificando la circunstancial variable b (variable en la versión reformulada y parámetro en el problema original). Como se ha destacado previamente, la resolución del problema de minimización y maximización para cada parámetro (a y b) se realizaría buscando la región factible de valores de este parámetro que permitan que la solución considerara $y = 1$ en el problema original.

$$\left\{ \begin{array}{ll} \min(\max) & b \\ & x, b \\ \text{s.a.} & \\ & \mathbf{a} - x^2 - y \leq 0 \\ & x + y \leq \mathbf{b} \\ & x \geq 0 \\ & a = 1,25, y = 1,0 \end{array} \right. \quad (5.3)$$

Se propone analizar el problema reformulado a partir del comportamiento de las restricciones que lo componen, tal como se efectuó anteriormente en las Figuras 4.2 y 4.3 para el problema original. Para esto, se grafica la variable x en función de b y a (uno a la vez), generando las Figuras 5.2.(a) y 5.2.(b), respectivamente. En cada una se destaca el punto C, correspondiente a la solución del problema de optimización reformulado de los casos en que se minimiza b y en el que se maximiza a . No se han establecido puntos que expresen la solución de la maximización de b y la minimización de a , ya que generan una solución no acotada. Se destaca también que no se han generado los gráficos en función de la variable binaria, debido a que se ha fijado su valor en estos casos ($y = 1$).

En la Figura 5.2 se observa que el menor valor de b factible es 1,5. Como el valor nominal de b es 1,6, se sugiere buscar el límite inferior entre 1,5 y 1,6, mientras que la información de la maximización de b no permite obtener conclusiones al respecto, ya que no se encuentra acotado. Respecto al parámetro a , se observa que su valor máximo posible corresponde a 1,36, mientras que no se obtiene una cota inferior para este parámetro.

Para el caso $b = 1,5$ y $a = 1,36$, el punto C observado en la Figura 5.2 y también los resultados de la *optimización de parámetros* implementada en el Anexo C.6, corresponden a extremos del rango de estabilidad de cada uno, determinados anteriormente. Sin embargo, no es posible asegurar que esta metodología permita determinar los límites en otros ejemplos, ya que solamente refleja valores extremos que pueden adoptar los parámetros para que sea factible que la solución del problema incluya el valor original de la variable binaria ($y = 1$).

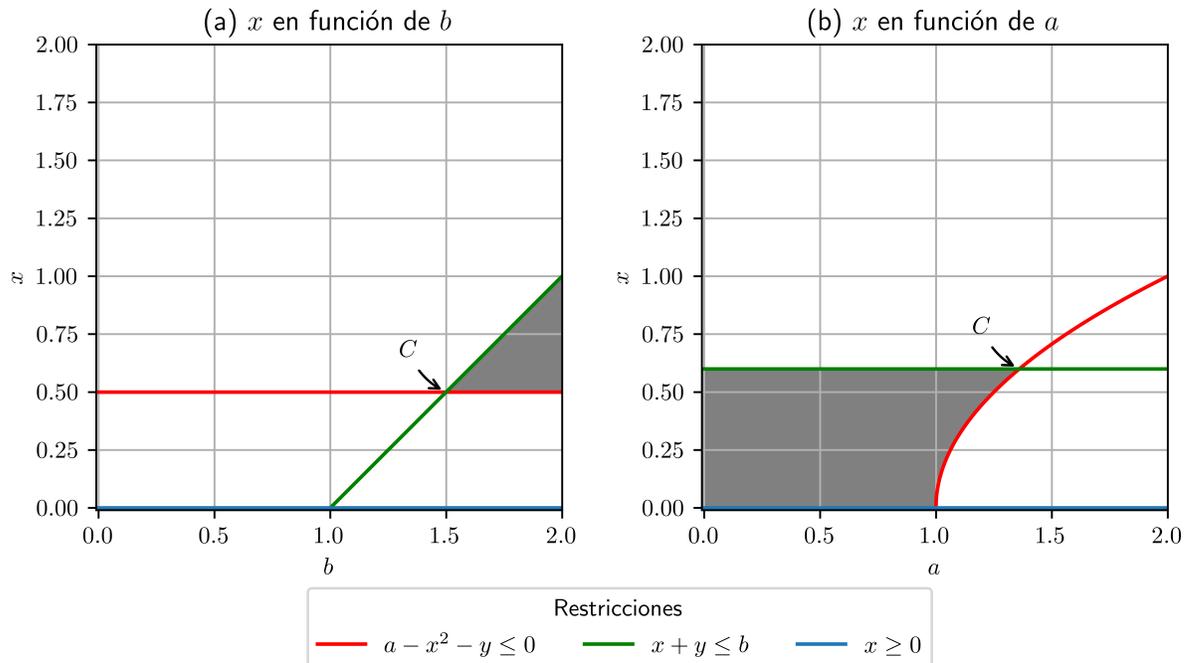


Figura 5.2: Restricciones del ejemplo 1 reformulado frente a cambios en los parámetros.

Es relevante considerar que el ejemplo utilizado permite reconocer la interpretación gráfica del subproblema, aunque para casos con un mayor número de variables y restricciones sería complejo obtener conclusiones visualmente, por lo que se requeriría resolver directamente esta nueva versión del problema. Ante la posibilidad de que los valores obtenidos en la reformulación del problema no formen parte del rango de estabilidad, si no que sean solamente puntos factibles para que la solución incluya $y = 1$, ¿cómo se podría implementar en la búsqueda de los límites del rango de estabilidad? Para que esta alternativa sea efectivamente favorable en la determinación del rango es necesario examinar los siguientes escenarios, ejemplificados para el caso en que se resuelva la minimización del problema reformulado, comprendiendo que la maximización sería análoga:

1. La evaluación de la solución del subproblema (b_{op}^{min}) en el problema original genera que la variable binaria se mantenga. Es decir, b_{op}^{min} se encuentra dentro del rango de estabilidad y es mayor o igual al límite inferior.
2. La evaluación de la solución del subproblema (b_{op}^{min}) en el problema original genera que la variable binaria cambie. Es decir, b_{op}^{min} se encuentra fuera del rango de estabilidad y es menor al límite inferior.

Para el primer caso, se recomienda utilizar este nuevo valor como punto inicial, ya que se encontraría más cercano al límite inferior que el valor nominal. En el segundo caso b_{op}^{min} se encuentra fuera del rango de estabilidad, razón por la que se sugiere no emplearlo como punto inicial. La desventaja de esta decisión radica en que se está eliminando la posibilidad de iniciar la búsqueda desde un punto fuera del rango pero cercano al límite. Como esto no se puede saber a priori, se ha optado por desestimar ese escenario ante el caso de iniciar la

búsqueda fuera y que se ejecuten múltiples resoluciones desde un punto que eventualmente podría ser lejano al límite buscado.

Frente a las situaciones descritas, se concluye que la utilización de una nueva estimación inicial se debe efectuar solamente en aquellos casos en que favorezca la búsqueda. Por esta razón se incorporaron pasos en el algoritmo de la Figura 5.1 que permiten cubrir los posibles escenarios y así realizar eficientemente la búsqueda del rango de estabilidad, asegurando comenzar a iterar desde un valor que mantiene la solución original.

5.2.3. Finalización

5.2.3.1. Índice de estabilidad

Una vez calculado el rango de estabilidad, una de las informaciones de salida del algoritmo será el cálculo del índice de estabilidad. Este índice permitirá comparar posteriormente los rangos de estabilidad de los parámetros analizados. Su implementación se explicará mediante los resultados del ejemplo 2, resumidos en la Tabla 5.2, donde se observa que los parámetros con un rango de estabilidad de menor longitud son D_{P_1} y luego D_{P_2} .

Tabla 5.2: Rango de estabilidad de los parámetros del ejemplo ilustrativo 2.

Parámetro [UN]	Descripción	Rango de estabilidad
D_{P_1} [t/h]	Demanda de agua de la planta P_1	$0 \leq D_{P_1} \leq 57,6$
D_{P_2} [t/h]	Demanda de agua de la planta P_2	$0 \leq D_{P_2} \leq 81,9$
$Cost_{T_1}$ [USD\$]	Costo de instalación de la planta T_1	$0 \leq Cost_{T_1} \leq 1.018.000$
$Cost_{T_2}$ [USD\$]	Costo de instalación de la planta T_2	$1.081.000 \leq Cost_{T_2}$

La comparación directa de las longitudes de los rangos de estabilidad podría ocasionar que la conclusión fuese que D_{P_1} y D_{P_2} son los parámetros que deben estudiarse con más detalle por el riesgo de cambiar la solución al tener un rango de menor longitud. La conclusión anterior se basa en una comparación entre valores absolutos, que podría no ser completamente correcta. Es por ello que se propone otro criterio para priorizar los parámetros y supuestos, basado en el análisis de la cercanía del valor nominal con los límites del rango de estabilidad. Para comparar parámetros con distintos órdenes de magnitud, el análisis se efectuará mediante una diferencia relativa normalizada, propuesta en la ecuación 5.4 y que se emplea para cada límite del rango de estabilidad.

$$IE_a^{inf(sup)} = \left| \frac{a^{nom} - a^{inf(sup)}}{a^{nom}} \right| \cdot 100[\%] \quad (5.4)$$

Donde:

- a^{nom} : Valor nominal del parámetro a .
- $a^{inf(sup)}$: Límite inferior (superior) del rango de estabilidad de a .
- $IE_a^{inf(sup)}$: Índice de estabilidad inferior (superior), que representa la distancia relativa entre el extremo inferior (superior) del rango de estabilidad de a y su valor nominal.

Mantener el valor original de los parámetros podría ser más riesgoso mientras más cerca se esté de los límites del rango de estabilidad, en caso de existir errores en la estimación de

aquel valor nominal. Por esta razón se plantea que el índice de estabilidad de a (IE_a) se calcule a partir de la menor distancia relativa a alguno de los extremos del rango, mediante la expresión 5.5 y que entrega los resultados de la Tabla 5.3.

$$IE_a = \min(IE_a^{inf}, IE_a^{sup}) \quad (5.5)$$

Tabla 5.3: Índice de estabilidad de los parámetros del ejemplo ilustrativo 2.

Parámetro [UN]	Rango de estabilidad	IE
D_{P_1} [t/h]	$0 \leq D_{P_1} \leq 57,6$	15 %
D_{P_2} [t/h]	$0 \leq D_{P_2} \leq 81,9$	100 %
$Cost_{T_1}$ [USD\$]	$0 \leq Cost_{T_1} \leq 1.018.000$	1,8 %
$Cost_{T_2}$ [USD\$]	$1.081.000 \leq Cost_{T_2}$	1,7 %

Pese a que las demandas D_{P_1} y D_{P_2} poseen un rango de estabilidad de menor longitud que los costos $Cost_{T_1}$ y $Cost_{T_2}$, razón por la que se sugería anteriormente profundizar el análisis en las demandas, el índice de estabilidad calculado sugiere lo contrario: Como los costos de tratamiento pueden variar máximo en un 1,8 o 1,7% (aumento de $Cost_{T_1}$ o disminución de $Cost_{T_2}$, respectivamente), se debería analizar con mayor detalle los supuestos asociados a estos parámetros, ya que el margen de error que presentan es inferior al de D_{P_1} y D_{P_2} . Como caso particular, $Cost_{T_2}$ no se encuentra acotado superiormente. En general, si un parámetro no estuviera acotado ni superior ni inferiormente, no es necesario determinar su índice de estabilidad, dado que se tendría la siguiente situación: si un parámetro tiene un rango de estabilidad no acotado, puede adoptar cualquier valor y la solución óptima se mantendrá.

La ventaja de utilizar el índice de estabilidad en vez de la longitud del rango como criterio de elección de los supuestos y parámetros a profundizar, radica en que permite comparar parámetros de distinta magnitud y considera el valor original asignado al parámetro, que estaba ausente en el criterio anterior.

5.3. Reflexiones de las ideas propuestas

Los enfoques de las mejoras propuestas buscan disminuir el tiempo requerido para alcanzar los límites del rango de estabilidad. El cumplimiento de este propósito se evaluará en el próximo capítulo, aunque se deben tener presentes las siguientes consideraciones:

Limitar el tiempo de resolución de cada iteración entrega soluciones que podrían no ser óptimas, aunque el análisis efectuado permitió reconocer que para los casos con un ponderador τ igual a 5, se alcanzan las mismas soluciones que para mayores tiempos de resolución. Esto se planteó a partir de los ejemplos analizados hasta ahora, lo que deberá ser corroborado en el caso de estudio principal del Capítulo 7, mediante su implementación y posterior etapa de validación anunciada en el Capítulo 3.

Respecto al uso de información previa, con esta se busca contextualizar los resultados, entendiendo que el algoritmo puede ser empleado en casos reales. Precisamente, esto aumenta la relevancia de acertar en la estimación de valores para los parámetros, enfatizando la importancia de analizar indicadores como el *índice de estabilidad* propuesto en este trabajo.

En relación a la *Optimización de parámetros*, se destaca que esta propuesta origina resoluciones previas a las que conforman el algoritmo referencial, por lo que su uso eficaz requerirá que la reducción en el tiempo del proceso iterativo del algoritmo sea superior al costo computacional de efectuar aquella optimización previa.

Una de las posibles extensiones del trabajo sería la implementación de algún indicador que sea similar a los multiplicadores de Lagrange para los problemas continuos. Sería conveniente contar con alguna expresión matemática cuya interpretación permita concluir sobre el efecto de los cambios en los parámetros con el menor número posible de iteraciones. Esta idea se encuentra fuera de los alcances de la presente investigación, pero se espera que el avance en el estudio de problemas MINLP y en las capacidades computacionales permitan complementar el algoritmo propuesto con algún indicador que presente las características señaladas.

Habiendo explicado y reflexionado sobre los elementos que componen el algoritmo propuesto, a continuación es ejemplificado para facilitar posteriores implementaciones.

5.4. Algoritmo propuesto

El procedimiento general de la Figura 5.1 es contextualizado al ejemplo 3 con los pasos de la Figura 5.3. La aplicación del algoritmo comienza con la selección del parámetro a estudiar (r_{T_3}) y las variables de interés (Y , que representa el conjunto de variables binarias). Luego, los parámetros de búsqueda son incluidos: paso inicial (Δl), tolerancia (t) y un contador de iteraciones (i); también es incorporada la información previa sobre cotas del parámetro ($r_{T_3,ref}^{min}$ y $r_{T_3,ref}^{max}$). Posteriormente, el problema es resuelto y la solución original de las variables binarias (Y_0) y el tiempo nominal (t^{nom}) son obtenidos. Este tiempo permite generar un límite de tiempo (t^{lim}) para cada resolución. La información mencionada y los parámetros son empleados para resolver un subproblema del original para determinar el valor optimizado del parámetro ($r_{T_3,op}^{min(max)}$). Este valor es evaluado en el problema original para examinar si la solución nominal de las variables binarias (Y_0) se mantiene o cambia.

Si el valor optimizado mantiene la solución original de las variables binarias ($Y^{min(max)} = Y_0$) y además es igual a alguno de los límites referenciales ($r_{T_3,ref}^{min}$ o $r_{T_3,ref}^{max}$), este valor corresponderá al límite buscado ya que el algoritmo no puede evaluar valores fuera del rango referencial. Por otro lado, si la solución se mantiene ($Y^{min(max)} = Y_0$) y el valor evaluado se encuentra dentro del rango referencial, entonces el proceso iterativo comienza desde esta nueva adivinanza inicial hasta encontrar el límite del rango por alcanzar la tolerancia (t) o el límite referencial. Mientras no ocurran las situaciones descritas, el paso (Δl) es actualizado tal como ocurre en el algoritmo de Yi y Lu [31], evaluando las condiciones en cada iteración. Si no ocurren cambios en las variables binarias después de 10 iteraciones ($i = 10$) para un mismo paso, entonces el algoritmo sugerirá actualizar los parámetros de búsqueda si es necesario continuar con la búsqueda del límite del rango de estabilidad. El proceso iterativo es el mismo si el parámetro optimizado genera una solución diferente ($Y^{min(max)} \neq Y_0$), aunque en este caso comenzaría desde el valor nominal. El límite (inferior o superior) del rango de estabilidad y el correspondiente índice ($SI_{r_{T_3}}^{inf(sup)}$) son obtenidos cuando la búsqueda finaliza.

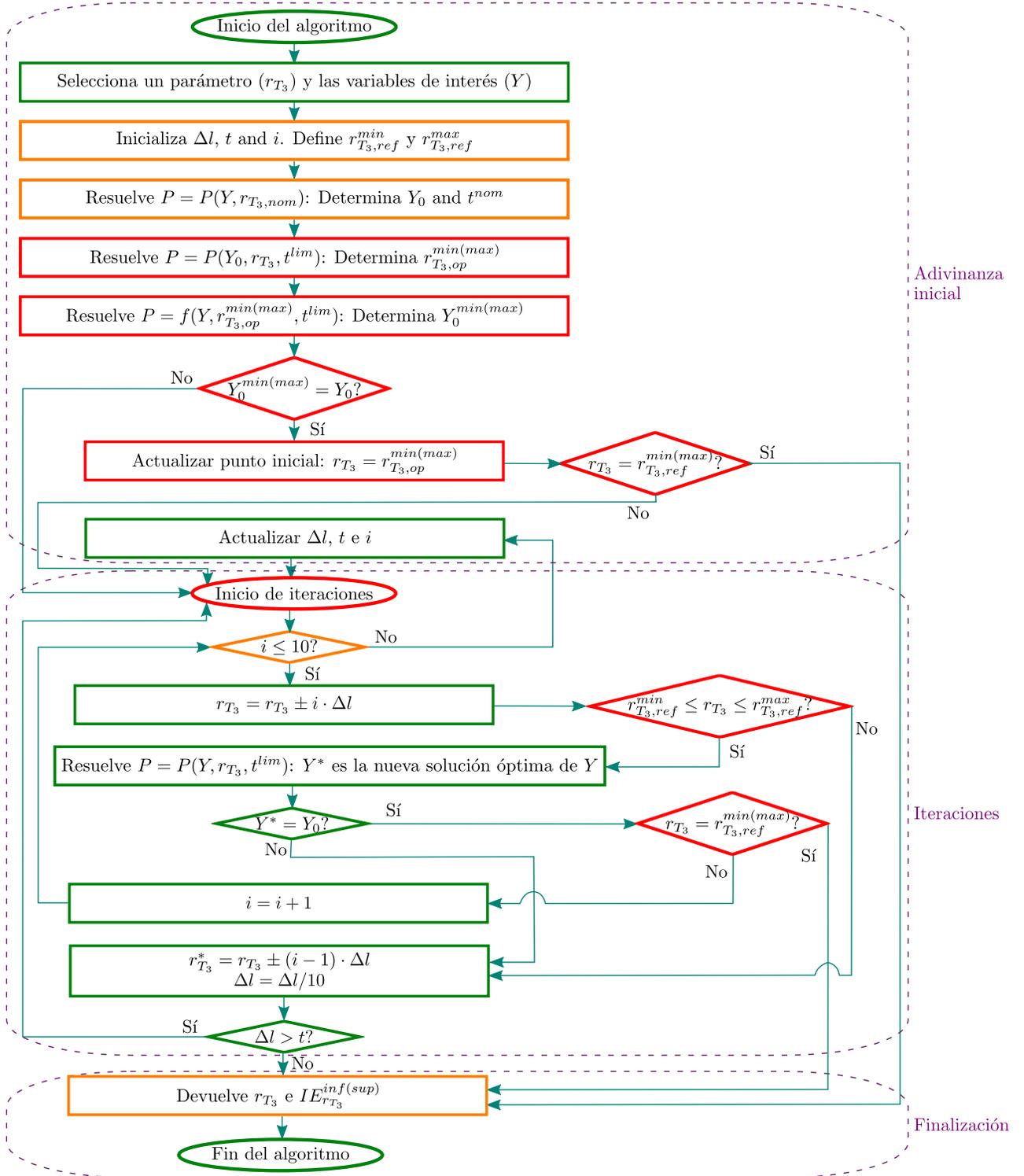


Figura 5.3: Algoritmo propuesto para obtener el rango de estabilidad de parámetros en problemas MINLP, aplicado al parámetro r_{T_3} . Etapas verdes: Pasos mantenidos del algoritmo de Yi y Lu [31]. Etapas naranjas: Modificadas del algoritmo de Yi y Lu [31]. Etapas rojas: Nuevos pasos propuestos para la versión MINLP. Elaboración propia.

Capítulo 6

Comparación de resultados del algoritmo propuesto y el referencial

El propósito de este capítulo es comparar el rendimiento del algoritmo referencial utilizado en el Capítulo 4 y el que se propuso en el Capítulo 5. El contraste se efectuará principalmente en base al tiempo empleado para la determinación del rango de estabilidad de cada parámetro, junto con la precisión de los rangos obtenidos.

Entre las principales diferencias de ambas alternativas están la propuesta efectuada para mejorar la adivinanza inicial (*Optimización de parámetros*) y la comparación de los rangos de estabilidad obtenidos (*Índice de estabilidad*). Estas ideas son implementadas y analizadas en este capítulo, que finaliza con una discusión respecto a las mejoras efectuadas y el contraste de los resultados de ambos algoritmos, con el objetivo de emplear la nueva versión eficientemente en el caso de estudio principal del Capítulo 7.

6.1. Optimización de parámetros

La denominada *Optimización de parámetros* es una alternativa para generar un nuevo punto inicial de búsqueda de los límites del rango de estabilidad, que es considerado para este fin solamente en los casos en que su evaluación en el problema original mantiene la solución inicial de las variables binarias. Esta es complementada con el uso de *Información previa de los parámetros*, comprendiendo que existen valores que física o lógicamente no es posible alcanzar, permitiendo que se incorporen estas limitaciones en la naturaleza de las variables (parámetros considerados como variables en el subproblema) o mediante restricciones adicionales.

La reformulación se implementó en los tres ejemplos ya resueltos en este trabajo, destacándose en la Tabla 6.1 si la solución de la minimización (Min) y maximización (Max) de cada parámetro mantiene el valor de las variables binarias, es decir, si se empleará como valor inicial en el algoritmo propuesto. Mayores detalles de los resultados y el tiempo requerido en los ejemplos 1, 2 y 3 se pueden encontrar en los Anexos C.6, D.6 y E.6, respectivamente.

A partir de la información expresada en la Tabla 6.1, se observa que para los ejemplos 1 y 2 es posible emplear la nueva estimación inicial en la mayoría de los parámetros. Por el contrario, en la mayoría de los casos del ejemplo 3 se desestima el uso de una nueva estimación inicial, comenzando la búsqueda desde el valor nominal.

Tabla 6.1: Factibilidad de la asignación de la solución del problema reformulado como valor inicial para determinar el límite inferior (Min) o superior (Max) del rango de estabilidad en el algoritmo propuesto.

Ej.	Parámetro	Min: ¿mantiene binaria?	Max: ¿mantiene binaria?
1	a	No	Sí
1	b	Sí	Sí
2	D_{P_1}	Sí	No
2	D_{P_2}	Sí	Sí
2	$Cost_{T_1}$	Sí	No
2	$Cost_{T_2}$	No	Sí
3	$FPU_{P_1}^{in}$	NOp ^a	NOp ^a
3	$FPU_{P_2}^{in}$	NOp ^a	NOp ^a
3	r_{T_3}	Sí	No
3	IC_{T_1}	No	Sí
3	IC_{T_2}	Sí	No ^b
3	IC_{T_3}	Sí	No ^b

^a NOp: No optimizado. La búsqueda de la solución es interrumpida por alcanzar el límite de tiempo establecido ($\tau = 5, 8,7[s]$ aprox.) y no se entrega una solución.

^b La evaluación del parámetro optimizado en el problema original generó que la solución sea infactible.

Uno de los factores asociados a la situación anterior es el tamaño del problema y, por ende, el tiempo requerido en su resolución. Los dos primeros ejemplos son de menor escala que el tercero, por lo que es posible obtener una solución en un tiempo inferior al límite impuesto. Para el caso específico del problema 3, existen cálculos previos que generan cotas adicionales y que dependen de los parámetros a optimizar, razón por la que son eliminadas para este procedimiento y su exclusión genera que el tiempo de resolución aumente, recordando que estas cotas adicionales son propuestas por Ahmetovic y Grossmann [73] para reducir el tiempo de resolución. De todas formas, se inspeccionó la opción de aumentar el tiempo límite, no obteniéndose resultados que mantengan la solución original de las variables binarias.

Finalmente, se destaca que la optimización de parámetros requiere cálculos anteriores a las iteraciones del algoritmo propuesto, razón por la que el tiempo empleado en este procedimiento previo debe ser compensado con la disminución que se logra por comenzar la búsqueda en un valor más cercano al límite del rango de estabilidad, lo que será considerado en la siguiente sección al implementar el resto de etapas del algoritmo propuesto.

6.2. Determinación del rango de estabilidad

El algoritmo propuesto es aplicado en los tres ejemplos ya analizados con el algoritmo referencial, comparándose en esta sección los resultados y el tiempo empleado por ambas alternativas. En la Tabla 6.2 se resumen los resultados de ambos algoritmos, encontrándose en mayor detalle en los Anexos C.7, D.7 y E.7, para los ejemplos 1, 2 y 3, respectivamente. Se ha destacado en color amarillo aquellos casos en que el algoritmo propuesto permitió reducir

el tiempo de búsqueda, mientras que el color celeste resalta las situaciones en que el límite obtenido fue diferente entre ambos algoritmos. Los resultados de la Tabla 6.2 muestran que en la mayoría de los casos se redujo el tiempo a partir del uso del algoritmo propuesto, mientras que la obtención de resultados diferentes para los límites en estudio son minoritarios.

Tabla 6.2: Comparación de los resultados y tiempo utilizado por el algoritmo referencial y el propuesto.

Ej.	Parámetro [UN]	Límite	Valor ref.	Valor prop.	Δt [%] ^a
1	a [–]	Inf.	0,25	0,25	–4,1
1	a [–]	Sup.	1,36	1,36	–19,9
1	b [–]	Inf.	1,5	1,5	–8,1
1	b [–]	Sup.	10,6 ^b	NA ^c	–90,9
2	D_{P_1} [t/h]	Inf.	0,0	0,0	–80,0
2	D_{P_1} [t/h]	Sup.	57,6	57,6	0,3
2	D_{P_2} [t/h]	Inf.	0,0	0,0	–90,8
2	D_{P_2} [t/h]	Sup.	81,9	81,9	–83,8
2	$Cost_{T_1}$ [USD\$]	Inf.	–1.100.000	0,0	–87,0
2	$Cost_{T_1}$ [USD\$]	Sup.	1.018.000	1.018.000	0,4
2	$Cost_{T_2}$ [USD\$]	Inf.	1.081.000	1.081.000	17,9
2	$Cost_{T_2}$ [USD\$]	Sup.	10.100.000 ^b	NA ^c	–94,7
3	$FPU_{P_1}^{in}$ [t/h]	Inf.	11,5	11,5	–97,7
3	$FPU_{P_1}^{in}$ [t/h]	Sup.	49,0 ^b	49,9	–97,3
3	$FPU_{P_2}^{in}$ [t/h]	Inf.	41,0 ^b	40,1	–97,3
3	$FPU_{P_2}^{in}$ [t/h]	Sup.	55,3	55,3	0,1
3	r_{T_3} [%]	Inf.	80,8	80,8	–78,4
3	r_{T_3} [%]	Sup.	97,4	97,4	43,4
3	IC_{T_1} [] ^d	Inf.	4.150	4.150	–93,1
3	IC_{T_1} [] ^d	Sup.	100.000	NA ^c	–99,6
3	IC_{T_2} [] ^d	Inf.	–65.000 ^b	0,0	–99,8
3	IC_{T_2} [] ^d	Sup.	115.000 ^b	125.000 ^b	64,6
3	IC_{T_3} [] ^d	Inf.	–60.000 ^b	0,0	–90,8
3	IC_{T_3} [] ^d	Sup.	120.000 ^b	130.000 ^b	38,2

^a Δt : Variación relativa del tiempo requerido al emplear el algoritmo propuesto respecto al referencial.

^b Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

^c NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresable en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

^d El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [\\$]. Ver Anexo E.1.

La reducción del tiempo necesario en los parámetros de los dos primeros ejemplos se basa principalmente en la obtención de una mejor estimación inicial a partir de la *Optimización de parámetros* y de realizar la búsqueda dentro de un rango acotado según la información previa que se tiene sobre los parámetros (demandas y costos son no negativos). Las estimaciones

iniciales también influyen en las reducciones de tiempo en los casos analizados del ejemplo 3, aunque se deben mayormente al límite de tiempo impuesto por iteración, logrando una reducción mayor a 90 % en la mayoría de los casos.

Existieron también casos en que el tiempo aumentó, aunque en magnitudes inferiores a las disminuciones presentadas en los otros parámetros. El factor principal para los crecimientos se debe al tiempo utilizado para estimar un nuevo valor inicial de búsqueda, y cuyo efecto es despreciable en los casos en que se puede emplear el nuevo valor inicial, pero en aquellos en que no permite mantener el valor de las variables binarias se traduce como un aumento de tiempo que se añade a lo requerido en las iteraciones desde el valor nominal.

Considerando el número de casos en que se disminuye el tiempo y la magnitud de los decrecimientos, se destaca que es un valor superior al aumento total de tiempo. Específicamente, con el nuevo algoritmo se logra reducir el tiempo total empleado en un 35,0 %, 37,3 % y 96,6 % para los ejemplos 1, 2 y 3, respectivamente. De esta forma, el uso del algoritmo propuesto permite lograr uno de los objetivos principales: reducir el tiempo necesario para determinar el rango de estabilidad de un parámetro.

En relación a los límites obtenidos, la mayoría de los rangos son iguales entre ambos algoritmos. Los casos en que difieren se deben principalmente a la inclusión de cotas asociadas a información previa de los parámetros y al aumento de iteraciones a realizar para un mismo paso (i), lo que permitió obtener una mayor precisión en los resultados. Particularmente, con el algoritmo propuesto ya no existen cotas inferiores negativas (en los casos en que lógicamente no es posible) y también se logra que la detención en la implementación sea mayoritariamente por alcanzar la tolerancia establecida. Con ello, se reduce la interacción con el usuario, aunque aún existen casos que la requieren y están ligados a aquellos en que no se logra detectar cambios en las variables binarias en ninguna de las iteraciones efectuadas.

A partir del nivel de precisión obtenido con el algoritmo propuesto por sobre el desempeño del algoritmo referencial, los resultados de la versión propuesta serán considerados para la determinación del *Índice de estabilidad* en la siguiente sección.

6.3. Índice de estabilidad

Una de las novedades de este trabajo es la creación de un índice de estabilidad que permita comparar los rangos de estabilidad de parámetros de un mismo problema, pese a que puedan poseer diferentes órdenes de magnitud. Este índice fue descrito en el Capítulo 5 (ecuación 5.5) y se implementó en los parámetros de los ejemplos 1, 2 y 3 en los Anexos C.8, D.8 y E.8, respectivamente. Sus resultados se resumen en la Tabla 6.3 y son analizados a continuación:

El índice de estabilidad de los parámetros del ejemplo 1 muestran que el valor nominal de b se encuentra relativamente más cerca de uno de los extremos del rango de estabilidad que a , razón por la que corresponde al parámetro crítico que debe ser analizado más detalladamente para evitar que los supuestos realizados generen una solución óptima que no fuese la real. Pese a no tener un contexto este problema, el razonamiento corresponde al que se efectuaría en casos cuya interpretación tiene relación con decisiones de instalación de plantas o conexiones en el diseño de redes de agua, tal como se analizará respecto a los ejemplos 2 y 3.

Tabla 6.3: Índice de estabilidad de los parámetros de los ejemplos 1, 2 y 3.

Ej.	Parámetro [UN]	Nominal	Rango de estabilidad	IE [%]
1	a [-]	1,25	$0,25 \leq a \leq 1,36$	8,8
1	b [-]	1,60	$1,50 \leq b$	6,3
2	D_{P_1} [t/h]	50	$0 \leq D_{P_1} \leq 57,6$	15,2
2	D_{P_2} [t/h]	40	$0 \leq D_{P_2} \leq 81,9$	100,0
2	$Cost_{T_1}$ [USD\$]	1.000.000	$0 \leq Cost_{T_1} \leq 1.018.000$	1,8
2	$Cost_{T_2}$ [USD\$]	1.100.000	$1.081.000 \leq Cost_{T_2}$	1,7
3	$FPU_{P_1}^{in}$ [t/h]	40	$11,5 \leq FPU_{P_1}^{in} \leq 49,9$	24,8
3	$FPU_{P_2}^{in}$ [t/h]	50	$40,1 \leq FPU_{P_2}^{in} \leq 55,3$	10,6
3	r_{T_3} [%]	95	$80,8 \leq r_{T_3} \leq 97,4$	2,5
3	IC_{T_1} [] ^a	10.000	$4.150 \leq IC_{T_1}$	58,5
3	IC_{T_2} [] ^a	25.000	$0 \leq IC_{T_2} \leq 125.000^b$	100,0
3	IC_{T_3} [] ^a	30.000	$0 \leq IC_{T_3} \leq 130.000^b$	100,0

^a El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [\$]. Ver Anexo E.1

^b Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

En cuanto al ejemplo 2, los resultados de la Tabla 6.3 muestran que, pese a que los costos ($Cost_{T_1}$ y $Cost_{T_2}$) poseen rangos de estabilidad de mayor amplitud que las demandas (D_{P_1} y D_{P_2}), estas últimas presentan una mayor distancia relativa al límite más cercano del rango de estabilidad, debido a la normalización efectuada. Entonces, como $Cost_{T_2}$ tiene el menor índice de estabilidad, se recomendaría comenzar analizando los supuestos y estimaciones de aquel parámetro, ya que un error superior al 1,7% generaría que la solución óptima fuese distinta a la obtenida con el valor nominal propuesto, implicando que la planta a instalar en la red de agua fuese otra y no la escogida. Si se tiene más tiempo, se continuaría analizando $Cost_{T_1}$, para luego estudiar las estimaciones de las demandas.

En relación al ejemplo 3, r_{T_3} corresponde al parámetro con menor índice de estabilidad. Específicamente, un error de un 2,5% en la estimación efectuada sobre el porcentaje de remoción de contaminante en la planta T_3 originaría que las variables binarias presentaran otro valor, es decir, la red óptima sería diferente a la que se propone instalar con los valores nominales estimados. De esta forma, se recomendaría analizar inicialmente aquel parámetro, para posteriormente estudiar los siguientes si se cuenta con los recursos para ello.

Se destaca que los parámetros IC_{T_2} e IC_{T_3} fueron los únicos en que no se determinó uno de sus límites del rango de estabilidad a partir de la tolerancia o alcanzar el mayor número expresable como solución. Entonces, para estos parámetros sería posible que el límite superior fuese mayor y, precisamente, mediante la inspección efectuada en el Anexo E.3 se corroboró que el límite de IC_{T_3} se encuentra entre 130.000 y 140.000, por lo que una eventual continuación en las iteraciones hubiese mostrado aquella situación. De todas formas, el índice calculado presenta los valores de 100% a raíz de la cercanía con el límite inferior, así que una mayor distancia respecto al superior no hubiese cambiado el valor del índice, según cómo se ha formulado.

6.4. Discusiones

La implementación del algoritmo propuesto permitió obtener rangos de estabilidad con mayor precisión que los determinados mediante el algoritmo referencial, priorizando la detención por alcanzar la tolerancia definida. Esto conlleva un mayor número de iteraciones al seguir calculando sin interactuar con el usuario tras haber detectado algún cambio en las variables binarias, situación que no ocurría con el algoritmo referencial. De todas formas, este mayor número de iteraciones en la cercanía del límite del rango de estabilidad no generó necesariamente un mayor tiempo, debido a la inclusión de otras propuestas para mejorar el rendimiento del algoritmo. Específicamente, el tiempo total empleado en todos los ejemplos se redujo mediante el uso del algoritmo propuesto.

Entre las ideas desarrolladas, se destaca que la incorporación de cotas lógicas o físicas provenientes de conocimientos previos favorece la disminución del tiempo necesario para la determinación del rango de estabilidad, ya que restringe el rango de búsqueda a valores que sean posibles de obtener según el contexto asociado a cada problema.

Otro de los cambios sugeridos se relaciona con la búsqueda de una mejor estimación inicial, es decir, un valor con el que se inicien las iteraciones y que esté más cerca del límite del rango de estabilidad buscado. A diferencia de la idea discutida anteriormente, la *Optimización de parámetros* empleada para encontrar una nueva estimación inicial no favorece en todos los casos al tiempo requerido, ya que en algunas ocasiones se evalúan valores que no permiten obtener un valor que mantenga la solución óptima, privilegiándose el inicio de las iteraciones desde el valor nominal del parámetro en esos casos. Teniendo presente aquella precaución, se destaca que en las situaciones en que se logra emplear la nueva estimación inicial se favorece la disminución del tiempo en mayor magnitud que el empleado en la determinación de ese valor, tal como se observó en el contraste entre ambos algoritmos.

El uso de una limitación para el tiempo empleado por iteración también generó una disminución en el tiempo total de búsqueda de los límites del rango de estabilidad. Se ha empleado un ponderador $\tau = 5$ para todos los casos, siendo suficiente para obtener el rango de estabilidad y con mayor precisión que la versión referencial, aunque se debe analizar si aquel valor propuesto es el mejor para casos de mayor escala. Esta limitación en el tiempo fue la más significativa para la diferencia de rendimiento con el algoritmo referencial, ya que este último presentó casos con iteraciones que alcanzaban un tiempo cercano a los 1.000[s], siendo superior en órdenes de magnitud al tiempo de las iteraciones del algoritmo propuesto.

En cuanto al *Índice de estabilidad*, se destaca que permitió comparar rangos de estabilidad de parámetros con diferentes unidades y magnitudes, empleando también el valor nominal para poder cuantificar el riesgo que existe en fallar sobre la estimación efectuada en aquel valor. Se espera que pueda ser empleado por las y los tomadores de decisiones para casos en que se necesite priorizar qué parámetros analizar con mayor detalle y evitar que la solución óptima obtenida no sea la que realmente es mejor en el contexto particular en que se enmarque el estudio o proyecto.

Las discusiones anteriores se tendrán en consideración para la validación del algoritmo en el ejemplo de escala de ciudad del siguiente capítulo.

Capítulo 7

Validación del algoritmo propuesto

El presente capítulo tiene como propósito validar el algoritmo propuesto mediante su implementación en un problema de escala de ciudad. Este caso de estudio consiste en la optimización de la red de agua del Gran Santiago y se determinará el rango de estabilidad de cinco parámetros. Conociendo estos resultados, se procederá a realizar evaluaciones con valores aleatorios de los parámetros para determinar si presentan el comportamiento esperado en base al rango de estabilidad determinado. El detalle del proceso de validación se presentará luego de la aplicación del algoritmo propuesto, descrita a continuación:

7.1. Ejemplo ilustrativo 4: Red de agua de Santiago

El caso de estudio principal corresponde a la red de agua del Gran Santiago, en la Región Metropolitana, Chile. El modelo de optimización que lo representa fue planteado inicialmente por Gormaz-Cuevas et al. [74] como un problema MILP. Considerando la posibilidad de emplear otras estrategias para la optimización de la red, fue formulado posteriormente este caso como un problema MINLP por Gormaz-Cuevas et al. [75], destacando la inclusión de una expresión no lineal: ecuación de Bernoulli. Según lo declarado por las y los autores, el problema a gran escala no puede ser resuelto en un tiempo inferior a 72 [hrs]. Es por ello que realizan simplificaciones, generándose un problema MINLP de menor escala que puede ser resuelto en un tiempo inferior. La formulación de esta versión se encuentra en el Anexo F.1, siendo analizada posteriormente en el Anexo F.2, realizándose modificaciones adicionales para la correcta implementación del algoritmo propuesto. Las características principales del modelo de optimización son las siguientes:

El problema busca decidir la configuración óptima del sistema de agua a nivel regional, dividiéndose Santiago en 4 sectores, según lo observado en la Figura 7.1: Noroeste (NW), Noreste (NE), Suroeste (SW) y Sureste (SE). El enfoque del problema considera aspectos ambientales y económicos, planteándose como un caso multi-objetivo, resuelto mediante el método de *programación por metas*. Específicamente, el modelo está estructurado para decidir la instalación de nuevas plantas y la necesidad de mantener las existentes, junto con las conexiones que se presentarían en la nueva versión integrada de la red. Se considera la reutilización de agua con 4 calidades o concentraciones diferentes (potable, dulce, riego y descarga), abarcando diversos consumidores (residencial, agrícola e industrial).

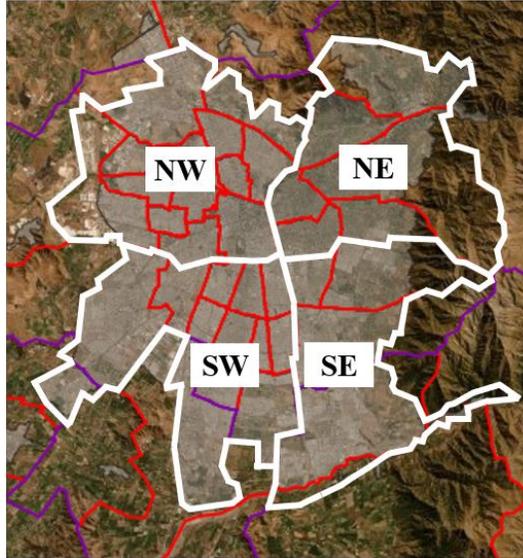


Figura 7.1: División de Santiago para el ejemplo ilustrativo 4. Donde NW: Noroeste, NE: Noreste, SW: Suroeste y SE: Sureste. Imagen obtenida de [75].

Entre los parámetros presentes en la formulación, en este capítulo se analizarán las demandas de agua para los distintos tipos de consumidores, el costo de electricidad y un parámetro que representa la cota empleada en las restricciones del tipo *Big M*, siendo este último estudiado para evidenciar el efecto de las decisiones en parámetros que no tienen una interpretación física o directa con la realidad. Otros elementos principales de la formulación MINLP se describen a continuación, mientras que el modelo completo se encuentra en el Anexo F.

7.1.1. Planteamiento

7.1.1.1. Superestructura

Para el planteamiento de este problema a escala ciudad, se consideran algunos nodos propios de cada uno de los 4 sectores en que se divide la ciudad, como también otros que permiten sus interconexiones. Las plantas de tratamiento de agua presentes pueden ser de gran y pequeña escala, por lo que sus conexiones podrían variar. Específicamente, aquellas plantas de gran escala solamente pueden suministrar agua potable a consumidores mediante el paso por un nodo de distribución, mientras que las pequeñas pueden conectarse directamente a estos. Por otro lado, el agua residual que se envía a plantas de gran escala requiere que los flujos se agrupen en el nodo de recolección, mientras que para las plantas de menor escala es posible enviar el flujo de agua residual directamente. Las conexiones de estas plantas con los consumidores dependerá de la calidad de agua de salida que tendrán, siendo una decisión en el problema de optimización. El número de nodos según su tipo se resume en la Tabla 7.1.

Los nodos mencionados se pueden conectar según la superestructura de la Figura 7.2. En esta imagen se han destacado los flujos según su calidad, lo que condiciona el tipo de consumidor al que se conectan. Las decisiones principales tendrán relación con la existencia en la red de las plantas de tratamiento de agua potable presentes en el caso base (DWTP(e)), plantas de tratamiento de agua residual existentes (WWTP(e)) y nuevas

Tabla 7.1: Número de nodos por tipo en la red de agua del ejemplo ilustrativo 4.

Tipo de nodo	Notación	Número
Fuentes de agua	$W(i=s,g)$	2^a
Plantas existentes de tratamiento de agua potable	DWTP(e)	2
Plantas existentes de tratamiento de agua residual	WWTP(e)	2
Plantas nuevas de tratamiento de agua	WWTP(n)	2
Plantas de tratamiento de efluentes industriales	IWTP	4
Consumo residencial	RC	4
Consumo agrícola y riego	AC	4
Consumo industrial	IC	4
Distribución	D	1
Recolección	C	1
Descarga a fuentes naturales	K	1

^a Se considera en el caso propuesto por Gormaz-Cuevas et al. [75] una fuente superficial y una subterránea, asociada a la planta existente de agua potable del distrito correspondiente. Para facilitar su visualización, se incluyen nodos virtuales de fuentes naturales en cada distrito.

plantas de tratamiento (WWTP(n)). Para estas plantas nuevas, cada una podría tener una de las cuatro calidades de agua mencionadas, como también ser de gran o pequeña escala. A diferencia del caso propuesto por Gormaz-Cuevas et al. [75], se ha incluido la posibilidad de transportar agua residual a plantas IWTP de diferentes distritos. Algunas consideraciones adicionales que se pueden observar en la superestructura son las siguientes:

- El consumo residencial e industrial utiliza agua potable como alimentación, mientras que el agua requerida para riego agrícola y de parques puede ser obtenida con aquella calidad específica desde plantas WWTP(n) o se puede utilizar agua directamente desde fuentes naturales con calidad de agua dulce.
- No puede existir más de una planta en la misma ubicación, por lo que las líneas discontinuas asociadas a WWTP(n) solo podrán ser de una misma calidad por planta.
- Existen fuentes naturales superficiales y subterráneas. En cada distrito se podrá escoger la fuente requerida, aunque las plantas DWTP(e) tienen asignado un tipo de fuente específico cada una.
- Existe un nodo que representa el agua que regresa a fuentes naturales (K). Sin embargo, también se considera un sumidero en la red para las pérdidas de agua, que no ha sido explicitado en la Figura 7.2. Estas pérdidas de agua se producen por la operación de plantas, absorción en el riego, entre otros.
- Las plantas existentes de gran escala son DWTP(e) en el distrito SE y WWTP(e) en SW, razón por la que se conectan con los nodos de distribución (D) y recolección (C).
- Las plantas existentes de pequeña escala son DWTP(e) en el distrito NW y WWTP(e) en NE, por lo que no se conectan con los nodos de distribución (D) y recolección (C), si no que se relacionan directamente con los nodos de consumo.

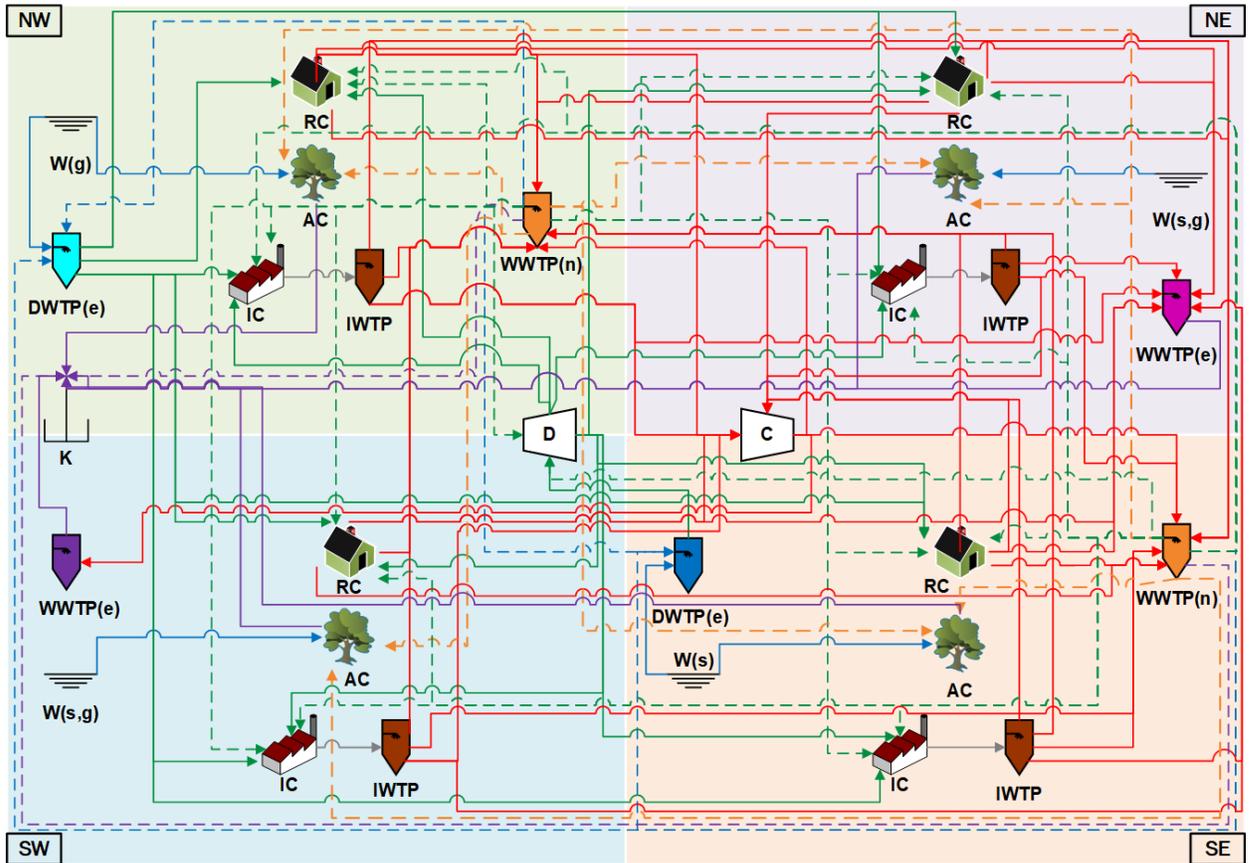


Figura 7.2: Superestructura del ejemplo ilustrativo 4. El color de las líneas representa la calidad del agua (verde: potable, azul: dulce, naranja: riego, roja: descarga y gris: efluentes industriales). Pese a no ser representadas, se permiten las conexiones de IC con IWTP de diferentes distritos. La existencia de las líneas discontinuas dependerá de la calidad de salida del agua de las WWTP(n). Elaboración propia con información de [75].

7.1.1.2. Parámetros

Los parámetros que componen el problema se encuentran en el Anexo F.1, destacándose en la Tabla 7.2 los principales para el análisis del rango de estabilidad. Particularmente, las tres demandas corresponden al consumo del distrito con mayor demanda por tipo de consumidor. Por otro lado, el costo de electricidad está relacionado con los costos de operación de la red. Finalmente, el parámetro M se escoge como representativo de aquellos parámetros empleados en el planteamiento matemático, pero que no tienen una interpretación directa con la realidad.

7.1.1.3. Variables

- $F_{i,j}$: Flujo de agua del nodo i al nodo j .
- E_i : Existencia del nodo i . El problema incluye un total de 20 variables de este tipo.
- γ : Variable a ser minimizada en el método de *programación por metas*.

Tabla 7.2: Valores nominales de los parámetros de interés del ejemplo ilustrativo 4.

Parámetro	Descripción	Valor	Unidad
$DM_{\{RC,NW\}}$	Demanda de agua para consumo residencial en el distrito NW	3,83	$[m^3/s]$
$DM_{\{AC,SE\}}$	Demanda de agua para consumo agrícola en el distrito SE	6,17	$[m^3/s]$
$DM_{\{IC,SE\}}$	Demanda de agua para consumo industrial en el distrito SE	1,18	$[m^3/s]$
E	Costo de electricidad	0,158	$[USD/kWh]$
M	Cota para restricciones del tipo <i>Big M</i>	1.000	$[-]$

7.1.1.4. Restricciones

Las restricciones del problema son descritas a continuación, mientras que las expresiones matemáticas correspondientes y reportadas por Gormaz-Cuevas et al. [75] en el problema MINLP se encuentran planteadas en el Anexo F.1.

Balances de masa

Se asume que los flujos de salida de las plantas de tratamiento presentan la calidad requerida por el consumidor, necesiéndose solamente balances de masa globales. Adicionalmente, se asume estado estacionario y densidad constante en los flujos, pudiendo ser planteados balances de flujos volumétricos.

Suplir la demanda

Cada consumidor tiene una demanda específica, la que debe ser suplida por los flujos que se dirigen a estos. Los parámetros de interés $DM_{\{RC,NW\}}$, $DM_{\{AC,SE\}}$ y $DM_{\{IC,SE\}}$ componen parte de estas restricciones.

Capacidad de tratamiento

La capacidad de una planta de agua de un determinado distrito debe cubrir la demanda de los usuarios de aquel distrito. Adicionalmente, se necesita que las plantas de tratamiento de agua residual existentes puedan adaptarse a fluctuaciones en el consumo de verano e invierno. En caso contrario, no son requeridas en la red.

Relaciones lógicas de existencia

Para que los flujos se dirijan a una planta, esta debe estar instalada. El parámetro M que se estudiará en este capítulo se emplea en este tipo de restricciones (*Big M*).

Costos

Los costos presentes en el problema se clasifican en operacionales y de inversión. Dentro de los operacionales se encuentra el costo de electricidad E .

Potencial de calentamiento global

Este indicador se calcula a partir del potencial de calentamiento global individual de cada planta y los flujos asociados a esta.

7.1.1.5. Funciones objetivo

Como se mencionó previamente, este problema es multi-objetivo e incluye tres funciones a ser minimizadas: consumo de agua de fuentes naturales, potencial de calentamiento global total y costos totales de la red. Las primeras son clasificadas como ambientales, mientras que el enfoque de la tercera es económico. Para la resolución mediante el método de *programación por metas* se ha asignado una importancia relativa de 50 % para la función económica y 50 % para las ambientales en conjunto, desglosándose en 25 % la minimización del consumo de agua y 25 % para la minimización del potencial de calentamiento global total.

La importancia relativa escogida permite que las diferentes funciones objetivo sean relevantes en el método de *programación por metas*, debido a que valores inferiores para el caso económico podrían subestimar la solución, tal como se analizó en detalle en el Anexo F.2, donde se reconoce que las normalizaciones efectuadas pueden afectar la relevancia de alguna de las funciones objetivo, en particular de la económica en este problema. Es por ello que se deben escoger importancias relativas y métodos de normalización que permitan una comparación entre las tres funciones objetivo, escogiéndose para este ejemplo una importancia igual para los aspectos ambientales y económicos. Se podrían utilizar otros ponderadores, pero se recomienda analizar la normalización y qué efecto tienen las funciones objetivo en la minimización de la variable γ para obtener conclusiones precisas respecto a la solución óptima.

7.1.2. Resultados del modelo de optimización

Al resolver el problema de optimización incluyendo los valores nominales de los parámetros se obtiene la red de la Figura 7.3, donde se observa que se han considerado tres plantas: se mantiene la planta existente de tratamiento de agua potable en el distrito NW, mientras que se incorporan dos plantas de tratamiento con calidad de salida de agua potable en los distritos NW y SE. Las tres plantas son de pequeña escala, por lo que se pueden conectar directamente con los consumidores, lo que permite disminuir pérdidas de agua por el uso de los nodos de distribución y recolección. Esta red óptima busca potenciar el reciclaje de agua, por lo que la minimización de la extracción de agua de fuentes naturales se basa en no generar agua con calidad de descarga que sea enviada al nodo de descarga K , salvo el agua propia de riego que regresa a fuentes naturales por filtración.

Para la implementación del algoritmo propuesto, se tendrá presente que en la solución óptima las tres variables binarias con valor igual a 1 son: instalación de planta de agua potable existente de pequeña escala en NW ($E_{\{DG,NW\}}$) e instalación de plantas de tratamiento nuevas de pequeña escala y con calidad de agua potable en los distritos SE ($E_{\{NA^s,SE\}}$) y NW ($E_{\{NA^s,NW\}}$). El resto de las variables binarias adoptan un valor igual a 0, según lo señalado en el Anexo F.3. Se destaca que solamente podrían existir un máximo de cuatro variables binarias iguales a 1 a la vez, ya que podrían haberse instalado también la planta de agua potable existente de gran escala en SE, pero cualquier otra opción se basa en el cambio de tamaño y calidad de salida de las plantas WWTP(n), instalándose en la misma ubicación y reemplazando las escogidas en el escenario mostrado en la Figura 7.3.

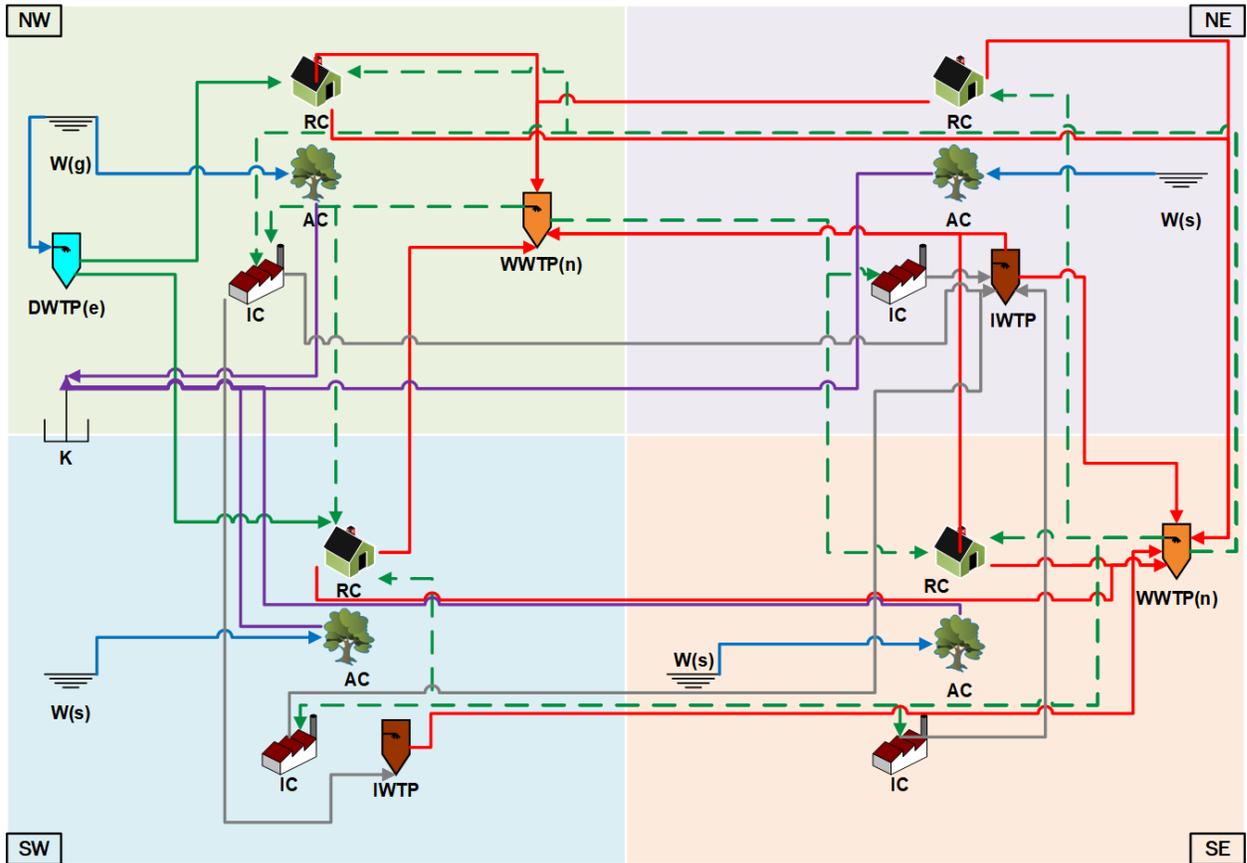


Figura 7.3: Red óptima del ejemplo ilustrativo 4. El color de las líneas representa la calidad del agua (verde: potable, azul: dulce, roja: descarga y gris: efluentes industriales). Las líneas verdes discontinuas señalan que las plantas WWTP(n) presentan una calidad de salida de agua potable.

7.1.3. Rango de estabilidad

La aplicación del algoritmo propuesto inicia con la *optimización de parámetros*, cuyos resultados están en el Anexo F.4 y se pueden emplear como adivinanza inicial solo si mantienen la solución original. La factibilidad de emplear el nuevo punto inicial en la búsqueda del límite inferior y superior de cada parámetro se resume en la Tabla 7.3:

Tabla 7.3: Factibilidad de la asignación de la solución de los subproblemas (Min. y Max.) del ejemplo 4 como valor inicial en el algoritmo propuesto.

Parámetro	Min: ¿mantiene binaria?	Max: ¿mantiene binaria?
$DM_{\{RC,NW\}}$	No	No
$DM_{\{AC,SE\}}$	Sí	No
$DM_{\{IC,SE\}}$	No	No
E	No	No ^a
M	Sí	Sí

^a GAMS reporta un error en la evaluación de este valor, no lográndose determinar una solución.

La optimización de parámetros permite concluir que es posible utilizar tres resultados de los subproblemas planteados como valor inicial en las iteraciones del algoritmo. Estas evaluaciones y las de los demás parámetros se detallan en el Anexo F.5, resumiéndose los resultados principales en la Tabla 7.4. Se destaca que los parámetros en que se emplea un nuevo valor inicial, $DM_{\{AC,SE\}}$ y M , son aquellos que requieren un tiempo menor en la determinación del rango de estabilidad. Particularmente, se observa que el parámetro M no está acotado superiormente, tal como se esperaba por ser asignado con un valor suficientemente grande en restricciones del tipo *Big M*.

Tabla 7.4: Rango de estabilidad de los parámetros del ejemplo ilustrativo 4, obtenidos mediante el algoritmo propuesto.

Parámetro [UN]	Nominal	Rango de estabilidad	Tiempo [s]
$DM_{\{RC,NW\}} [m^3/s]$	3,83	$3,11 \leq DM_{\{RC,NW\}} \leq 4,55$	4.306
$DM_{\{AC,SE\}} [m^3/s]$	6,17	$2,96 \leq DM_{\{AC,SE\}} \leq 8,66$	1.244
$DM_{\{IC,SE\}} [m^3/s]$	1,18	$0,43 \leq DM_{\{IC,SE\}} \leq 1,46$	8.501
$E [USD/kWh]$	0,158	$0,1342 \leq E \leq 0,2664$	6.600
$M [-]$	1.000	$7,22 \leq M$	256

7.1.4. Índice de estabilidad

A partir de los rangos de estabilidad obtenidos, junto con el valor nominal de cada parámetro, se determinan los índices de estabilidad inferiores y superiores en el Anexo F.6, mientras que el índice de estabilidad de cada uno (mínimo valor entre el índice inferior y superior) se expresa en la Tabla 7.5. Con estos resultados se concluye que el parámetro crítico para mantener la solución óptima es el costo de electricidad E , ya que una disminución mayor al 15,1% genera que la solución sea diferente.

Tabla 7.5: Índice de estabilidad de los parámetros del ejemplo ilustrativo 4.

Parámetro [UN]	Nominal	Rango de estabilidad	IE [%]
$DM_{\{RC,NW\}} [m^3/s]$	3,83	$3,11 \leq DM_{\{RC,NW\}} \leq 4,55$	18,8
$DM_{\{AC,SE\}} [m^3/s]$	6,17	$2,96 \leq DM_{\{AC,SE\}} \leq 8,66$	40,4
$DM_{\{IC,SE\}} [m^3/s]$	1,18	$0,43 \leq DM_{\{IC,SE\}} \leq 1,46$	23,7
$E [USD/kWh]$	0,158	$0,1342 \leq E \leq 0,2664$	15,1
$M [-]$	1.000	$7,22 \leq M$	99,3

7.1.5. Reflexiones parciales

La aplicación inicial del algoritmo en el problema propuesto por Gormaz-Cuevas et al. [75] permitió detectar inconvenientes que podrían generarse para el algoritmo según la formulación de los diferentes modelos. Particularmente, el problema presentaba múltiples soluciones óptimas, notándose los cambios al emplear diferentes valores iniciales en las variables para la resolución en GAMS. De esta forma, cualquier cambio en los parámetros generaba una nueva solución óptima. Es por ello que se requirieron simplificaciones detalladas en el Anexo F.2 para obtener soluciones óptimas únicas o que no cambiaran según el valor inicial.

La formulación está condicionada por la estrategia de resolución del problema de optimización multi-objetivo. Particularmente, el método de *programación por metas* requiere modificar las funciones objetivo para considerarlas como restricciones, añadiendo una función objetivo relacionada con una variable adicional (γ). Es por ello que también se detectó la necesidad de analizar si las funciones objetivo (replanteadas como restricciones) son relevantes para la solución, ya que el tipo de normalización en el método mencionado podría ocasionar que alguna de las funciones objetivo pueda variar libremente al estar condicionada la variable γ solamente por las otras. Si ese es el caso, como ocurría con la formulación original propuesta por Gormaz-Cuevas et al. [75], se generan diversas opciones de instalación para un mismo valor de los parámetros. Es por ello que la inclusión de las simplificaciones adicionales permitió reconocer los rangos e índices de estabilidad de los diversos parámetros, concordante con la información previa que se posee de cada uno.

7.2. Validación

La validación del algoritmo se realiza resolviendo el problema original para diferentes valores aleatorios de los parámetros, analizando si el comportamiento de las soluciones concuerda con lo esperado: los valores al interior del rango deben mantener la solución original, mientras que en el exterior se producen cambios en la solución. Específicamente, se utilizan 100 valores aleatorios de cada parámetro, que están en el entorno e interior del rango de estabilidad calculado y son diferentes a los empleados para estimar cada límite. Considerando que los valores de interés son los cercanos al valor nominal y a los límites determinados, debido a posibles cambios o errores en la estimación, el rango de generación de valores aleatorios se debe acotar. Particularmente, se decide analizar un rango de valores de tres veces la longitud del rango de estabilidad, estableciéndose según la distancia entre el valor nominal y cada límite, tal como se ilustra en la Figura 7.4. Para los parámetros en que existe información referencial que acota los posibles valores a adoptar, se ha considerado para restringir el rango de generación de valores aleatorios. Específicamente, esto ha influido en los valores menores al límite inferior de los parámetros, ya que todos deben ser no negativos.

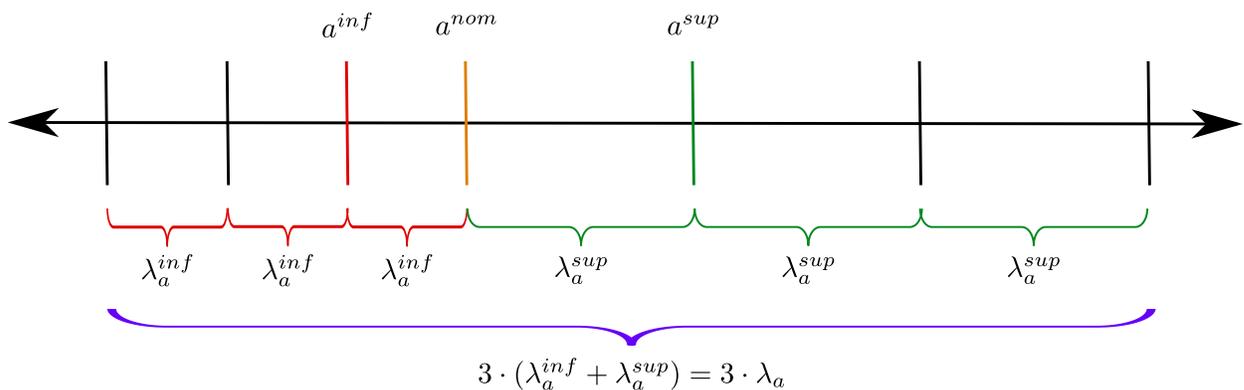


Figura 7.4: Rango de valores aleatorios generados para la validación del algoritmo. Se ejemplifica con un parámetro a , cuyo valor nominal es a^{nom} , el límite inferior es a^{inf} y el superior es a^{sup} . La longitud del rango de estabilidad (λ_a) se compone de la diferencia entre el valor nominal y el límite inferior (λ_a^{inf}) y la distancia entre el valor nominal y el límite superior (λ_a^{sup}).

La representación visual del comportamiento esperado se describió en el Capítulo 3, mientras que los resultados obtenidos para cada parámetro de este ejemplo se encuentran graficados en el Anexo F.7. En términos generales, las evaluaciones permitieron corroborar que cuatro parámetros ($DM_{\{RC,NW\}}$, $DM_{\{IC,SE\}}$, E y M) presentan el comportamiento ideal esperado, donde todos los valores al interior del rango mantienen la solución, mientras que aquellos fuera de este la modifican, tal como se ejemplifica mediante el parámetro $DM_{\{RC,NW\}}$ en la Figura 7.5. Adicionalmente, la mayoría de los valores al interior del rango se obtienen en un tiempo inferior al límite establecido ($\tau = 5$), determinándose soluciones óptimas. Por el contrario, la mayoría de los valores fuera del rango requieren un tiempo mayor al límite impuesto, por lo que la solución obtenida es *entera*, es decir, es la mejor solución encontrada en el tiempo de ejecución que contiene valores binarios. Para eventuales extensiones, los altos requerimientos de tiempo podrían ser una señal de que el valor se encuentra fuera del rango de estabilidad, por lo que se podrían explorar alternativas que analicen el tiempo que está tardando cada ejecución para estimar si se encuentra dentro o fuera del rango, lo que reduciría el tiempo total de implementación.

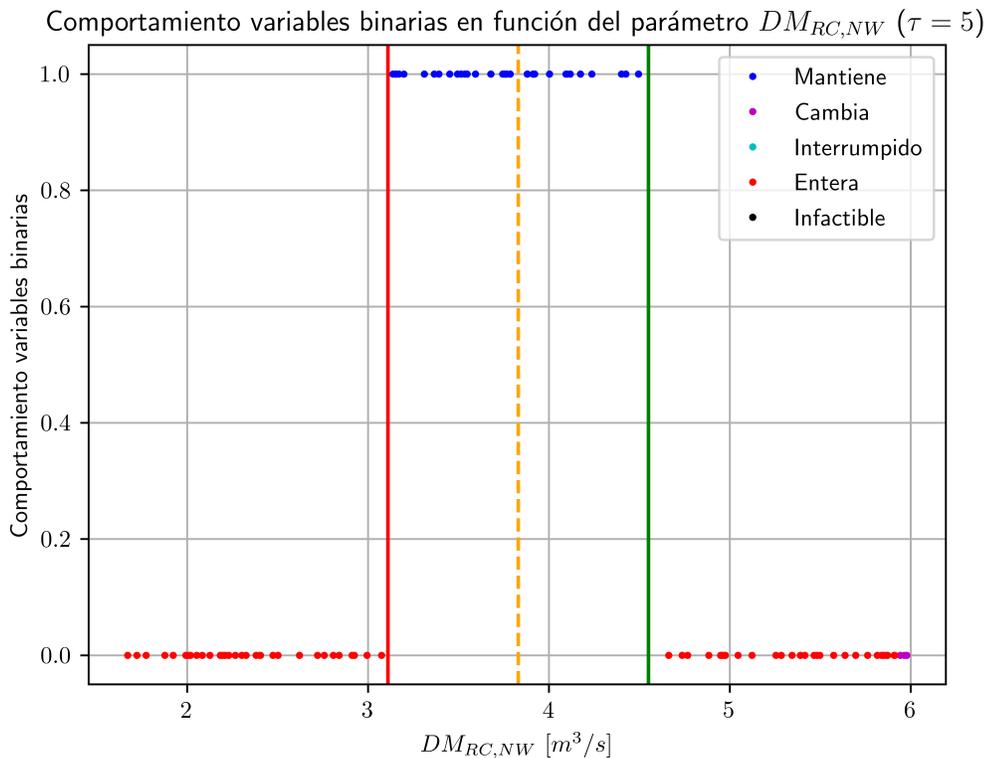


Figura 7.5: Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{RC,NW\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.

Respecto al quinto parámetro ($DM_{\{AC,SE\}}$), los valores al interior del rango mantienen la solución, tal como se esperaba. Sin embargo, existen evaluaciones al exterior que también pueden conservarlos, aunque con valores intermedios que originan un cambio en la solución, tal como se aprecia en la Figura 7.6 y se estimaba como un escenario factible. Considerando que el algoritmo no evalúa valores mayores cuando detecta una solución diferente, es posible que existan algunos que nuevamente permitan obtener la solución original. Sin embargo,

estos tendrían una región de estabilidad diferente, ya que se establecería en el entorno de otro valor nominal, distinto al escogido en el presente problema por no tratarse de un rango continuo. De esta forma, la prioridad del algoritmo es asegurar que los valores al interior del rango determinado mantienen la solución, mientras que los casos particulares en el exterior se escapan de la capacidad de determinación del algoritmo, ya que se debe evitar el riesgo de incluir valores intermedios que cambien la solución en los rangos de estabilidad calculados.

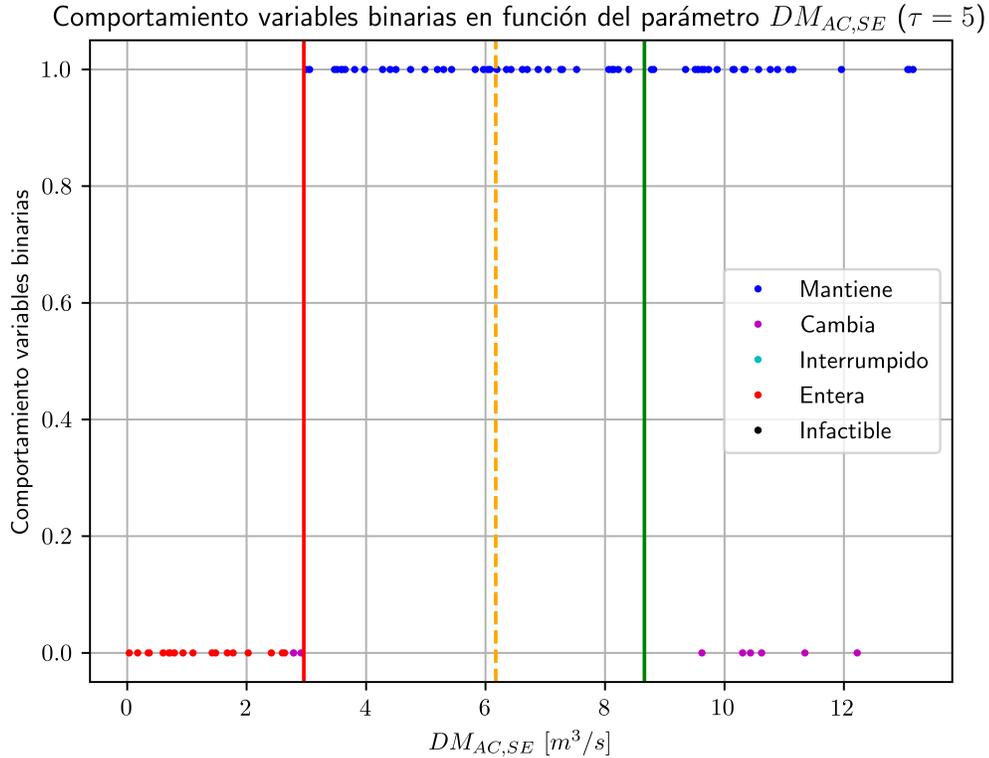


Figura 7.6: Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{AC,SE\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.

Previamente se ha reflexionado sobre el efecto del método de resolución en la estabilidad de la solución, particularmente en relación a *programación por metas*, que podría ser el origen de los comportamientos cambiantes al exterior del rango de estabilidad. Esta zona se escapa del análisis a realizar con el algoritmo propuesto, debido a que está relacionada con la formulación del problema. Sin embargo, si se estudiara en profundidad el problema y se evaluaran diferentes metodologías de resolución se podría conocer si el comportamiento se mantiene o cambia en la zona exterior del rango de $DM_{\{AC,SE\}}$, específicamente para los valores superiores a 9 e inferiores a 13. Uno de los enfoques para comenzar ese análisis podría ser estudiar el efecto de las normalizaciones y ponderadores en *programación por metas*, para luego probar otras formas de resolución.

A continuación, y tras haber validado el algoritmo a partir del comportamiento de la solución en los diferentes parámetros, se presentan las principales reflexiones de la estrategia propuesta, incluyendo los aspectos más relevantes de las discusiones parciales del trabajo y aquellas obtenidas mediante la validación del algoritmo.

7.3. Reflexiones del algoritmo y la validación

Los cuatro ejemplos estudiados han permitido observar que el rango de estabilidad es un indicador cuya interpretación requiere una comparación, ya sea con su valor nominal o con rangos de otros parámetros, ya que por sí solo no entrega suficiente información. Solo comparando un rango de estabilidad con otro es posible distinguir si uno de los parámetros es más estable que otro. Sin embargo, es complejo realizar comparaciones entre distintos parámetros directamente, ya que estos pueden presentar diferentes unidades y órdenes de magnitud. Es por ello que se propone el uso del índice de estabilidad, ya que emplea el rango de estabilidad y el valor escogido previamente para el parámetro, además de una normalización que permite comparar casos con distintas unidades y magnitudes.

Otro aspecto relevante del presente trabajo tiene relación con que el algoritmo busca destacar supuestos que deben ser analizados para mejorar la calidad del modelo, específicamente en cuanto a la verificación de la validez que poseen. Para los casos en que se conozca información detallada sobre los parámetros (por ejemplo, la distribución de probabilidad de los valores que pueden alcanzar), podría no ser necesario utilizar el algoritmo que se propone, ya que precisamente busca orientar sobre qué supuestos requieren ser analizados hasta alcanzar aquel nivel de detalle para la comprensión del comportamiento de los parámetros. De todas formas, si la información detallada no fue aplicada aún y se optó por un valor en específico para el parámetro, la estrategia permitiría señalar que el layout podría no ser óptimo si los valores de la distribución de probabilidad no están dentro del rango de estabilidad determinado.

La utilidad del nuevo algoritmo está ligada con la dificultad de modificar decisiones estratégicas por sobre las operacionales, teniendo como ejemplos la ubicación de plantas y la modificación de caudales, respectivamente. La decisión de instalar una planta puede ser más costosa que la conexión entre nodos, por lo que el enfoque se ha establecido priorizando el primer caso y su efecto en los costos de inversión. Sin embargo, si las conexiones son representadas mediante variables binarias, se ha observado con el tercer ejemplo de la tesis que es posible emplear también el algoritmo. De todas formas, si se analizaran casos en que las instalaciones de plantas y conexiones son representadas mediante variables binarias, podría ser conveniente priorizar la mantención de las primeras porque condicionan a las segundas, mientras que la implicancia inversa no es directa. Estos análisis se deben efectuar según cada contexto particular en que se enmarque el problema o proyecto a estudiar con el algoritmo.

El aumento en el tamaño de los problemas generó un incremento en el tiempo de resolución de cada uno y del requerido para la aplicación del algoritmo. Este comportamiento era esperado, aunque la implementación del algoritmo en los ejemplos tres y cuatro de la tesis ha permitido explicitar aquella situación y evidenciar la necesidad de que el algoritmo sea eficiente. Las propuestas realizadas en el nuevo algoritmo han permitido reducir el tiempo en contraste con el algoritmo referencial, aunque aún existe un margen de mejora. En particular, podría estudiarse detalladamente el fenómeno que se produce en las zonas exteriores del rango de estabilidad en la validación realizada, ya que los altos tiempos de resolución se generaban mayormente fuera del rango. Las resoluciones con alto tiempo podrían indicar que el valor estudiado es candidato a estar fuera del rango, aunque no es posible de asegurar en todos los casos con el análisis efectuado, ya que en las iteraciones realizadas en los ejemplos se observó

que algunos valores que estaban dentro del rango también requerían un elevado tiempo. Es por ello que se propone explorar esta alternativa, aunque con las precauciones relatadas.

La inclusión de límites de tiempo afecta a la solución en dos formas: si el tiempo límite es bajo, la mejor solución alcanzada en cada iteración podría no ser la óptima. Por el contrario, si el tiempo límite es alto, se podría requerir un elevado tiempo en obtener la solución. Es por ello que se recomienda no utilizar un ponderador τ igual a 1, ya que no se puede asegurar que cada iteración requiera el mismo tiempo que el problema original, siendo necesario incluir un margen de error. Por el contrario, podría ser prudente considerar tiempos de resolución del mismo o similar orden de magnitud para las resoluciones, por lo que se podría considerar un ponderador τ inferior o igual a 10. En particular, para los ejemplos estudiados se observó que con $\tau = 5$ se logra una reducción del tiempo de implementación del algoritmo sin afectar la solución. Específicamente, se logró reducir órdenes de magnitud del tiempo total de ejecución del algoritmo propuesto en el tercer ejemplo, en contraste con la versión referencial.

Otro de los factores relevantes en la eficiencia del algoritmo es la inclusión de información referencial, que permite analizar solamente el rango de valores que son de interés y que concuerdan con la realidad. Por ejemplo, si no se incluyen los límites referenciales, el algoritmo podría evaluar valores negativos en costos, flujos u otras expresiones que en la realidad no pueden alcanzar aquellos valores, pese a que matemáticamente pudiese ser posible.

Los límites referenciales no se restringen solamente a información lógica o física del contexto del problema. También pueden ser incorporados valores límite si el usuario está interesado en analizar el comportamiento de las variables binarias en una región determinada. Con ello, el algoritmo no requeriría evaluar valores que sobrepasen los límites impuestos. Se destaca que la inclusión de los límites referenciales genera regiones más estrechas que las factibles matemáticamente, aunque son requeridos por el enfoque práctico del algoritmo propuesto. Específicamente, los límites permiten reducir el número total de iteraciones al evitar valores que no son de interés y esta idea es posible de implementar también en el algoritmo referencial creado para casos MILP que estén vinculados con la realidad.

Las mayores dificultades para la implementación del algoritmo se presentan en modelos que incorporan cálculos previos sujetos a los parámetros en estudio y también en problemas con múltiples soluciones globales. El primer caso se presentó en el ejemplo 3, donde la inclusión de cotas dependientes de los valores de los parámetros podría generar inconvenientes al producir errores de redondeo en las estimaciones previas. Como los cálculos anteriores se realizan para acelerar la resolución del problema y no modifican la solución, se recomienda inicialmente descartar estos cálculos con el propósito de evitar errores en la implementación del algoritmo. Por otro lado, la segunda situación mencionada se observó en el ejemplo 4, que requirió nuevos supuestos para adaptar el problema y evitar las múltiples soluciones similares. Al tratarse de un problema multi-objetivo, se aconseja revisar la importancia relativa de las funciones objetivo en la solución, buscando evitar que el resultado del modelo varíe incluso al modificar los valores iniciales de las variables. En estas situaciones la solución original podría seguir siendo óptima al modificar alguno de los parámetros, pero el solver y software de optimización entregarían una solución óptima diferente, no permitiendo determinar el rango de estabilidad. Se destaca que ambas limitaciones están asociadas a la formulación del problema, por lo que se deben revisar antes de la aplicación del algoritmo.

El algoritmo propuesto puede ser aplicado en diferentes problemas MINLP y con distintos contextos. Tras establecerse las precauciones necesarias en base a las limitaciones mencionadas, se destaca que el mayor potencial del algoritmo se aprovecha al ser implementado en casos reales, ya sea basados en redes de agua u otros, porque permiten la inclusión de las cotas referenciales que mejoran la eficiencia del algoritmo. Sin esta información previa, el algoritmo generará valores factibles matemáticamente, pero que podrían no ser útiles o convenientes según el contexto a analizar.

Diferentes problemas y contextos podrían requerir otros solver. Para este trabajo se utilizó BARON, que fue el utilizado por las y los autores del caso de estudio principal. Sin embargo, es posible emplear otros dependiendo de las características del problema y los recursos computacionales de las y los usuarios. De todas formas, se destaca que la elección del solver está ligada al problema de optimización y no al algoritmo. Este último utilizará el mismo solver del problema original con el propósito de que los resultados y tiempos de ejecución sean comparables. Particularmente, para que sea posible utilizar un ponderador del tiempo máximo que tenga relación entre el desempeño del problema original y las diferentes iteraciones.

Finalmente, el trabajo se enfocó en realizar una primera aproximación para cambios de un parámetro a la vez. Una posible evolución de la investigación sería la realización de un análisis de sensibilidad global. Inicialmente, se podría explorar el efecto del cambio de más de un parámetro a la vez de forma iterativa. Por ejemplo, en las variaciones analizadas de los parámetros b y a del ejemplo 1, expresadas en las Figuras 4.2 y 4.3, se determinan límites del rango de estabilidad mediante la intersección de restricciones. Es por ello que una propuesta inicial sería variar ambos parámetros acercando las restricciones en que están involucrados. Los pasos necesarios para determinar eficientemente el rango de estabilidad para múltiples cambios en los parámetros se encuentra fuera del alcance del presente trabajo, aunque se destaca como posible extensión de la investigación. El desafío podría ser la identificación de una posible regla que genere el rango de estabilidad de un parámetro en función de los otros.

Capítulo 8

Conclusión

En el presente documento se ha identificado que no existen trabajos que permitan calcular la sensibilidad de una solución ante el cambio en los valores de parámetros en problemas MINLP. De esta forma, se presenta una estrategia para la determinación del rango de estabilidad de los parámetros de un problema MINLP, generando una primera aproximación para cubrir esta carencia y cumpliendo con el objetivo general de la tesis.

El algoritmo propuesto se basa en una versión creada para casos MILP por Yi y Lu [31], que ha sido mejorada con la implementación de nuevos pasos que permiten acelerar la búsqueda iterativa del rango de estabilidad, esta vez para casos MINLP. Las principales mejoras son: (i) el rango de búsqueda de los límites del rango de estabilidad se acota por información previa conocida sobre el parámetro; (ii) se estima un nuevo valor inicial para las iteraciones del algoritmo mediante un subproblema del original; (iii) se incorpora una cota en el tiempo de resolución de cada evaluación realizada por el algoritmo, basada en el tiempo requerido por el problema con los valores nominales de los parámetros; (iv) se incluye un índice de estabilidad como información de salida del algoritmo.

La comparación de los algoritmos mencionados se realizó mediante tres ejemplos de diferentes escalas, cuyas características se resumen en la Tabla 8.1, destacándose que el algoritmo propuesto logró una reducción del tiempo por sobre el 35 % en todos los problemas en relación al referencial. Con ello, se concluye que el algoritmo propuesto es más rápido en la determinación del rango de estabilidad tras aplicar las mejoras destacadas.

Tabla 8.1: Características y reducción del tiempo de implementación de los ejemplos utilizados para evaluar el desempeño de los algoritmos.

Ej.	Nº var. binarias	Nº ecuaciones	Tiempo alg. referencial [s]	Tiempo alg. propuesto [s]	Reducción de tiempo [%]
1	1	3	9,0	5,8	35,0
2	2	28	130,3	81,7	37,3
3	29	247	13.433,7	462,3	96,6

La incorporación de información previa de los parámetros permite considerar su interpretación en la realidad. Con ello, los valores analizados para determinar el rango de

estabilidad son los que efectivamente podría adoptar el parámetro. De esta forma, es posible acelerar la búsqueda y vincular el algoritmo con el contexto del problema o proyecto a desarrollar. Esta propuesta se podría asimilar a la mejora en el rendimiento de los solver al incluir cotas en las variables para determinar la solución de un problema de optimización.

La formulación de un subproblema del original para estimar un nuevo valor inicial del algoritmo también produce una disminución en el tiempo, basada en la reducción de las iteraciones necesarias. Específicamente, en aquellos casos en que se estima una nueva adivinanza inicial que mantiene la solución original, la búsqueda comienza en un valor más cercano al límite buscado, en contraste con iniciar las iteraciones desde el valor nominal.

La inclusión de un límite de tiempo por iteración es la mejora con mayor efecto en la velocidad del algoritmo en los problemas de mayor escala. Al evitar alcanzar el tiempo máximo por defecto de GAMS (1.000[s]) se logró una reducción de tiempo por sobre el 96 % en el tercer ejemplo, sin afectar la precisión de la solución. Específicamente, los rangos determinados por el algoritmo propuesto son más precisos, ya que se prioriza la detención por alcanzar la tolerancia definida y se disminuye la interacción con el usuario.

Se cumplió con el objetivo de crear un índice de estabilidad, que presenta ventajas para una priorización entre parámetros que sean necesarios de analizar con más profundidad, en contraste con el uso exclusivo del rango de estabilidad. En particular, permite comparar parámetros de distintas magnitudes e incluye el valor original del parámetro utilizado en la obtención de la solución óptima. Este índice es una de las principales novedades y se implementó en los cuatro ejemplos de esta tesis. En todos los casos fue posible determinar el parámetro crítico, es decir, aquel que se encuentra más cercano a alguno de los límites del rango de estabilidad, representando un mayor riesgo de cambiar la solución óptima si los supuestos y estimaciones realizadas sobre el parámetro crítico presentan errores.

La validación del algoritmo se realizó mediante la resolución del cuarto ejemplo y de escala de ciudad para 100 valores diferentes de los cinco parámetros estudiados. Todos los casos presentaron el comportamiento esperado, es decir, los valores al interior del rango calculado mantienen la solución, mientras que al exterior se producen cambios. Existen valores fuera del rango que generan la solución original, pero la existencia de valores intermedios que la modifiquen significa que pertenecen a un rango de estabilidad diferente al del valor nominal.

El algoritmo propuesto podría ser aplicado en otros casos MINLP. Se han utilizado las redes de agua como caso de estudio, aunque el algoritmo se ha propuesto de forma general y su mayor potencial estará en los problemas que tengan un contexto real asociado y cuyas variables de interés sean binarias. Con ello, la inclusión del conocimiento del usuario permitirá favorecer la búsqueda, tanto por los límites referenciales como tiempos de ejecución.

Se detectaron limitaciones asociadas a la formulación de los problemas analizados. Particularmente, los casos que presenten más de un óptimo global podrían afectar las conclusiones del algoritmo, ya que mediante el uso de diferentes valores iniciales se podría obtener una solución distinta a la original, pese a que también fuese óptimo mantenerlas por ser ambos (o más) casos soluciones óptimas globales. También aquellos problemas en que se incluyan cálculos previos a cada resolución para estrechar la región factible podrían

generar errores de redondeo que ocasionen inconvenientes en la resolución. Es por ello que se recomienda que aquellos cálculos opcionales se omitan al implementar el algoritmo.

Entre las posibles proyecciones, se propone continuar buscando alternativas para acelerar la determinación de los rangos de estabilidad. Específicamente, se sugiere analizar el efecto en el tiempo de resolución de las evaluaciones de valores fuera del rango de estabilidad, ya que aquellos casos son los que en su mayoría determinaron soluciones *enteras* tras alcanzar el tiempo límite en la etapa de validación. Con ello, se podrían definir criterios en relación a la tardanza de la resolución y estimar que son valores que se podrían encontrar, potencialmente, fuera del rango de estabilidad.

Otro aspecto a considerar en el trabajo futuro es explorar en cada contexto en que se aplique el algoritmo la relevancia de cada una de las variables del problema. Particularmente, en el algoritmo propuesto se escogen por defecto todas las variables binarias como aquellas que se desean mantener, aunque podrían existir criterios del usuario que prefieran unas sobre otras. Por ejemplo, situaciones en que algunas instalaciones sean más costosas o relevantes en la red. Podría ser interesante analizar casos en que se busque mantener variables continuas, aunque se presume que se requerirían otros pasos para ello en el algoritmo por la existencia de cambios infinitesimales en estas variables. Una propuesta para cubrir aquella situación es la inclusión de tolerancias permitidas en las variaciones en futuras evoluciones del algoritmo.

Finalmente, en este trabajo se ha realizado un análisis de sensibilidad *One-At-a-Time*, es decir, variando un parámetro a la vez. Esto no permite estudiar en profundidad la dependencia de los parámetros del problema entre sí. Es por ello que podría ser conveniente extender el estudio hacia escenarios con variaciones de múltiples parámetros en un análisis de sensibilidad global, aunque se advierte que podría requerir un alto costo computacional, por lo que se necesitarán nuevos pasos para acelerar la búsqueda de los rangos o regiones de estabilidad.

Glosario

1. CAE: Computer aided engineering o ingeniería asistida por computadora en español.
2. GSA: Global sensitivity analysis o análisis de sensibilidad global en español.
3. LP: Linear programming o programación lineal en español.
4. MILP: Mixed-integer linear programming o programación lineal mixta en español.
5. MINLP: Mixed-integer nonlinear programming o programación no lineal mixta en español.
6. MOO: Multi-objective optimization u optimización multi-objetivo en español.
7. NLP: Nonlinear programming o programación no lineal en español.
8. Rango o región de estabilidad: Corresponde al intervalo de valores que puede adoptar un parámetro sin que se cambie la solución óptima del problema.
9. RHS: Right hand side. Son los parámetros del vector de recursos de las restricciones.
10. WDN: Water distribution network o red de distribución de agua en español.

Bibliografía

- [1] Walker, P. y Lovat, T., “Towards a proportionist approach to moral decision making in medicine,” *Ethics and Medicine*, vol. 32, no. 3, pp. 153–161, 2016.
- [2] Wallace, K. y Burgess, S., “Methods and tools for decision making in engineering design,” *Design Studies*, vol. 16, no. 4, pp. 429–446, 1995.
- [3] Araújo, D., Davids, K., y Hristovski, R., “The ecological dynamics of decision making in sport,” *Psychology of Sport and Exercise*, vol. 7, no. 6, pp. 653–676, 2006.
- [4] Kurilovas, E., “On data-driven decision-making for quality education,” *Computers in Human Behavior*, no. September, pp. 1–9, 2018.
- [5] Goel, H., Herder, P., Weijnen, M., y Grievink, J., “Hierarchical approach for selecting uncertain parameters at the conceptual stage of design,” en *Computer Aided Chemical Engineering*, vol. 10, pp. 199–204, 2002, [doi:10.1016/S1570-7946\(02\)80061-X](https://doi.org/10.1016/S1570-7946(02)80061-X).
- [6] Vickers, W., “The balance between design and development,” en *Engineering Design*, pp. 47 – 60, Norwich: Pergamon Press Ltd., 1966.
- [7] Diani, M., Pievatolo, A., Colledani, M., y Lanzarone, E., “A comminution model with homogeneity and multiplication assumptions for the Waste Electrical and Electronic Equipment recycling industry,” *Journal of Cleaner Production*, vol. 211, pp. 665–678, 2019, [doi:10.1016/j.jclepro.2018.11.084](https://doi.org/10.1016/j.jclepro.2018.11.084).
- [8] Gerdtzen, Z., “Modeling metabolic networks for mammalian cell systems: General considerations, modeling strategies, and available tools,” *Advances in Biochemical Engineering/Biotechnology*, vol. 127, pp. 71–108, 2012.
- [9] Dindar, S., Kaewunruen, S., y An, M., “Rail accident analysis using large-scale investigations of train derailments on switches and crossings: Comparing the performances of a novel stochastic mathematical prediction and various assumptions,” *Engineering Failure Analysis*, vol. 103, no. April, pp. 203–216, 2019.
- [10] Wu, C. y Kopp, G., “Examination of the physical assumptions of a quasi-steady vector model using the integral momentum equation,” *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 187, no. January, pp. 73–84, 2019.
- [11] Filippi, C., “A fresh view on the tolerance approach to sensitivity analysis in linear programming,” *European Journal of Operational Research*, vol. 167, no. 1, pp. 1–19, 2005.
- [12] Shaw, M., “What Engineers Do,” en *Engineering Problem Solving: A Classical Perspective*, p. 450, 1 ed., 2001.
- [13] Yassine, A. y Chelst, K., “Opportunities for Decision Analysis in Engineering Management,” *IEEE Engineering Management Review*, vol. 46, no. 2, pp. 151 – 161,

2018.

- [14] Mamun, M. y Hansson, J., “Review and Challenges of Assumptions in Software Development,” Second Analytic Virtual Integration of Cyber-Physical Systems Workshop (AVICPS), no. September, 2011.
- [15] Benjamins, R., Fensel, D., y Straatman, R., “Assumptions of problem-solving Methods and their Role in Knowledge Engineering,” Proc. 12th International Conference on Artificial Intelligence, no. January 2000, pp. 1–16, 1996.
- [16] Pennock, M. y Wade, J., “The top 10 illusions of systems engineering: A research agenda,” Procedia Computer Science, vol. 44, no. C, pp. 147–154, 2015, <http://dx.doi.org/10.1016/j.procs.2015.03.033>.
- [17] Sen, R., “Underwater repair with composites,” en Marine Applications of Advanced Fibre-Reinforced Composites, pp. 259–278, Elsevier Ltd., 2016.
- [18] Alastruey, J., Xiao, N., Fok, H., Schaeffter, T., y Figueroa, C. A., “On the impact of modelling assumptions in multi-scale, subject-specific models of aortic haemodynamics,” Journal of The Royal Society Interface, vol. 13, p. 20160073, 2016, [doi:10.1098/rsif.2016.0073](https://doi.org/10.1098/rsif.2016.0073).
- [19] Ortiz-León, G., Araya-Luna, D., y Vílchez-Monge, M., “Revisión de modelos teóricos de la dinámica de fluidos asociada al flujo de sangre,” Revista Tecnología en Marcha, vol. 27, p. 66, 2014, [doi:10.18845/tm.v27i1.1697](https://doi.org/10.18845/tm.v27i1.1697).
- [20] Mahmoodi, S., Abbasi, M., y Sharifi, M., “New fluid flow model for hydraulic fractured wells with non-uniform fracture geometry and permeability,” Journal of Natural Gas Science and Engineering, vol. 68, no. November 2018, 2019.
- [21] Chaudhry, A., “Application of Fluid Flow Equations to Gas Systems,” en Gas Well Testing Handbook, pp. 11–83, 2003.
- [22] Carver, J., VanVoorhis, J., y Basili, V., “Understanding the impact of assumptions on experimental validity,” en Proceedings. 2004 International Symposium on Empirical Software Engineering, 2004. ISESE '04., pp. 251–260, IEEE, 2004, [doi:10.1109/ISESE.2004.1334912](https://doi.org/10.1109/ISESE.2004.1334912).
- [23] Note, M., “Selection and Prioritization,” en Project Management for Information Professionals, cap. 2, pp. 19–39, Chandos Publishing, 2016.
- [24] Bakshi, B., “The Quest for Sustainability : Challenges for Process Systems Engineering,” AIChE Journal, vol. 49, no. 6, pp. 1350 – 1358, 2003.
- [25] Solomon, S., Qin, D., Manning, M., Chen, Z., Marquis, M., y Averyt, K., “Climate Change 2007 : The Physical,” Cambridge: Cambridge University Press, 2007.
- [26] Misra, A., “Climate change and challenges of water and food security,” International Journal of Sustainable Built Environment, vol. 3, pp. 153–165, 2014.
- [27] Ursache, M., “Sustainable development between recent experiences and future challenges,” Procedia Economics and Finance, vol. 15, no. 14, pp. 1316–1323, 2014, [doi:10.1016/S2212-5671\(14\)00594-2](https://doi.org/10.1016/S2212-5671(14)00594-2).
- [28] Zhang, X., Liu, J., Zhao, X., Yang, H., Deng, X., Jiang, X., y Li, Y., “Linking physical water consumption with virtual water consumption: Methodology, application and implications,” Journal of Cleaner Production, vol. 228, pp. 1206–1217, 2019,

[doi:10.1016/j.jclepro.2019.04.297](https://doi.org/10.1016/j.jclepro.2019.04.297).

- [29] FAO, “Base de Datos Principal AQUASTAT, Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO),” 2016.
- [30] Luo, T., Young, R., y Reig, P., “Aqueduct Projected Water Stress Country Rankings. Technical Note.,” rep. tec., World Resources Institute, Washington, D.C., 2015.
- [31] Yi, C. y Lu, M., “Mixed-Integer Linear Programming-Based Sensitivity Analysis in Optimization of Temporary Haul Road Layout Design for Earthmoving Operations,” *Journal of Computing in Civil Engineering*, vol. 33, no. 3, pp. 1–14, 2019.
- [32] Brunaud, B. y Grossmann, I. E., “Perspectives in multilevel decision-making in the process industry,” *Frontiers of Engineering Management*, vol. 4, no. 3, p. 256, 2017, [doi:10.15302/j-fem-2017049](https://doi.org/10.15302/j-fem-2017049).
- [33] Kocis, G. y Grossmann, I., “Global Optimization of Nonconvex Mixed-Integer Nonlinear Programming (Minlp) Problems in Process Synthesis,” *Industrial and Engineering Chemistry Research*, vol. 27, no. 8, pp. 1407–1421, 1988.
- [34] Du, J. y Bormann, J., “Improved similarity measure in case-based reasoning with global sensitivity analysis: An example of construction quantity estimating,” *Journal of Computing in Civil Engineering*, vol. 28, no. 6, pp. 1–18, 2014.
- [35] Ferretti, F., Saltelli, A., y Tarantola, S., “Trends in sensitivity analysis practice in the last decade,” *Science of the Total Environment*, vol. 568, pp. 666–670, 2016.
- [36] Iooss, B. y Lemaître, P., “Uncertainty Management in Simulation-Optimization of Complex Systems,” *Operations Research/ Computer Science Interfaces Series*, vol. 59, pp. 101–122, 2015, [doi:10.1007/978-1-4899-7547-8](https://doi.org/10.1007/978-1-4899-7547-8).
- [37] Wagener, T. y Pianosi, F., “What has Global Sensitivity Analysis ever done for us? A systematic review to support scientific advancement and to inform policy-making in earth system modelling,” *Earth-Science Reviews*, vol. 194, pp. 1–18, 2019.
- [38] Aly, Z., Casagrande, A., Pastore, G., y Brown, N. R., “Variance-based sensitivity analysis applied to the hydrogen migration and redistribution model in Bison. Part II: Uncertainty quantification and optimization,” *Journal of Nuclear Materials*, vol. 523, pp. 478–489, 2019, [doi:10.1016/j.jnucmat.2019.06.023](https://doi.org/10.1016/j.jnucmat.2019.06.023).
- [39] Li, G. y Rabitz, H., “Relationship between sensitivity indices defined by variance- and covariance-based methods,” *Reliability Engineering and System Safety*, vol. 167, no. May, pp. 136–157, 2017.
- [40] Xiao, N., Alastruey, J., y Figueroa, C., “A systematic comparison between 1-D and 3-D hemodynamics in compliant arterial models,” *International Journal for Numerical Methods in Biomedical Engineering*, vol. 30, no. 2, pp. 204–231, 2013.
- [41] Liu, P., Liu, Y., Huang, Z., Cai, B., Sun, Q., Wei, X., y Xin, C., “Design optimization for subsea gate valve based on combined analyses of fluid characteristics and sensitivity,” *Journal of Petroleum Science and Engineering*, vol. 182, p. 106277, 2019, [doi:10.1016/j.petrol.2019.106277](https://doi.org/10.1016/j.petrol.2019.106277).
- [42] Floudas, C. A., “Introduction,” en *Nonlinear and Mixed-Integer Optimization*, p. 462, New York: Oxford University Press, 1995, [doi:10.1093/oso/9780195100563.003.0004](https://doi.org/10.1093/oso/9780195100563.003.0004).
- [43] Arenas, F., “Optimización del Proceso Kraft de Celulosa para minimizar la generación

- de Dregs y Grits,” Master Thesis, pp. 1–151, 2019.
- [44] Asenjo, J. A. y Andrews, B. A., “Protein purification using chromatography: Selection of type, modelling and optimization of operating conditions,” *Journal of Molecular Recognition*, vol. 22, no. 2, pp. 65–76, 2009.
- [45] Castillo, P. A. C., Castro, P. M., y Mahalec, V., “Global Optimization of Nonlinear Blend-Scheduling Problems,” *Engineering*, vol. 3, pp. 188–201, 2017, doi:10.1016/J.ENG.2017.02.005.
- [46] Grossmann, I., “Enterprise-wide optimization: A new frontier in process systems engineering,” *AIChE Journal*, vol. 51, no. 7, pp. 1846–1857, 2005.
- [47] Kusiak, A. y Heragu, S., “The facility layout problem,” *European Journal of Operational Research*, vol. 29, no. 3, pp. 229–251, 1987.
- [48] Latifi, S., Mohammadi, E., y Khakzad, N., “Process plant layout optimization with uncertainty and considering risk,” *Computers and Chemical Engineering*, vol. 106, pp. 224–242, 2017.
- [49] Faria, C. y Bagajewicz, M., “Pyrolysis of Heavy Oil in the Presence of Supercritical Water: The Reaction Kinetics in Different Phases,” *AIChE Journal*, vol. 61, no. 3, pp. 857–866, 2015.
- [50] Lienqueo, M. E., Salgado, J. C., Giaverini, O., y Asenjo, J. A., “Computer-aided design to select optimal polypeptide tags to assist the purification of recombinant proteins,” *Separation and Purification Technology*, vol. 65, no. 1, pp. 86–94, 2009.
- [51] Copado-Méndez, P., Guillén-Gosálbez, G., y Jiménez, L., “Rigorous computational methods for dimensionality reduction in multi-objective optimization,” *Computer Aided Chemical Engineering*, vol. 30, no. June, pp. 1292–1296, 2012, doi:10.1016/B978-0-444-59520-1.50117-2.
- [52] Vaskan, P., Guillén-Gosálbez, G., y Jiménez, L., “Multi-objective design of heat-exchanger networks considering several life cycle impacts using a rigorous MILP-based dimensionality reduction technique,” *Applied Energy*, vol. 98, pp. 149–161, 2012.
- [53] Girlikh, É., Kovalev, M., y Kravtsov, M., “Stability, pseudostability, and quasistability of multicriterion problem on a system of subsets,” *Cybernetics and System Analysis*, vol. 35, no. 5, pp. 777–788, 1999.
- [54] Sherali, H. y Driscoll, P., “Evolution and state-of-the-art in integer programming,” *Journal of Computational and Applied Mathematics*, vol. 124, no. 1-2, pp. 319–340, 2000.
- [55] Nowozin, S. y Jegelka, S., “Solution stability in linear programming relaxations: Graph partitioning and unsupervised learning,” *ACM International Conference Proceeding Series*, vol. 382, no. May, 2009.
- [56] Jansen, B., de Jong, J., Roos, C., y Terlaky, T., “Sensitivity analysis in linear programming: just be careful!,” *European Journal of Operational Research*, vol. 101, pp. 15–28, 1997, doi:10.1016/S0377-2217(96)00172-5.
- [57] Straus, J. y Skogestad, S., “Self-Optimizing Control in Chemical Recycle Processes,” *IFAC*, vol. 51, no. 18, pp. 536–541, 2018.
- [58] Björk, K. y Westerlund, T., “Global optimization of heat exchanger network synthesis

- problems with and without the isothermal mixing assumption,” *Computers and Chemical Engineering*, vol. 26, no. 11, pp. 1581–1593, 2002.
- [59] Krishnamoorthy, D., Thombre, M., Skogestad, S., y Jäschke, J., “Data-driven Scenario Selection for Multistage Robust Model Predictive Control,” *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 462–468, 2018, doi:10.1016/j.ifacol.2018.11.046.
- [60] Krishnamoorthy, D., Jahanshahi, E., y Skogestad, S., “Gas-lift Optimization by Controlling Marginal Gas-Oil Ratio using Transient Measurements,” *IFAC-PapersOnLine*, vol. 51, no. 8, pp. 19–24, 2018, doi:10.1016/j.ifacol.2018.06.349.
- [61] Fiacco, A. y Ishizuka, Y., “Sensitivity and stability analysis for nonlinear programming,” *Annals of Operations Research*, vol. 27, pp. 215–236, 1990.
- [62] Büskens, C. y Maurer, H., “Sensitivity Analysis and Real-Time Optimization of Parametric Nonlinear Programming Problems,” en *Online Optimization of Large Scale Systems*, pp. 3–16, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, doi:10.1007/978-3-662-04331-8_1.
- [63] Fiacco, A., *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, vol. 165. 1st ed., 1983, doi:10.1016/S0076-5392(08)60650-5.
- [64] Kronqvist, J., Bernal, D., Lundell, A., y Grossmann, I., *A review and comparison of solvers for convex MINLP*, vol. 20. Springer US, 2019, doi:10.1007/s11081-018-9411-8.
- [65] Sahinidis, N. V., “Mixed-integer nonlinear programming 2018,” *Optimization and Engineering*, vol. 20, no. 2, pp. 301–306, 2019, doi:10.1007/s11081-019-09438-1.
- [66] Caballero, J. y Ravagnani, M., “Water distribution networks optimization considering unknown flow directions and pipe diameters,” *Computers and Chemical Engineering*, vol. 127, pp. 41–48, 2019.
- [67] Jezowski, J., “Review of Water Network Design Methods with Literature Annotations,” *Ind. Eng. Chem. Res.*, vol. 49, no. 10, pp. 4475–4516, 2010.
- [68] Eck, B. y Mevissen, M., “Valve Placement in Water Networks: Mixed-Integer Non-Linear Optimization with Quadratic Pipe Friction Valve Placement in Water Networks: Mixed-Integer Non-Linear Optimization with Quadratic Pipe Friction,” *IBM Research Report*, vol. 25307, 2012.
- [69] Balaman, S., *Basics of Decision-Making in Design and Management of Biomass-Based Production Chains*. 2019, doi:10.1016/b978-0-12-814278-3.00006-6.
- [70] Wechsung, A., Oldenburg, J., Yu, J., y Polt, A., “Supporting chemical process design under uncertainty,” *Brazilian Journal of Chemical Engineering*, vol. 27, no. 3, pp. 451–460, 2010, doi:10.1590/S0104-66322010000300009.
- [71] Floudas, C., Aggarwal, A., y Ciric, A., “Global optimum search for nonconvex NLP and MINLP problems,” *Computers and Chemical Engineering*, vol. 13, no. 10, pp. 1117–1132, 1989, doi:10.1016/0098-1354(89)87016-4.
- [72] Karuppiah, R. y Grossmann, I., “Global optimization for the synthesis of integrated water systems in chemical processes,” *Computers and Chemical Engineering*, vol. 30, pp. 650–673, 2006.
- [73] Ahmetovic, E. y Grossmann, I., “Global Superstructure Optimization for the Design of Integrated Process Water Networks,” *AIChE Journal*, vol. 57, no. 2, pp. 434–457,

2010.

- [74] Gormaz-Cuevas, D., Riffo-Rivas, J., Montastruc, L., Brüning-González, M., y Díaz-Alvarado, F. A., “A multi-objective optimization model to plan city-scale water systems with economic and environmental objectives: A case study in Santiago, Chile,” *Journal of Cleaner Production*, vol. 279, 2021, doi:10.1016/j.jclepro.2020.123737.
- [75] Gormaz-Cuevas, D., Quiñones-López, C., Arenas-Araya, F., y Díaz-Alvarado, F. A., “Modeling Frictional Head Loss in the Redesign of City-Scale Water Networks through Multiobjective Optimization: Strategies to Deal with Nonlinear Terms,” *Industrial & Engineering Chemistry Research*, vol. 61, pp. 8918–8931, 2022, doi:10.1021/acs.iecr.2c00225.
- [76] Nikjoofer, A. y Zarghami, M., “Water Distribution Networks Designing by the Multiobjective Genetic Algorithm and Game Theory,” *Metaheuristics in Water, Geotechnical and Transport Engineering*, no. 1997, pp. 99–119, 2013.
- [77] Takama, N., Kuriyama, T., Shiroko, K., y Umeda, T., “Optimal water allocation in a petroleum refinery,” *Computers & Chemical Engineering*, vol. 4, pp. 251–258, 1980, doi:10.1016/0098-1354(80)85005-8.
- [78] Sieniutycz, S. y Jeżowski, J., “Approaches to water network design,” en *Energy Optimization in Process Systems and Fuel Cells*, cap. 20, pp. 641–681, Amsterdam: Elsevier, 3rd ed., 2018.
- [79] Bagajewicz, M., “A review of recent design procedures for water networks in refineries and process plants,” *Computers and Chemical Engineering*, vol. 24, no. 9-10, pp. 2093–2113, 2000.
- [80] Foo, D., “State-of-the-art review of pinch analysis techniques for Water network synthesis,” *Industrial and Engineering Chemistry Research*, vol. 48, no. 11, pp. 5125–5159, 2009.
- [81] Khor, C., Chachuat, B., y Shah, N., “Optimization of water network synthesis for single-site and continuous processes: Milestones, challenges, and future directions,” *Industrial and Engineering Chemistry Research*, vol. 53, no. 25, pp. 10257–10275, 2014.
- [82] Bhojwani, S., Topolski, K., Mukherjee, R., Sengupta, D., y El-Halwagi, M. M., “Technology review and data analysis for cost assessment of water treatment systems,” *Science of The Total Environment*, vol. 651, pp. 2749–2761, 2019, doi:10.1016/j.scitotenv.2018.09.363.
- [83] Doyle, S. J. y Smith, R., “Targeting water reuse with multiple contaminants,” *Process Safety and Environmental Protection*, vol. 75, no. 3, pp. 181–189, 1997.
- [84] Alva-Argáez, A., Kokossis, A. C., y Smith, R., “Wastewater minimisation of industrial systems using an integrated approach,” *Computers and Chemical Engineering*, vol. 22, no. SUPPL.1, pp. 741–744, 1998.
- [85] McCormick, G., “Computability of global solutions to factorable nonconvex programs: Part I - Convex underestimating problems,” *Mathematical Programming*, vol. 10, no. 1, pp. 147–175, 1976.
- [86] Berardi, L., Laucelli, D., y Giustolisi, O., “Accounting for local water storages in assessing WDN supply capacity,” *Procedia Engineering*, vol. 70, pp. 142–151, 2014.

- [87] Pepper, I. L., Choi, C. Y., y Gerba, C. P., “Microorganisms and Bioterrorism,” en *Environmental Microbiology*, pp. 565–574, Elsevier, 2º ed ed., 2009, [doi:10.1016/B978-0-12-370519-8.00028-6](https://doi.org/10.1016/B978-0-12-370519-8.00028-6).
- [88] Jose, N. y Sumam, K., “Optimal Water Distribution Network Design Accounting for Valve Closure,” *Procedia Technology*, vol. 24, pp. 332–338, 2016.
- [89] Rossi, M., Nigro, A., Pisaturo, G. R., y Renzi, M., “Technical and economic analysis of Pumps-as-Turbines (PaTs) used in an Italian Water Distribution Network (WDN) for electrical energy production,” *Energy Procedia*, vol. 158, pp. 117–122, 2019, [doi:10.1016/j.egypro.2019.01.055](https://doi.org/10.1016/j.egypro.2019.01.055).
- [90] Djebedjian, B., Mohamed, Mohamed, S., Mondy, A.-G., Rayan, M. A., y Petroleum, S., “Network optimization for steady flow and water hammer using genetic algorithms,” *International Water Technology Conference - IWTC*, no. May 2014, pp. 1099–1113, 2005.
- [91] GAMS Development Corporation, “General Algebraic Modeling System (GAMS) Release 28.2.0,” 2019.
- [92] Sahinidis, N., “BARON: A general purpose global optimization software package,” *Journal of Global Optimization*, vol. 8, no. 2, pp. 201–205, 1996.
- [93] Python Software Foundation, “Python Language Reference, Release 3.6.8,” 2018.
- [94] Sabio, N., Gadalla, M., Guillén-Gosálbez, G., y Jiménez, L., “Strategic planning with risk control of hydrogen supply chains for vehicle use under uncertainty in operating costs: A case study of Spain,” *International Journal of Hydrogen Energy*, vol. 35, no. 13, pp. 6836–6852, 2010.
- [95] WWAP (United Nations World Water Assessment Programme)/UN-Water, *The United Nations World Water Development Report 2018: Nature-Based Solutions for Water*. 2018.
- [96] Banco Mundial, “CHILE: Diagnóstico de la gestión de los recursos hídricos,” rep. tec., 2011.
- [97] Escenarios Hídricos 2030, “Radiografía del Agua: Brecha y Riesgo Hídrico en Chile,” rep. tec., Fundación Chile, Santiago, Chile, 2018.
- [98] Garreaud, R. D., Alvarez-Garretón, C., Barichivich, J., Boisier, J. P., Christie, D., Galleguillos, M., LeQuesne, C., McPhee, J., y Zambrano-Bigiarini, M., “The 2010–2015 megadrought in central Chile: impacts on regional hydroclimate and vegetation,” *Hydrology and Earth System Sciences*, vol. 21, pp. 6307–6327, 2017, [doi:10.5194/hess-21-6307-2017](https://doi.org/10.5194/hess-21-6307-2017).
- [99] DeNicola, E., Aburizaiza, O. S., Siddique, A., Khwaja, H., y Carpenter, D. O., “Climate change and water scarcity: The case of Saudi Arabia,” *Annals of Global Health*, vol. 81, no. 3, pp. 342–353, 2015, [doi:10.1016/j.aogh.2015.08.005](https://doi.org/10.1016/j.aogh.2015.08.005).
- [100] Escenarios Hídricos 2030, “Transición Hídrica: El Futuro del Agua en Chile,” rep. tec., Fundación Chile, Santiago, Chile, 2019.
- [101] He, T., Karimi, I. A., y Ju, Y., “Review on the design and optimization of natural gas liquefaction processes for onshore and offshore applications,” *Chemical Engineering Research and Design*, vol. 132, pp. 89–114, 2018, [doi:10.1016/j.cherd.2018.01.002](https://doi.org/10.1016/j.cherd.2018.01.002).

- [102] Merino, A., Acebes, L., Mazaeda, R., y de Prada, C., “Modelado y Simulación del Proceso de Producción del Azúcar,” *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 6, pp. 21–31, 2009, [doi:10.1016/S1697-7912\(09\)70261-4](https://doi.org/10.1016/S1697-7912(09)70261-4).
- [103] Díaz-Alvarado, F. y Gracia, F., “Steam reforming of ethanol for hydrogen production: Thermodynamic analysis including different carbon deposits representation,” *Chemical Engineering Journal*, vol. 165, no. 2, pp. 649–657, 2010.
- [104] Valenzuela-Venegas, G., Salgado, J. C., y Díaz-Alvarado, F., “Sustainability indicators for the assessment of eco-industrial parks: classification and criteria for selection,” *Journal of Cleaner Production*, vol. 133, pp. 99–116, 2016.
- [105] Caballero, A., *Métodos Sistemáticos para el Diseño, Síntesis, Planificación y Optimización de Procesos Químicos (CAPE)*. PhD thesis, Universidad de Alicante, 2008.
- [106] Díaz-Alvarado, F., *Reformado de Alcoholes para la producción de Hidrógeno: Análisis termodinámico y propuesta de mecanismos de reacción mediante grafos-P*. PhD thesis, Universidad de Chile, 2012.
- [107] Fernández, A., *Optimización matemática en procesos industriales. Aplicación al estudio de aparatos de tratamiento de alimentos por campos eléctricos*. PhD thesis, Universidad Politécnica de Madrid, 2014.
- [108] Biegler, L. T., Grossmann, I. E., y Westerberg, A., “Optimization approaches to process synthesis and design,” en *Systematic Methods of Chemical Process Design*, cap. 15, p. 796, New Jersey: Pearson Education, 1 ed., 1997.
- [109] Mosavi, A., “Optimization and Decision Making in Chemical Engineering Problems,” *Global Journal of Researches in Engineering*, vol. 10, no. 3, pp. 8 – 13, 2010.
- [110] Arora, J., “Multi-objective Optimum Design Concepts and Methods,” en *Introduction to Optimum Design (Third Edition)*, 2012.
- [111] Choi, S. y Manousiouthakis, V., “Global optimization methods for chemical process design: Deterministic and stochastic approaches,” *Korean J. Chem. Eng.*, vol. 19, no. 2, pp. 227–232, 2002.
- [112] Horst, R. y Tuy, H., *Global Optimization: Deterministic Approaches*. 2 ed., 1993.
- [113] Ryoo, H. y Sahinidis, N., “Global optimization of nonconvex NLPs and MINLPs with applications in process design,” *Computers and Chemical Engineering*, vol. 19, no. 5, pp. 551–566, 1995.
- [114] Vaidyanathan, R. y El-Halwagi, M., “Global optimization of nonconvex nonlinear programs via interval analysis,” *Computers and Chemical Engineering*, vol. 18, no. 10, pp. 889–897, 1994.
- [115] Kirkpatrick, S., Gelatt, C. D., y Vecchi, M. P., “Optimization by Simulated Annealing,” *Science*, vol. 220, pp. 671–680, 1983, [doi:10.1126/science.220.4598.671](https://doi.org/10.1126/science.220.4598.671).
- [116] Goldberg, D., *Genetic Algorithms in Search, Optimization, and Machine Learning*. 1989.
- [117] Sahinidis, N., “BARON User manual v.17.8.9,” rep. tec., 2017.
- [118] Muralikrishna, I. y Manickam, V., “Sustainable Development,” en *Environmental*

Management, cap. 2, pp. 5–21, 2017.

- [119] Juwana, I., Muttill, N., y Perera, B., “Indicator-based water sustainability assessment — A review,” *Science of The Total Environment*, vol. 438, pp. 357–371, 2012, [doi:10.1016/j.scitotenv.2012.08.093](https://doi.org/10.1016/j.scitotenv.2012.08.093).
- [120] Garbie, I., “An analytical technique to model and assess sustainable development index in manufacturing enterprises,” *International Journal of Production Research*, vol. 52, no. 16, pp. 4876–4915, 2014.
- [121] Garbie, I., “Optimizing Sustainability Indexes,” en *Sustainability in Manufacturing Enterprises*, pp. 191–201, 2016.
- [122] Zhu, L., Zhou, J., Cui, Z., y Liu, L., “A method for controlling enterprises access to an eco-industrial park,” *Science of the Total Environment*, vol. 408, no. 20, pp. 4817–4825, 2010, [doi:10.1016/j.scitotenv.2010.06.035](https://doi.org/10.1016/j.scitotenv.2010.06.035).
- [123] Drechsler, K., Kisekka, I., y Upadhyaya, S., “A comprehensive stress indicator for evaluating plant water status in almond trees,” *Agricultural Water Management*, vol. 216, no. September 2018, pp. 214–223, 2019.
- [124] Mao, F., Zhao, X., Ma, P., Chi, S., Richards, K., Clark, J., Hannah, D. M., y Krause, S., “Developing composite indicators for ecological water quality assessment based on network interactions and expert judgment,” *Environmental Modelling & Software*, vol. 115, pp. 51–62, 2019, [doi:10.1016/j.envsoft.2019.01.011](https://doi.org/10.1016/j.envsoft.2019.01.011).
- [125] Gómez-Llanos, E., Durán-Barroso, P., y Matías-Sánchez, A., “Management effectiveness assessment in wastewater treatment plants through a new water footprint indicator,” *Journal of Cleaner Production*, vol. 198, pp. 463–471, 2018.
- [126] Arenas-Araya, F., Vergara-Zambrano, J., y Díaz-Alvarado, F. A., “Design Water Networks through MINLP Formulation: An Algorithm to Find the Stability Region of Parameters,” *Industrial & Engineering Chemistry Research*, vol. 61, pp. 16112–16127, 2022, [doi:10.1021/acs.iecr.2c01542](https://doi.org/10.1021/acs.iecr.2c01542).

Anexo A

Situación hídrica de Chile frente al Cambio Climático

El crecimiento de la población, desarrollo económico y los cambios en los patrones de consumo de agua, entre otros factores, han generado que la extracción de agua a nivel mundial haya aumentado en alrededor de 1,8 % por año [29]. Se espera que durante las próximas dos décadas continúe creciendo la demanda global de agua, siendo la agricultura el sector que continúe con una mayor demanda por sobre el industrial y domiciliario [95].

En Chile, la distribución del uso de agua superficial y subterránea se resume en la Figura A.1 y se estima que la demanda para el año 2025 habrá aumentado [96]. Por otro lado, para el año 2040 se proyecta que Chile presente un estado de estrés hídrico extremadamente alto, siendo el único país latinoamericano en esa situación [30].

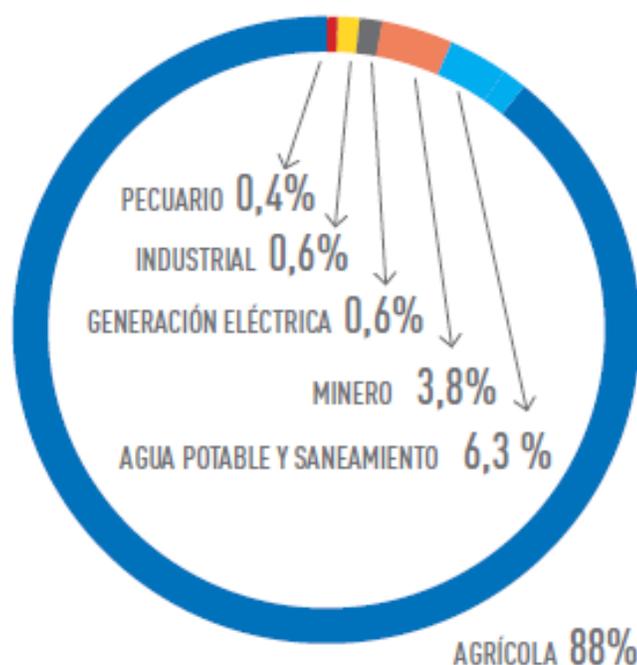


Figura A.1: Distribución del consumo de agua superficial y subterránea en Chile. Imagen obtenida de [97].

En la zona central del país se ha generado una disminución del caudal anual medio de los principales ríos, debido a un déficit de lluvia y nieve [98]. De esta forma, el Cambio Climático, que genera una menor disponibilidad de agua, provoca un estado de escasez hídrica que se irá acentuando ante los nuevos y crecientes requerimientos productivos [96].

Es importante entender que la escasez de agua no se trata solamente de problemas en la disponibilidad, sino que también considera la calidad del agua utilizable. Además del aumento de temperatura y períodos de sequía, se produce una degradación de las fuentes de agua a raíz de posibles eventos de precipitaciones extremas, desplazando agentes patógenos u otros contaminantes en los flujos de agua [99].

Para poder contrarrestar el problema de escasez de agua, se pueden adoptar distintas medidas. Actualmente, en Chile se ha creado la iniciativa *Escenarios Hídricos 2030* con el apoyo de distintas organizaciones, tanto públicas como privadas. Esta iniciativa propone “una transición hídrica para alcanzar la seguridad y sustentabilidad del vital recurso al año 2050, a través de Medidas, Acciones y Soluciones (MAS) que aborden la demanda, oferta, gestión e institucionalidad del agua para habilitar un desarrollo económico, social y ambiental, sostenible, basado en recursos hídricos suficientes en calidad y cantidad” [100]. Los ejes principales para lograr la transición se listan a continuación:

1. Gestión e institucionalidad del agua

Mecanismos institucionales, legales, educacionales y financieros que permiten la gestión del agua, colaboración y coordinación de las intervenciones sobre este recurso desde los distintos tipos de usuarios. Los temas principales en este eje son:

- Política Hídrica de Estado.
- Institucionalidad que lidera a nivel nacional.
- Nueva institucionalidad en cuencas.
- Gestión a nivel nacional y local.
- Condiciones regulatorias y económicas adecuadas.

2. Conservación y protección de nuestros ecosistemas hídricos

Medidas para cuidar los ecosistemas que son directamente fuentes de agua y también aquellos que permiten el funcionamiento de los primeros. Entre los principales temas asociados a este eje se encuentran:

- Resiliencia al Cambio Climático.
- Servicios ecosistémicos.
- Calidad del agua.
- Conservación de biodiversidad.
- Caudal ecológico

3. Eficiencia y uso estratégico del recurso hídrico

Medidas, acciones o soluciones asociadas a la demanda de agua para tener una mayor eficiencia y uso responsable, logrando así un equilibrio con la oferta disponible. Se centra principalmente en:

- Reducción del consumo y uso óptimo del agua.
- Manejo de la demanda de agua.
- Prioridad del uso a la vida y derecho humano al agua.
- Desarrollo diversificado.

4. Migración e incorporación de nuevas fuentes de agua

Son medidas, acciones o soluciones para lograr una mayor oferta de agua disponible para distintos fines. Los temas asociados a este eje son:

- Sistemas multipropósito de uso compartido, con economías de escala.
- Desacople del desarrollo productivo con el uso de agua en la cuenca.
- Incorporación de agua en las cuencas.
- Reducción de conflictos entre los usuarios.

Entre los ejes destacados, la presente tesis tiene una mayor relación con el tercero, debido a que se busca emplear eficientemente el agua. Específicamente, colaborar con el diseño óptimo de redes de agua. Este propósito se complementa con el cuarto eje, ya que se busca diversificar el origen del agua de cada consumidor, fomentándose en las redes óptimas analizadas el reciclaje del agua.

Anexo B

Antecedentes generales sobre optimización

Los antecedentes generales asociados a optimización de procesos, relacionados con este trabajo, se encuentran resumidos a continuación [43]:

La optimización de procesos se ha desarrollado con diferentes fines y en distintos contextos, configurando diseños que permitan disminuir el consumo energético, según la distribución de los equipos de la línea productiva o creando modelos matemáticos que representen el proceso para determinar condiciones de operación óptimas, abarcando sectores específicos o el proceso completo [101] [102].

B.1. Optimización matemática

El objetivo de la optimización matemática es obtener la mejor solución posible de un problema estudiado, buscando el mejor valor de las variables cuando se poseen grados de libertad positivos, apoyando la toma de decisiones. Para esto se plantean modelos que corresponden a una herramienta para representar la realidad, utilizando la optimización matemática tanto a escala molecular como industrial [103] [104].

La aplicación de la optimización en los procesos se realiza mediante tres etapas principales:

Representación de alternativas

Se plantea una superestructura, que corresponde a un diagrama de flujos que incluye todas las alternativas a evaluar, considerando flujos que podrían ser factibles u óptimos. La superestructura planteada dependerá de cada proceso, según el número de equipos y los flujos que los conectan [42] [105]. Por ejemplo, la producción en una planta manufacturera se puede representar a partir de nodos y arcos, para luego plantear matemáticamente la búsqueda del óptimo, es decir, aquel punto más cercano al ideal del proceso: alta productividad, baja emisión de residuos, entre otros [106].

Modelo matemático

Se crea un modelo de optimización representado matemáticamente de la siguiente forma:

$$\left\{ \begin{array}{l} \min_{x,y} \quad f(x, y) \\ \text{s.a.} \\ h(x, y) = 0 \\ g(x, y) \leq 0 \\ x \in X \subseteq \mathbb{R}^n \\ y \in Y = \{0, 1\}^l \end{array} \right. \quad (\text{B.1})$$

Donde:

- x es un vector con n variables continuas que pueden representar flujos, concentraciones, temperaturas, presiones, entre otras variables del proceso.
- y es un vector de l variables binarias que puede denotar la existencia ($y = 1$) o ausencia ($y = 0$) de una unidad del proceso en el diagrama de flujos óptimo. Por ejemplo, puede representar la inclusión o eliminación de un equipo, materia prima o producto del proceso.
- $f(x, y)$ es la función objetivo, que corresponde a una relación entre variables de decisión, cuyo mejor valor (mayor o menor) corresponde a la solución óptima del problema.
- $h(x, y)$ son las restricciones de igualdad, que pueden representar los balances de masa, energía, relaciones de equilibrio u otras expresiones similares que constituyen las restricciones del proceso.
- $g(x, y)$ corresponde a inequaciones que denotan especificaciones de diseño, restricciones de factibilidad o lógicas, entre otras [42] [107].

En el caso en que no existan variables y binarias (o enteras) y que $f(x), g(x)$ y $h(x)$ sean funciones lineales, se considera que los elementos presentes en la expresión B.1 componen un problema de programación lineal (LP). Si algunas de las funciones anteriores son no lineales, transforman el problema en uno de programación no lineal (NLP). Tanto para LP como para NLP existen versiones generalizadas en que se incorporan variables y binarias, denominándose problemas de programación lineal mixta (MILP) o problemas de programación no lineal mixta (MINLP), respectivamente [108].

Una de las expresiones más comunes y características de la Ingeniería de Procesos corresponde a los balances de masa de los equipos y sus especies, cuya naturaleza genera problemas no lineales, debido a los términos bilineales que los conforman: multiplicación entre flujos y concentraciones [73].

En los problemas de ingeniería es posible encontrar casos en que se requiere optimizar más de un objetivo a la vez, originando el planteamiento de problemas de optimización multiobjetivo (MOO⁷) [109]. Dependiendo de la estrategia de resolución escogida, se configura de distinta forma la función $f(x)$ del modelo descrito anteriormente [110].

⁷ Por su sigla en inglés, proveniente de *Multi-objective Optimization*.

Algoritmo de resolución

Aquellos problemas de naturaleza no lineal, como la mayoría de las formulaciones de optimización para la Ingeniería de Procesos, pueden presentar distintos óptimos locales [111]. Debido a ello, es recomendable utilizar algoritmos de optimización global, previniendo la convergencia a óptimos locales.

La mayoría de los algoritmos de optimización pueden ser clasificados como determinísticos o estocásticos. Ejemplos de algoritmos con enfoque determinísticos son: cutting planes [112], branch and bound [113] y análisis de intervalos [114]. Por otro lado, algoritmos de enfoque estocástico pueden ser algoritmos de simulación [115] y algoritmos genéticos [116], que poseen una alta probabilidad de encontrar el óptimo global, pero no lo garantizan.

Una de las principales herramientas para resolver problemas de optimización no lineal corresponde al solver BARON (*Branch-And-Reduce Optimization Navigator*), que es un sistema computacional que implementa algoritmos con enfoque determinístico para la solución global de problemas NLP y MINLP [117] [92]. Dadas las características de los problemas estudiados en el presente trabajo, este solver es el escogido para las diferentes implementaciones.

B.2. Indicadores de sustentabilidad

La optimización de procesos se puede realizar con distintos enfoques. En el caso del desarrollo sustentable, este se basa en el equilibrio entre necesidades u objetivos ambientales, sociales y económicos durante la toma de decisiones [118]. Uno de los mayores desafíos al respecto está relacionado con medir o cuantificar la sustentabilidad, razón por la que se han creado herramientas de evaluación basadas en indicadores o índices de sustentabilidad, que son utilizados como apoyo para los tomadores de decisiones o actores involucrados en el contexto respectivo [119].

Garbie [120] recopiló distintos índices que se emplean para medir la sustentabilidad desde cada uno de los enfoques que la componen. En términos económicos, se pueden destacar los costos, aumento de la población, tiempo de producción, etc.; dentro de los aspectos sociales se encuentran las condiciones de trabajo, empleabilidad, acceso a servicios esenciales, entre otros; y desde el punto de vista ambiental se puede medir la biodiversidad, contaminación, uso de suelo, etc. [120] [121].

Se destaca que la elección de indicadores se realiza tomando decisiones, según distintos criterios [122]. Este proceso puede requerir supuestos, que es el foco principal del presente trabajo.

B.3. Indicadores de sustentabilidad del agua

Al plantear un problema de optimización de redes de agua se pueden buscar distintos objetivos. En particular, la medición y cuantificación de la sustentabilidad del agua se realiza mediante indicadores. Juwana et al. [119] elaboraron un *review* sobre este tema, destacándose como ejemplos los siguientes: uso de agua (industrial, domiciliario o agrícola), estrés hídrico, calidad del agua, biodiversidad, disponibilidad de agua, entre otros.

Es posible emplear este tipo de indicadores para representar distintos criterios a considerar en un problema de optimización de redes de agua, con el propósito de cuantificar el impacto de las distintas soluciones que se podrían obtener. Para ello, se deben establecer las consideraciones y supuestos necesarios para definir los indicadores a emplear.

La elección del indicador o indicadores pertinentes en el planteamiento de un problema de optimización puede estar basada en distintos criterios. Valenzuela-Venegas et al. [104] proponen que se consideren 4 criterios principales, los que fueron definidos para Parques Eco-Industriales (PEI), adaptándose para el caso de una WDN según se propone a continuación:

- *Comprensión*: Un indicador debe ser fácil de entender.
- *Pragmatismo*: Un indicador debe ser medible, y su valor debe ser fácil de obtener.
- *Relevancia*: Un indicador debe ser relevante para cumplir con el objetivo principal de la red de distribución de agua y para el futuro de los consumidores y fuentes de agua.
- *Representación parcial de la sustentabilidad*: Un indicador debe representar adecuadamente una o más dimensiones de la sustentabilidad, permitiendo comparar entre distintas configuraciones de una red de agua o su evolución histórica.

Además de los criterios anteriores, la elección o el cálculo de indicadores de sustentabilidad de agua se efectúa realizando supuestos, tal como se ejemplifica en la Tabla B.1:

Tabla B.1: Ejemplos de supuestos empleados en la elección de indicadores para problemas de Ingeniería.

Supuesto (Área de estudio)	Explicación	Ref.
Estado estacionario (agricultura)	Para el desarrollo de un Indicador de Estrés Integral (CSI ^a) para plantas, se asume que no existe una variación en la temperatura de la hoja, estableciéndose que no cambia mayormente durante pequeños intervalos de tiempo.	[123]
Igualdad de cada importancia relativa (ecología del agua)	Para la construcción de indicadores compuestos de calidad de agua se consideran distintos elementos con una importancia relativa para cada uno de ellos. En caso de no tener claridad o sea dificultosa la comparación entre dos elementos, se asume que poseen la misma importancia relativa entre sí.	[124]
Concentración nula de contaminantes (Tratamiento de las aguas residuales)	En la creación de una huella de operación de aguas grises, se realizan balances de masa en los que se asume que la concentración de contaminante en el flujo es nula aguas arriba de la descarga.	[125]

^a Por su sigla en inglés, proveniente de *Comprehensive Stress Indicator*.

Anexo C

Ejemplo ilustrativo 1

C.1. Formulación

La formulación original del problema se observa en la expresión C.1:

$$\left\{ \begin{array}{ll} \min & z = 2x + y \\ x, y & \\ \text{s.a.} & \\ 1,25 - x^2 - y & \leq 0 \\ x + y & \leq 1,6 \\ x & \geq 0 \\ & y \in \{0, 1\} \end{array} \right. \quad (\text{C.1})$$

La adaptación realizada en la expresión C.2 destaca dos parámetros para ser analizados. Los valores nominales de a y b son 1,25 y 1,6, respectivamente.

$$\left\{ \begin{array}{ll} \min & z = 2x + y \\ x, y & \\ \text{s.a.} & \\ \mathbf{a} - x^2 - y & \leq 0 \\ x + y & \leq \mathbf{b} \\ x & \geq 0 \\ & y \in \{0, 1\} \end{array} \right. \quad (\text{C.2})$$

C.2. Solución óptima

Utilizando los valores nominales de los parámetros a y b , la solución del problema corresponde a $x = 0,5$, $y = 1$ y $z = 2$.

C.3. Rango de estabilidad

C.3.1. Parámetro a

El límite inferior del rango de estabilidad de a se determinó en la Tabla C.1, mientras que el superior se observa en la Tabla C.2.

Tabla C.1: Determinación del límite inferior del rango de estabilidad del parámetro a , con b constante.

i	a	$y' = y_0?$	a'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,25	Sí	–	–	0,26
1	0,25	Sí	–	–	0,26
2	–0,75	No	0,25	$0,1 > 0,0005$	0,08
Ciclo 2: $\Delta l = 0,1$					
0	0,25	Sí	–	–	0,24
1	0,15	No	0,25	$0,01 > 0,0005$	0,27
Ciclo 3: $\Delta l = 0,01$					
0	0,25	Sí	–	–	0,27
1	0,24	No	0,25	$0,001 > 0,0005$	0,27
Ciclo 4: $\Delta l = 0,001$					
0	0,25	Sí	–	–	0,26
1	0,249	No	0,25	$0,0001 < 0,0005$	0,30
Tiempo total:					2,21[s]

Tabla C.2: Determinación del límite superior del rango de estabilidad del parámetro a , con b constante.

i	a	$y' = y_0?$	a'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,25	Sí	–	–	0,38
1	2,25	No	1,25	$0,1 > 0,0005$	0,12
Ciclo 2: $\Delta l = 0,1$					
0	1,25	Sí	–	–	0,29
1	1,35	Sí	–	–	0,21
2	1,45	No	1,35	$0,01 > 0,0005$	0,11
Ciclo 3: $\Delta l = 0,01$					
0	1,35	Sí	–	–	0,28
1	1,36	Sí	–	–	0,23
2	1,37	No	1,36	$0,001 > 0,0005$	0,11
Ciclo 4: $\Delta l = 0,001$					
0	1,36	Sí	–	–	0,22
1	1,361	No	1,36	$0,0001 < 0,0005$	0,11
Tiempo total: 2,06[s]					

C.3.2. Parámetro b

El límite inferior del rango de estabilidad de b se observa en la Tabla C.3, mientras que las evaluaciones realizadas para el límite superior se encuentran en la Tabla C.4. Debido a que la variable binaria no cambia para valores superiores al caso nominal, no presenta un límite superior.

Tabla C.3: Determinación del límite inferior del rango de estabilidad del parámetro b , con a constante.

i	b	$y' = y_0?$	b'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,6	Sí	–	–	0,30
1	0,6	No	1,6	$0,1 > 0,00064$	0,07
Ciclo 2: $\Delta l = 0,1$					
0	1,6	Sí	–	–	0,28
1	1,5	Sí	–	–	0,31
2	1,4	No	1,5	$0,01 > 0,00064$	0,19
Ciclo 3: $\Delta l = 0,01$					
0	1,5	Sí	–	–	0,27
1	1,49	No	1,5	$0,001 > 0,00064$	0,14
Ciclo 4: $\Delta l = 0,001$					
0	1,5	Sí	–	–	0,29
1	1,499	No	1,5	$0,0001 < 0,00064$	0,13
Tiempo total: 1,98[s]					

Gráficamente, en el entorno del valor nominal de b , la variable binaria se comporta según se resume en la Figura C.1, concordando con los valores determinados para el rango de estabilidad. Se destaca que en este caso se realizaron 100 evaluaciones, tal como para el parámetro a . A diferencia de lo ocurrido para ese parámetro, este procedimiento permitió determinar que valores inferiores a 1,119 generan que la solución fuera infactible.

Tabla C.4: Determinación del límite superior del rango de estabilidad del parámetro b , con a constante.

i	b	$y' = y_0?$	b'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,6	Sí	—	—	0,29
1	2,6	Sí	—	—	0,27
2	3,6	Sí	—	—	0,27
3	4,6	Sí	—	—	0,28
4	5,6	Sí	—	—	0,25
5	6,6	Sí	—	—	0,26
6	7,6	Sí	—	—	0,29
7	8,6	Sí	—	—	0,26
8	9,6	Sí	—	—	0,26
9	10,6	Sí	—	—	0,31
Tiempo total: 2,74[s]					

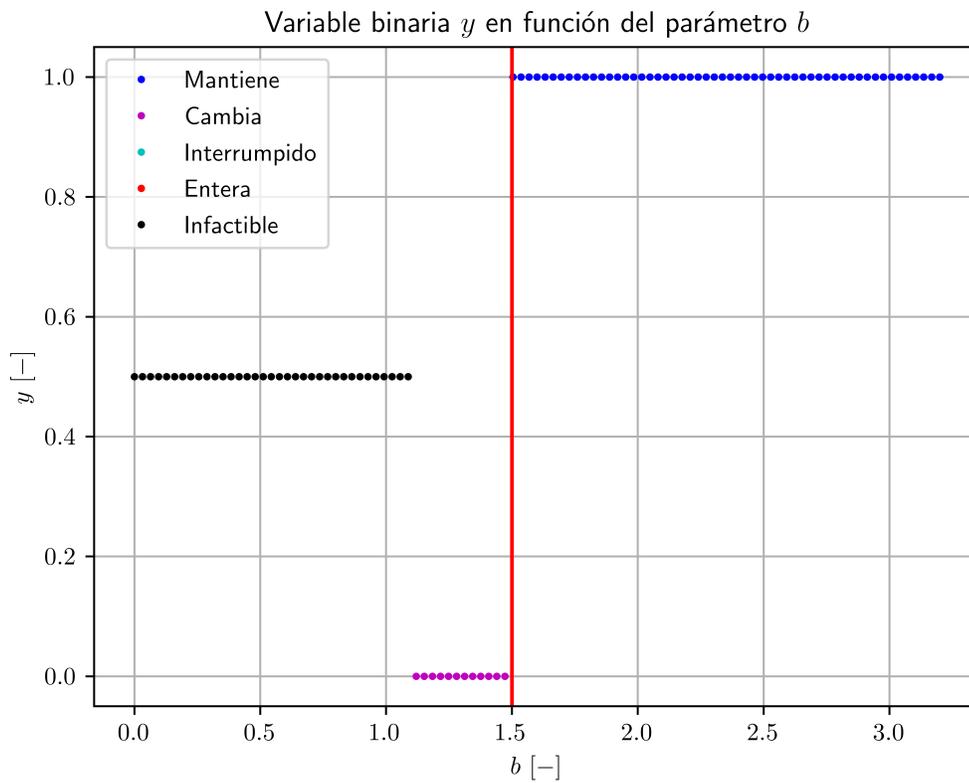


Figura C.1: Variación de la variable binaria y frente a cambios en el parámetro b . Línea roja: límite inferior calculado.

C.4. Efecto del tiempo de resolución

Con el propósito de analizar el tiempo requerido para las diferentes evaluaciones de los valores de los parámetros de interés en el presente ejemplo, a continuación se representará el tiempo utilizado en cada caso en relación al tiempo del problema original ($\tau = 1$).

C.4.1. Parámetro a

Para este primer parámetro, las múltiples resoluciones efectuadas para corroborar el comportamiento de la variable binaria según los cambios de a , resumidas en la Figura 4.1, requirieron diferentes tiempo de resolución. El tiempo empleado por cada evaluación se agrupa en la Figura C.2, notando que la mayoría requirió un tiempo inferior al nominal ($\tau < 1$), existiendo solamente 6 casos en que se necesitó un tiempo mayor, el que no superó las 5 veces el valor original.

Número de iteraciones de a resueltas según el tiempo de ejecución (ponderado por τ)

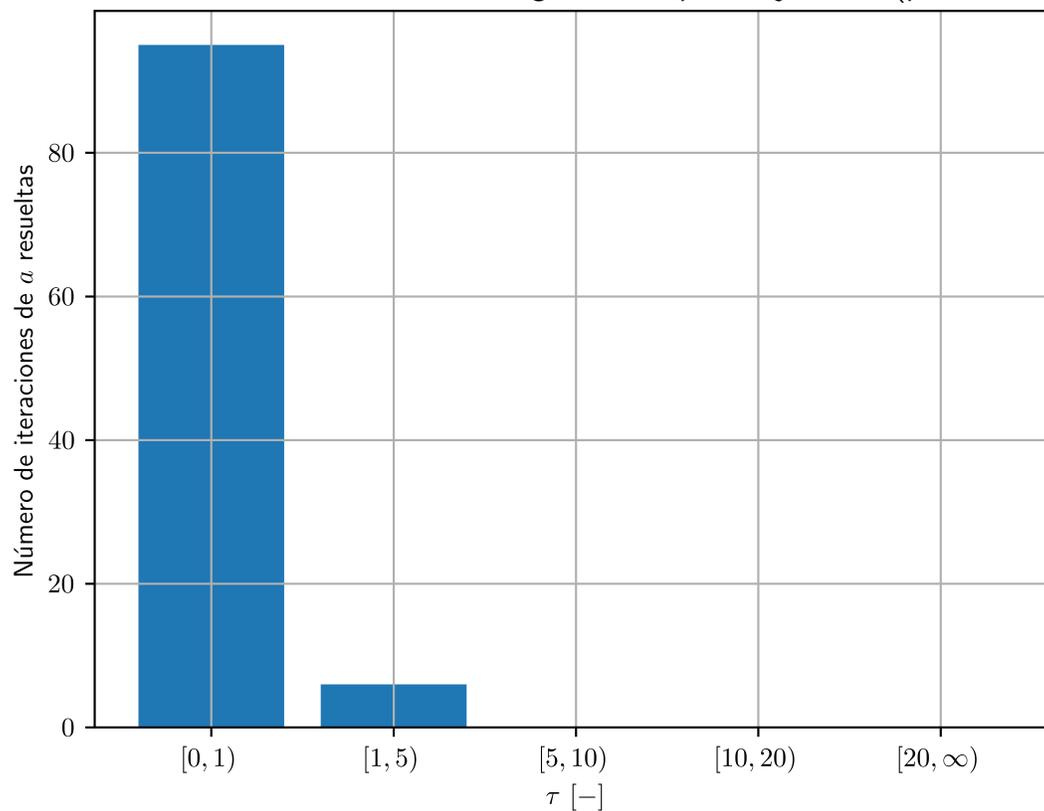


Figura C.2: Número de iteraciones de a resueltas según el tiempo de ejecución.

C.4.2. Parámetro b

Respecto a las múltiples resoluciones del ejemplo 1 modificando b , resumidas en la Figura C.1, estas requiriendo diferentes tiempos de ejecución, agrupándose según se observa en la Figura C.3. El gráfico permite notar que la mayoría necesitó un tiempo menor que el problema original, aunque se presentaron más casos que para a con un tiempo superior al nominal. De todas formas, si se aplicara un tiempo límite igual a 5 veces el nominal, se podrían haber realizado todas las iteraciones sin que se interrumpieran las ejecuciones.

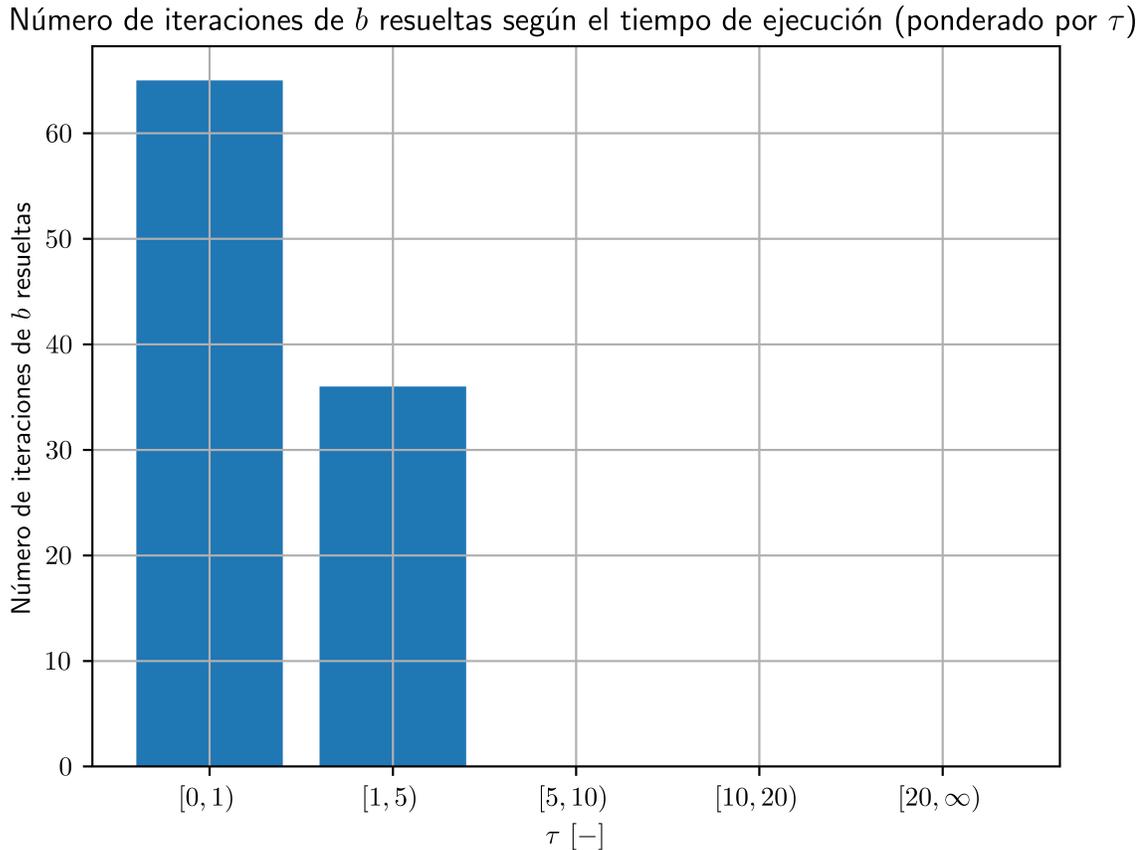


Figura C.3: Número de iteraciones de b resueltas según el tiempo de ejecución.

C.5. Análisis de inecuaciones

El estudio del efecto de los cambios en los parámetros sobre las restricciones del problema se realiza resolviendo el ejemplo 1 para distintos valores arbitrarios de ambos parámetros de interés. Para analizar las restricciones se estructuran de la forma $A \cdot x_s - b \leq 0$, según se observa en las ecuaciones C.3 y C.4, siendo aquellas en que se encuentran inmersos explícitamente los parámetros a y b , respectivamente.

$$Ec_1 : a - x^2 - y \leq 0 \quad (C.3)$$

$$Ec_2 : x + y - b \leq 0 \quad (C.4)$$

Las variaciones arbitrarias del parámetro a se encuentran en la Tabla C.5, donde se resumen sus valores característicos, correspondientes a los que produjeron cambios en la variable binaria y y el entorno de esos valores. Junto con ello, se incluyen los valores alcanzados en las restricciones C.3 y C.4.

Tabla C.5: Solución óptima y restricciones del ejemplo ilustrativo 1 frente a variaciones en el parámetro a , con b constante.

a	y	$a - x^2 - y$	$x + y - b$
1,500	0	0	-0,375
1,360	1	0	0
1,250	1	0	-0,100
1,000	1	0	-0,600
0,500	1	-0,50	-0,600
0,250	1	-0,75	-0,600
0,249	0	0	-1,100
0,100	0	0	-1,280

En la Tabla C.5 se observa que, variando solamente el parámetro a (b se mantiene constante), el valor de la variable binaria y se mantiene mientras $a \geq 0,25$, tal como se había determinado previamente. Para valores inferiores, y tiene un valor de 0, ocurriendo también que la expresión x^2 es igual a a , generando que la primera restricción se encuentre activa. Para complementar este análisis, las restricciones fueron graficadas en la Figura C.4, denominando como Ec_1 a la ecuación C.3 y Ec_2 a la ecuación C.4. Los valores utilizados corresponden a las soluciones obtenidas en las múltiples resoluciones que permitieron generar la Figura 4.1, mencionada anteriormente. Con ello, se observa cómo cambian las restricciones en función del parámetro a , mientras que el efecto de b se explicará más adelante.

Según la Figura C.4, tanto para definir el límite inferior como el superior del rango de estabilidad de a , se produce un cambio en las restricciones activas del problema al igualar o superar estos límites. Disminuyendo el parámetro a , al alcanzar el límite inferior la restricción Ec_1 se activa. Por el contrario, al aumentar a , en el límite superior las restricciones Ec_1 y Ec_2 se encuentran activas en simultáneo. Se destaca este aspecto para ser considerado en la creación del algoritmo a proponer en esta tesis.

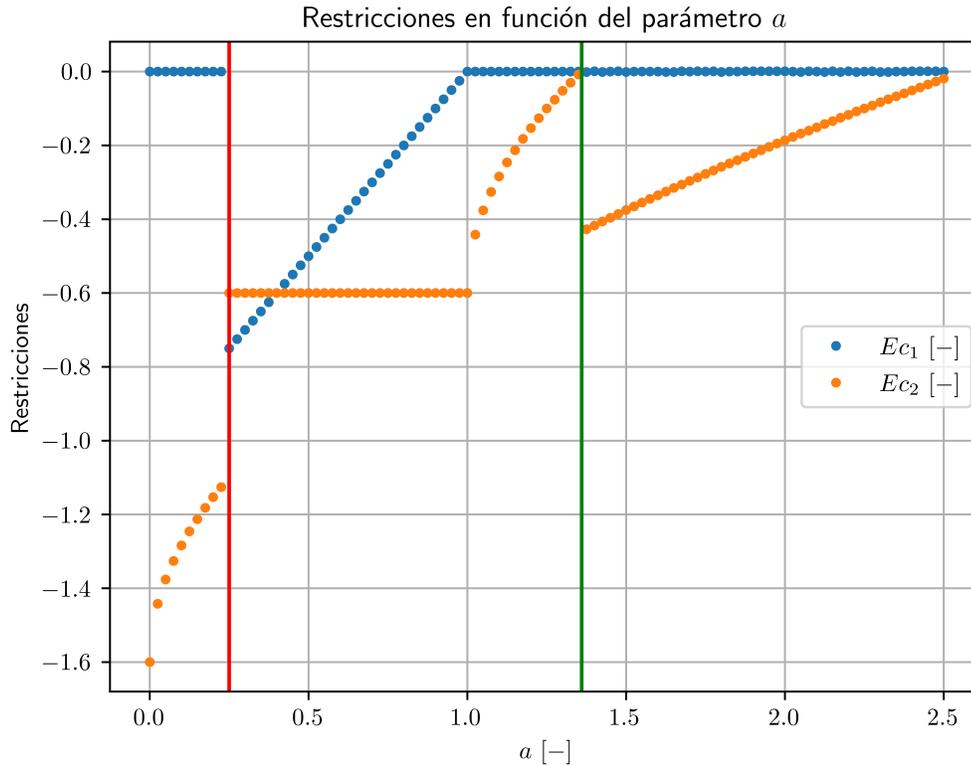


Figura C.4: Variación del valor de las restricciones frente a cambios en el parámetro a . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

Realizando cambios en el parámetro b se observa que al alcanzar el límite inferior del rango de estabilidad ($b = 1,5$) la segunda restricción está activa, tal como se destaca en la Tabla C.6 y la Figura C.5. Como la restricción en que se encuentra b no está activa para valores superiores al nominal (1,6), no se requiere una cota superior. Es decir, la segunda restricción no genera una cota para valores superiores de b , dado que el objetivo es minimizar el valor de x e y (la función objetivo es $2x + y$), por lo que no hay incentivo en que la suma de estas variables crezca. Este motivo explica que no haya existido un cambio en las variables binarias, incluso aunque se cambiaran los pasos de búsqueda y tolerancias con el algoritmo de Yi y Lu [31], destacándose que el análisis de las restricciones activas puede reflejar información importante para evitar múltiples resoluciones para concluir el rango de estabilidad.

Tabla C.6: Solución óptima y restricciones del ejemplo ilustrativo 1 frente a variaciones en el parámetro b , con a constante.

b	y	$a - x^2 - y$	$x + y - b$
10,000	1	0	-0,85
2,000	1	0	-0,50
1,600	1	0	-0,100
1,500	1	0	0
1,499	0	0	-0,38
1,450	0	0	-0,33
1,400	0	0	-0,28

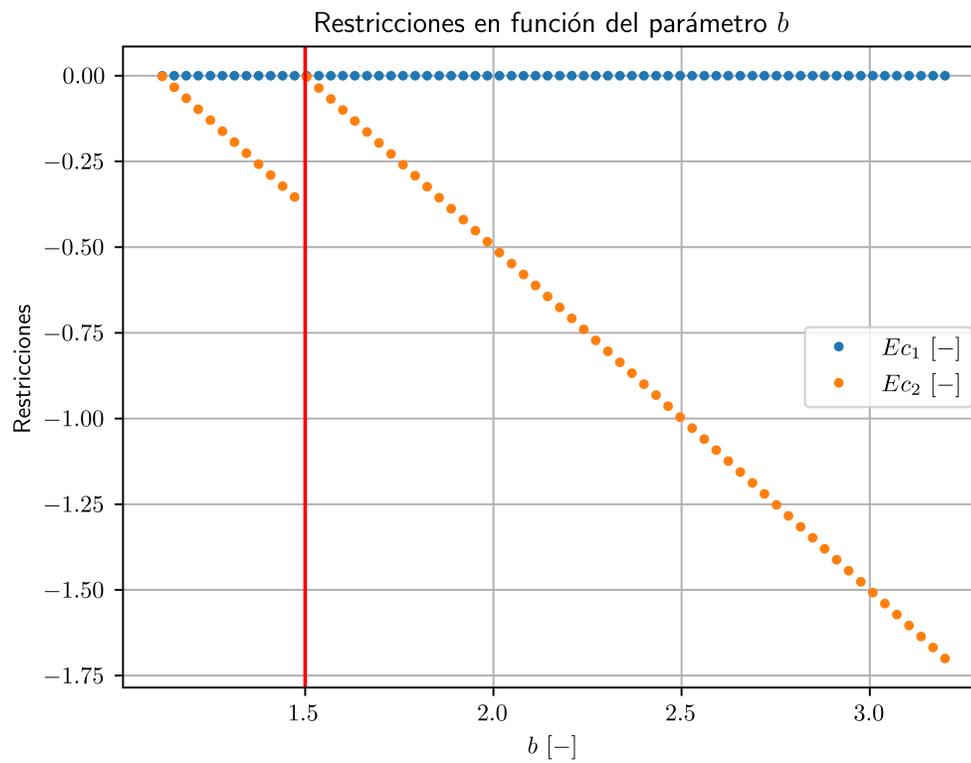


Figura C.5: Variación del valor de las restricciones frente a cambios en el parámetro b . Línea roja: límite inferior calculado.

C.6. Optimización de parámetros

Según se ha destacado anteriormente, el problema de optimización ha sido reformulado, intercambiando la naturaleza de algunos parámetros a variables y también en el sentido opuesto. Particularmente, en este primer ejemplo los parámetros a y b pasan a ser variables, mientras que la variable binaria y se considera en esta sección como un parámetro igual a 1, correspondiente a la solución original de esta variable.

La solución de cada subproblema para los parámetros a y b , el tiempo requerido para determinarlos, el utilizado en la evaluación de estos valores en el caso original y su posibilidad de uso como nuevo punto inicial se resume en la Tabla C.7:

Tabla C.7: Parámetros optimizados del ejemplo ilustrativo 1.

Parámetro optimizado [UN]	Valor obtenido	Tiempo ejec. reformulado [s]	Tiempo de evaluación [s]	¿Mantiene binaria?
a_{op}^{min} [-]	-10^{50}	0,09	0,08	No
a_{op}^{max} [-]	1,36	0,18	0,25	Sí
b_{op}^{min} [-]	1,50	0,09	0,33	Sí
b_{op}^{max} [-]	10^{50}	0,11	0,14	Sí

Se anunció que el nuevo punto inicial se utilizaría solamente en los casos en que el valor obtenido mantuviera el valor de las variables binarias. Es por ello que el valor optimizado podría ser empleado para buscar el límite superior de a , además del inferior y superior de b , mientras que se descarta su uso en la determinación del límite inferior de a .

La implementación de los valores obtenidos en esta sección se efectúa en el Anexo C.7, siendo parte del algoritmo propuesto.

C.7. Rango de estabilidad (algoritmo propuesto)

C.7.1. Parámetro a

Las iteraciones realizadas para determinar el límite inferior del rango de estabilidad de a se encuentran en la Tabla C.8, mientras que las evaluaciones para el cálculo del límite superior se observan en la Tabla C.9. Considerando los resultados de la *Optimización de parámetros*, para el límite inferior se comenzó la búsqueda desde el valor nominal de a , mientras que el proceso de determinación del límite superior considera el inicio de las iteraciones desde el valor optimizado, obtenido en el Anexo C.6.

Tabla C.8: Determinación del límite inferior del rango de estabilidad del parámetro a mediante el algoritmo propuesto, con b constante.

i	a	$y' = y_0?$	a'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,25	Sí	–	–	0,26
1	0,25	Sí	–	–	0,22
2	–0,75	No	0,25	$0,1 > 0,0005$	0,08
Ciclo 2: $\Delta l = 0,1$					
0	0,25	Sí	–	–	0,22
1	0,15	No	0,25	$0,01 > 0,0005$	0,25
Ciclo 3: $\Delta l = 0,01$					
0	0,25	Sí	–	–	0,23
1	0,24	No	0,25	$0,001 > 0,0005$	0,23
Ciclo 4: $\Delta l = 0,001$					
0	0,25	Sí	–	–	0,22
1	0,249	No	0,25	$0,0001 < 0,0005$	0,24
Tiempo iteraciones: 1,95[s]					
Tiempo total (iteraciones + opt. de parámetros): 2,12[s]					

En las Tablas C.8 y C.9 se ha destacado el tiempo empleado en las iteraciones y aquel que incluye el tiempo utilizado para determinar una nueva estimación inicial, proveniente del Anexo C.6 (opt. de parámetros). El tiempo total corresponderá al utilizado por el nuevo algoritmo y se empleará en la comparación con el algoritmo referencial.

A partir de los resultados destacados en esta sección, se concluye que el rango de estabilidad de a concuerda con lo obtenido mediante el algoritmo referencial en el Anexo C.3, según se resume en la Tabla C.10. Se observa también que la determinación del límite inferior necesitó un tiempo similar al utilizado por el algoritmo de Yi y Lu [31], mientras que para el límite superior se logró una disminución del 20% aproximadamente, favorecido por la utilización de una estimación inicial más cercana al límite buscado. En particular, el valor inicial correspondió al límite del rango de estabilidad, razón por la que el tiempo utilizado en las iteraciones se asocia principalmente a alcanzar la tolerancia previamente definida y así corroborar que se ha determinado el límite superior de a .

Tabla C.9: Determinación del límite superior del rango de estabilidad del parámetro a mediante el algoritmo propuesto, con b constante.

i	a	$y' = y_0?$	a'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,36	Sí	–	–	0,20
1	2,36	No	1,36	$0,1 > 0,0005$	0,10
Ciclo 2: $\Delta l = 0,1$					
0	1,36	Sí	–	–	0,20
1	1,46	No	1,36	$0,01 > 0,0005$	0,10
Ciclo 3: $\Delta l = 0,01$					
0	1,36	Sí	–	–	0,20
1	1,37	No	1,36	$0,001 > 0,0005$	0,11
Ciclo 4: $\Delta l = 0,001$					
0	1,36	Sí	–	–	0,21
1	1,361	No	1,36	$0,0001 < 0,0005$	0,10
Tiempo iteraciones: 1,22[s]					
Tiempo total (iteraciones + opt. de parámetros): 1,65[s]					

Tabla C.10: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de a .

Límite	Valor ref. [–]	Valor prop. [–]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	0,25	0,25	2,21	2,12	–4,1
Superior	1,36	1,36	2,06	1,65	–19,9

C.7.2. Parámetro b

El límite inferior del rango de estabilidad de b se calculó mediante el algoritmo propuesto y se obtuvieron los resultados de la Tabla C.11, observándose que las iteraciones comenzaron en el límite del rango de estabilidad, siendo necesarias para corroborar que se ha alcanzado la tolerancia preestablecida. En relación al límite superior, su valor se encuentra en la Tabla C.12, no siendo necesarias nuevas ejecuciones en este caso debido a los resultados de la *Optimización de parámetros* del Anexo C.6.

Tabla C.11: Determinación del límite inferior del rango de estabilidad del parámetro b mediante el algoritmo propuesto, con a constante.

i	b	$y' = y_0?$	b'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,5	Sí	–	–	0,27
1	0,5	No	1,6	$0,1 > 0,00064$	0,07
Ciclo 2: $\Delta l = 0,1$					
0	1,5	Sí	–	–	0,25
1	1,4	No	1,5	$0,01 > 0,00064$	0,10
Ciclo 3: $\Delta l = 0,01$					
0	1,5	Sí	–	–	0,25
1	1,49	No	1,5	$0,001 > 0,00064$	0,10
Ciclo 4: $\Delta l = 0,001$					
0	1,5	Sí	–	–	0,26
1	1,499	No	1,5	$0,0001 < 0,00064$	0,10
Tiempo iteraciones: 1,40[s]					
Tiempo total (iteraciones + opt. de parámetros): 1,82[s]					

Tabla C.12: Determinación del límite superior del rango de estabilidad del parámetro b mediante el algoritmo propuesto, con a constante.

i	b	$y' = y_0?$	b'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	10^{50}	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 0,25[s]					

Debido a que no se efectúan iteraciones para calcular el límite superior de b , el tiempo expresado en la Tabla C.12 corresponde al empleado en la obtención del nuevo valor inicial y su evaluación en el problema original. Particularmente, el valor obtenido es el máximo posible de expresar en GAMS, razón por la que se considera no acotado o cuyo valor máximo estaría por sobre 10^{50} en caso de existir. Como se ha discutido anteriormente, por la estructura del problema es posible asumir que aquel límite no existiría, concluyéndose que no estaría acotado superiormente el rango de estabilidad de b .

Contrastando los resultados del algoritmo propuesto en relación a lo obtenido anteriormente en el referencial, se observa que en ambos casos el límite inferior corresponde al mismo valor, tal como se expresa en la Tabla C.13. El procedimiento de *Optimización de parámetros* para el límite inferior permitió reducir el tiempo total en un 8,1 %.

Tabla C.13: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de b .

Límite	Valor ref. [-]	Valor prop. [-]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	1,5	1,5	1,98	1,82	-8,1
Superior	10,6 ^a	NA ^b	2,74	0,25	-90,9

^a Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

^b NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresado en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

En relación al límite superior, la disminución del tiempo se encuentra por sobre el 90 % debido a que el nuevo punto inicial es el máximo valor que es posible obtener en GAMS, concluyéndose que no está acotado o su límite estaría por sobre aquel valor. Por el contrario, en el algoritmo referencial se efectuaron diferentes evaluaciones hasta alcanzar el valor de 10,6, donde se sugiere al usuario actualizar los parámetros de búsqueda en caso de requerir continuar buscando el límite superior. Pese a que la estructura del problema permite concluir que no existirían cambios en las variables binarias para valores superiores, esta interacción y decisión del usuario se realizaría después de las evaluaciones que ocasionaron un mayor costo computacional, en contraste con la rapidez alcanzada por el algoritmo propuesto.

C.8. Índice de estabilidad

El rango de estabilidad determinado para cada parámetro en la sección C.7 se contrasta con su valor nominal correspondiente para obtener el *Índice de estabilidad*, según se propuso en el Capítulo 5, obteniéndose como índices de estabilidad inferiores y superiores los observados en la Tabla C.14:

Tabla C.14: Índice de estabilidad inferior y superior de los parámetros del ejemplo 1.

Par. [UN]	Nominal	Rango de estabilidad	IE^{inf} [%]	IE^{sup} [%]
a [-]	1,25	$0,25 \leq a \leq 1,36$	80	8,8
b [-]	1,60	$1,50 \leq b$	6,3	NA ^a

^a NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresable en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

Considerando que el índice de estabilidad corresponde al menor valor entre el índice superior e inferior, se observa que el menor índice lo presenta el parámetro b con un valor de 6,3%. Particularmente, este parámetro no se encuentra acotado superiormente o su cota estaría por sobre el valor máximo posible de expresar. Sin embargo, la cercanía entre el valor nominal y el límite inferior origina que se deba analizar más detalladamente su estimación para evitar que la solución óptima fuese otra realmente, en caso de que representara un problema con contexto definido, como ocurre con los ejemplos 2 y 3.

Anexo D

Ejemplo ilustrativo 2

D.1. Formulación

D.1.1. Conjuntos

- $I = \{P_1, P_2\}$: Conjunto de plantas de procesos.
- $J = \{T_1, T_2\}$: Conjunto de plantas de tratamiento.
- $K = \{N, P_1, P_2, T_1, T_2\}$: Conjunto de nodos que poseen flujos de salida.
- $L = \{P_1, P_2, T_1, T_2, S\}$: Conjunto de nodos que poseen flujos de entrada.

D.1.2. Parámetros

Parámetros operacionales

- D_i : Demanda de agua de la planta de procesos $i \in I$.
- Cap_j : Capacidad de la planta de tratamiento $j \in J$.
- r_j : Porcentaje de remoción de contaminante en la planta de tratamiento $j \in J$.
- $Cont_i$: Flujo másico de contaminante inyectado en la planta de procesos $i \in I$.
- Cal_i : Calidad o concentración máxima de contaminante permitida en la planta de procesos $i \in I$.
- C_N : Concentración de contaminante en la fuente (N).

Parámetros económicos

- $Cost_j$: Costo de instalación de la planta de tratamiento $j \in J$.

Otros

- M : Parámetro al que se le asigna un valor alto, utilizado para restricciones del tipo *Big M*.
- ϵ : Parámetro incluido para evitar soluciones no acotadas por denominadores nulos al modificar los parámetros.

Los valores nominales de los parámetros se encuentran en la Tabla D.1.

Tabla D.1: Valores nominales de los parámetros del ejemplo ilustrativo 2.

Parámetro	Valor	Unidad	Parámetro	Valor	Unidad
D_{P_1}	50	[t/h]	Cap_{T_1}	60	[t/h]
D_{P_2}	40	[t/h]	Cap_{T_2}	80	[t/h]
Cal_{P_1}	30	[ppm]	r_{T_1}	5	[%]
Cal_{P_2}	20	[ppm]	r_{T_2}	10	[%]
$Cont_{P_1}$	1	[kg/h]	$Cost_{T_1}$	1.000.000	[USD\$]
$Cont_{P_2}$	1,5	[kg/h]	$Cost_{T_2}$	1.100.000	[USD\$]
C_N	0	[ppm]	M	500	[-]
ϵ	10^{-10}	[-]			

D.1.3. Variables

Las variables presentes son clasificadas según su naturaleza:

Continuas y positivas

- $F_{k,l}$: Flujo proveniente del nodo $k \in K$ que ingresa al nodo $l \in L$, con $k \neq l$.
- C_k : Concentración de salida del nodo $k \in K \setminus \{N\}$.

Binarias

- y_j : Variable binaria asociada a la existencia ($y_j = 1$) o ausencia ($y_j = 0$) de la planta de tratamiento $j \in J$.

D.1.4. Restricciones

Las restricciones consideradas en este problema son las siguientes:

Balances de masa

- Balance global:

Considerando la superestructura del problema, el balance de masa global incluye los flujos provenientes de la fuente (N) hacia las plantas de procesos (P_1 y P_2) y aquellos que terminan en el sumidero (S) desde las plantas de tratamiento (T_1 y T_2).

$$\sum_{i \in I} F_{N,i} = \sum_{j \in J} F_{j,S} \quad (D.1)$$

- Balance por planta:

Asumiendo que no existe acumulación en cada planta, el flujo másico total que ingresa es igual al de salida. En cada planta de procesos $i \in I$, se cumple lo siguiente:

$$\sum_{k \in K} F_{k,i} = \sum_{l \in L \setminus \{S\}} F_{i,l} \quad \forall i \in I \quad (D.2)$$

Para el caso de las plantas de tratamiento $j \in J$ se considera la siguiente expresión y flujos:

$$\sum_{i \in I} F_{i,j} = \sum_{i \in I \cup \{S\}} F_{j,i} \quad \forall j \in J \quad (\text{D.3})$$

- Balance por especie:

Como la concentración de cada flujo se asume igual para un mismo origen, la expresión del balance de masa por especies se estructura de la siguiente forma para las plantas de procesos $i \in I$, incluyendo el flujo másico de contaminante ($Cont_i$) adicionado en cada una de estas:

$$\sum_{k \in K} F_{k,i} \cdot C_k + Cont_i \cdot 1.000 \cdot \left(\frac{D_{P_i}}{D_{P_i} + \epsilon} \right) = \sum_{l \in L \setminus \{S\}} F_{i,l} \cdot C_l \quad \forall i \in I \quad (\text{D.4})$$

En la expresión anterior se ha multiplicado el flujo de contaminante por un factor de 1.000 para que la ecuación sea dimensionalmente correcta [72]. Adicionalmente, esa expresión se ha multiplicado por un término dependiente de la demanda, ya que la inclusión de contaminante en el sistema requiere que exista la demanda de agua de la planta correspondiente. El planteamiento escogido permite no incluir variables binarias adicionales en el problema asociadas a esta condición.

Para el caso de las plantas de tratamiento, se debe considerar la fracción de contaminante que es removida (r_j) en cada una de estas:

$$\left(\sum_{i \in I} F_{i,j} \cdot C_i \right) \cdot (1 - r_j) = \sum_{i \in I \cup \{S\}} F_{j,i} \cdot C_j \quad \forall j \in J \quad (\text{D.5})$$

Operacionales

- Demanda de agua:

$$\sum_{k \in K} F_{k,i} = D_i \quad \forall i \in I \quad (\text{D.6})$$

- Concentración máxima de contaminante:

Debido a los requerimientos de cada planta de procesos, se considera una concentración máxima para el flujo total de entrada (Cal_i). Esta restricción se expresa a continuación:

$$\sum_{k \in K} F_{k,i} \cdot C_k \leq \sum_{k \in K} F_{k,i} \cdot Cal_i \quad \forall i \in I \quad (\text{D.7})$$

- Instalación de plantas:

Para que existan flujos desde (y hacia) una planta de tratamiento es necesario que se encuentre instalada. De esta forma, se utiliza una restricción del tipo *Big M* para relacionar los flujos y la existencia de las plantas:

$$\sum_{i \in I} F_{i,j} + \sum_{i \in I \cup \{S\}} F_{j,i} \leq y_j \cdot M \quad \forall j \in J \quad (\text{D.8})$$

- Capacidad de tratamiento:

Las plantas de tratamiento poseen un flujo límite a tratar (independiente de la concentración de ese flujo), estableciéndose esta restricción de la siguiente forma:

$$\sum_{i \in I} F_{i,j} \leq Cap_j \quad \forall j \in J \quad (\text{D.9})$$

D.1.5. Funciones objetivo

La función objetivo se encuentra estructurada según la formulación de sumas ponderadas, considerando dos funciones objetivo principales. La primera posee un enfoque ambiental y corresponde a la minimización del uso de agua proveniente de la fuente (N), que se encuentra normalizada por la demanda completa de las plantas de procesos. La segunda es económica, asociada al costo de instalación de las plantas de tratamiento y está normalizada para el caso en que se instalaran ambas opciones.

$$F.O. = \left(\frac{F_{N,P_1} + F_{N,P_2}}{D_{P_1} + D_{P_2} + \epsilon} \right) + \left(\frac{Cost_{T_1} \cdot y_{T_1} + Cost_{T_2} \cdot y_{T_2}}{Cost_{T_1} + Cost_{T_2} + \epsilon} \right) \quad (\text{D.10})$$

Se ha incluido en los denominadores el término ϵ para evitar inconvenientes asociados a los cambios en los costos y demandas. Específicamente, si en una expansión del análisis del rango de estabilidad se decidiera modificar más de un parámetro a la vez, podría ocurrir que el denominador fuese nulo, por lo que el término de holgura ϵ evitaría los inconvenientes numéricos asociados.

D.1.6. Supuestos del modelo

Los supuestos presentes en la formulación del ejemplo ilustrativo 2 se resumen en la Tabla D.2. Aquellos que influyen en los parámetros del problema son discutidos en el presente trabajo, mientras que el resto se incorpora para evidenciar la existencia de diferentes supuestos en la formulación de problemas de redes de agua.

Tabla D.2: Supuestos empleados en el ejemplo ilustrativo 2.

Elemento				Supuesto	Explicación
P	V	R	O		
X				Demanda conocida y constante	Se asume que la demanda de agua de las plantas de procesos no varía.
X				Costo conocido y constante	Se considera que el costo de instalar cada planta de tratamiento no cambia.
X				Contaminación es constante	Se asume que el flujo de contaminante que se inyecta en cada planta de procesos no cambia, independiente de la calidad (concentración) de los flujos de agua que ingresan en estas.
	X			Posibilidad de compartir flujos entre plantas de procesos	Se considera que los flujos del problema no presentan una disposición lineal, si no que pueden ser compartidos entre plantas de procesos o pueden regresar desde plantas de tratamiento, agregando variables en el sistema.
	X			Plantas de tratamiento no comparten flujos entre sí	Se asume que no es posible enviar agua entre plantas de tratamiento, lo que imposibilita que exista una remoción aún mayor de los flujos tratados.
	X			Concentración de todos los flujos de salida de un nodo es igual	Se asume que la concentración de salida de un nodo es igual para cada uno de los flujos que provienen de este.
		X		Estado estacionario	Se considera que no existe acumulación en las plantas, por lo que se desestiman las variaciones de los flujos en el tiempo.
		X		Descarga ilimitada al sumidero	Se asume que no existe una cota para el flujo o la concentración de contaminante que se descarga en el sumidero.
			X	Costos de bombeo y operación son despreciables	Se considera que la función objetivo económica está representada solamente por los costos de instalación, desestimando los asociados a la operación de plantas y bombeo entre nodos.
			X	Importancia relativa igual entre funciones objetivo	En la formulación de sumas ponderadas se asume que las funciones objetivo económica y ambiental poseen la misma importancia relativa.

D.2. Solución óptima

El modelo descrito anteriormente, junto con los valores nominales de los parámetros, presenta la solución óptima detallada en la Tabla D.3. Para esta configuración de las variables, la función objetivo alcanza un valor de 0,99 [–].

Tabla D.3: Resultados del ejemplo 2, empleando los valores nominales de los parámetros.

Variable	Valor	Unidad	Variable	Valor	Unidad
F_{N,P_1}	22,26	[t/h]	$F_{T_1,S}$	46,26	[t/h]
F_{N,P_2}	24,00	[t/h]	F_{T_2,P_1}	-	[t/h]
F_{P_1,P_2}	16,00	[t/h]	F_{T_2,P_2}	-	[t/h]
F_{P_1,T_1}	34,00	[t/h]	$F_{T_2,S}$	-	[t/h]
F_{P_1,T_2}	-	[t/h]	C_{T_1}	50,60	[ppm]
F_{P_2,P_1}	14,00	[t/h]	C_{T_2}	-	[ppm]
F_{P_2,T_1}	26,00	[t/h]	C_{P_1}	50,00	[ppm]
F_{P_2,T_2}	-	[t/h]	C_{P_2}	57,50	[ppm]
F_{T_1,P_1}	13,74	[t/h]	y_{T_1}	1,00	[–]
F_{T_1,P_2}	-	[t/h]	y_{T_2}	-	[–]

D.3. Rango de estabilidad

D.3.1. Parámetro D_{P_1}

El límite inferior del rango de estabilidad de D_{P_1} se determinó en la Tabla D.4. Las iteraciones realizadas para valores $D_{P_1} < 0$ ocasionan que la solución sea infactible. Habiendo determinado que el rango de estabilidad está acotado inferiormente por $D_{P_1} = 0$, en la Tabla D.5 se resume el valor de las variables del problema para ese caso, obteniéndose también que la función objetivo alcanza un valor de 1,11.

Tabla D.4: Determinación del límite inferior del rango de estabilidad del parámetro D_{P_1} .

i	D_{P_1}	$y' = y_0?$	D_{P_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	50,0	Sí	–	–	1,80
1	40,0	Sí	–	–	1,40
2	30,0	Sí	–	–	1,44
3	20,0	Sí	–	–	1,07
4	10,0	Sí	–	–	0,44
5	0,0	Sí	–	–	0,19
6	–10,0	No	0,0	$1,0 > 0,02$	0,07
Ciclo 2: $\Delta l = 1$					
0	0,0	Sí	–	–	0,15
1	–1,0	No	0,0	$0,1 > 0,02$	0,07
Ciclo 3: $\Delta l = 0,1$					
0	0,0	Sí	–	–	0,14
1	–0,1	No	0,0	$0,01 < 0,02$	0,09
Tiempo total: 6,86[s]					

De la Tabla D.5 se observa que los flujos que ingresan a P_1 disminuyen hasta el valor cero debido a la nula demanda de agua, por lo que no existirían las conexiones correspondientes. No hay ingreso de contaminante a los flujos de agua desde P_1 debido a que está condicionado a que existiera una salida de agua de aquella planta.

Respecto a la búsqueda del límite superior, las iteraciones realizadas están en la Tabla D.6. La solución del problema en el límite superior se resume en la Tabla D.7, obteniéndose un valor de función objetivo igual a 0,976 cuando $D_{P_1} = 57,6$.

Tabla D.5: Valores de las variables en el límite inferior de D_{P_1} .

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	-	22,26	[t/h]	$F_{T_1,S}$	25,35	46,26	[t/h]
F_{N,P_2}	25,35	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	-	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	-	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	54,63	50,60	[ppm]
F_{P_2,P_1}	-	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	40,00	26,00	[t/h]	C_{P_1}	-	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	57,50	57,50	[ppm]
F_{T_1,P_1}	-	13,74	[t/h]	y_{T_1}	1,00	1,00	[-]
F_{T_1,P_2}	14,65	-	[t/h]	y_{T_2}	-	-	[-]

Tabla D.6: Determinación del límite superior del rango de estabilidad del parámetro D_{P_1} .

i	D_{P_1}	$y' = y_0?$	D_{P_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	50,0	Sí	-	-	1,70
1	60,0	No	50,0	$1,0 > 0,02$	1,33
Ciclo 2: $\Delta l = 1,0$					
0	50,0	Sí	-	-	1,81
1	51,0	Sí	-	-	0,87
2	52,0	Sí	-	-	1,70
3	53,0	Sí	-	-	1,63
4	54,0	Sí	-	-	1,26
5	55,0	Sí	-	-	0,98
6	56,0	Sí	-	-	1,45
7	57,0	Sí	-	-	0,77
8	58,0	No	57,0	$0,1 > 0,02$	1,10
Ciclo 3: $\Delta l = 0,1$					
0	57,0	Sí	-	-	0,83
1	57,1	Sí	-	-	1,45
2	57,2	Sí	-	-	1,21
3	57,3	Sí	-	-	1,08
4	57,4	Sí	-	-	1,27
5	57,5	Sí	-	-	0,90
6	57,6	Sí	-	-	0,73
7	57,7	No	57,6	$0,01 < 0,02$	1,41
Tiempo total: 23,48[s]					

Tabla D.7: Valores de las variables en el límite superior de D_{P_1} .

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	25,72	22,26	[t/h]	$F_{T_1,S}$	48,83	46,26	[t/h]
F_{N,P_2}	23,11	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	16,89	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	40,71	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	48,09	50,60	[ppm]
F_{P_2,P_1}	20,71	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	19,29	26,00	[t/h]	C_{P_1}	47,36	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	57,50	57,50	[ppm]
F_{T_1,P_1}	11,17	13,74	[t/h]	y_{T_1}	1,00	1,00	[-]
F_{T_1,P_2}	-	-	[t/h]	y_{T_2}	-	-	[-]

En términos generales, los resultados obtenidos se adecúan a lo esperado: A mayor demanda, la mayoría de los flujos que ingresan a P_1 aumentan. Por otro lado, la interpretación gráfica también corresponde a la esperada, en cuanto a los valores de las variables binarias dentro y fuera del rango calculado, tal como se observa en la Figura D.1. Los valores graficados corresponden a 100 variaciones equidistantes del parámetro D_{P_1} .

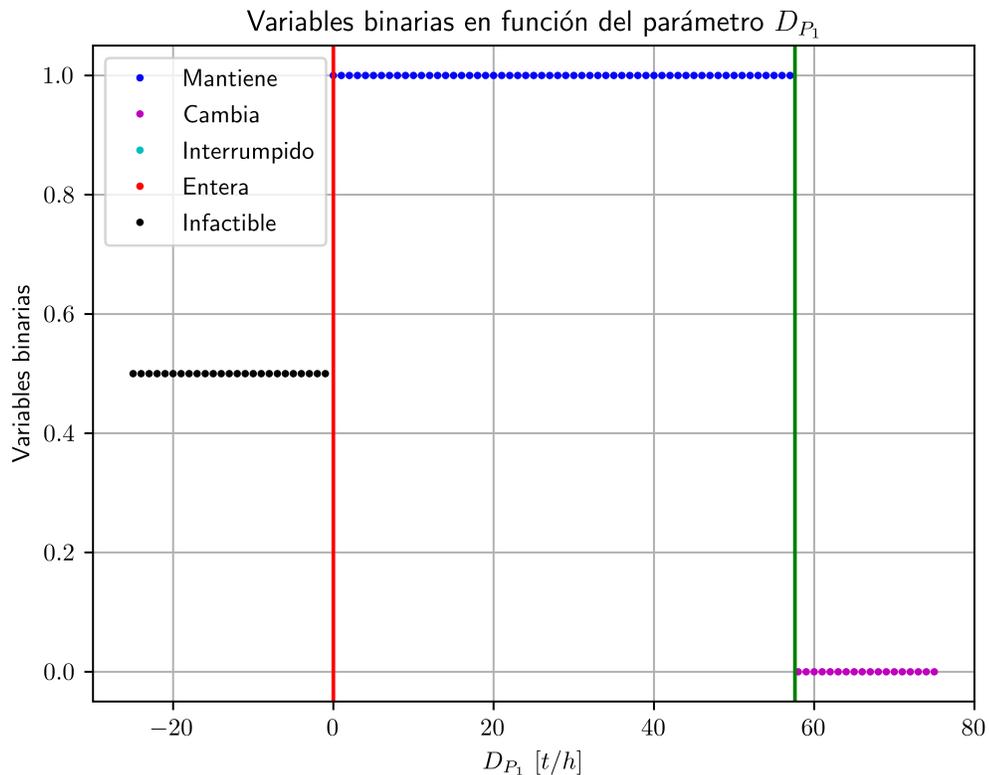


Figura D.1: Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro D_{P_1} . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

D.3.2. Parámetro D_{P_2}

El límite inferior del rango de estabilidad de D_{P_2} se calculó con la información de la Tabla D.8, mientras que los valores de las variables en aquel límite se encuentran en la Tabla D.9, alcanzando la función objetivo un valor de 0,84 para $D_{P_2} = 0$.

Tabla D.8: Determinación del límite inferior del rango de estabilidad del parámetro D_{P_2} .

i	D_{P_2}	$y' = y_0?$	D_{P_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	40,0	Sí	–	–	1,89
1	30,0	Sí	–	–	1,11
2	20,0	Sí	–	–	0,80
3	10,0	Sí	–	–	0,72
4	0,0	Sí	–	–	0,20
5	–10,0	No	0,0	$1,0 > 0,016$	0,06
Ciclo 2: $\Delta l = 1$					
0	0,0	Sí	–	–	0,24
1	–1,0	No	0,0	$0,1 > 0,016$	0,07
Ciclo 3: $\Delta l = 0,1$					
0	0,0	Sí	–	–	0,19
1	–0,1	No	0,0	$0,01 < 0,016$	0,07
Tiempo total: 5,35[s]					

Tabla D.9: Valores de las variables en el límite inferior de D_{P_2} .

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	18,42	22,26	[t/h]	$F_{T_1,S}$	18,42	46,26	[t/h]
F_{N,P_2}	-	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	-	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	50,00	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	47,50	50,60	[ppm]
F_{P_2,P_1}	-	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	-	26,00	[t/h]	C_{P_1}	50,00	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	-	57,50	[ppm]
F_{T_1,P_1}	31,58	13,74	[t/h]	y_{T_1}	1,00	1,00	[–]
F_{T_1,P_2}	-	-	[t/h]	y_{T_2}	-	-	[–]

Los resultados de la Tabla D.9 señalan que los valores obtenidos tras la disminución de D_{P_2} son similares a lo que sucede con D_{P_1} : Los flujos que ingresan y salen de la planta P_2 son nulos, no existiendo las conexiones asociadas a esta planta.

Las iteraciones realizadas para determinar el límite superior del rango de estabilidad se encuentran en la Tabla D.10. Se observa que la detención se produce en el valor 81,9, ya que se ha alcanzado el número máximo de iteraciones para un mismo paso ($i < 10$). Ante esta situación el algoritmo sugiere actualizar los parámetros de búsqueda, aunque en este caso se ha notado que el valor 82,0 produce un cambio en las variables binarias. A partir de esta información, se concluye que el límite se encuentra entre 81,9 y 82,0, siendo el usuario quien analice si la precisión alcanzada es suficiente. Para este caso particular, si se decidiera continuar con la búsqueda desde 81,9 con un paso 10 veces inferior al anterior (emplear 0,01 en lugar de 0,1), que es la actualización por defecto, el algoritmo se detiene por ser inferior el paso a la tolerancia. Con esto se concluye que el límite es 81,9. Para ese valor, la solución del problema es la informada en la Tabla D.11, logrando una función objetivo igual a 0,931.

Tabla D.10: Determinación del límite superior del rango de estabilidad del parámetro D_{P_2} .

i	D_{P_2}	$y' = y_0?$	D_{P_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	40,0	Sí	—	—	0,90
1	50,0	Sí	—	—	10,61
2	60,0	Sí	—	—	1,01
3	70,0	Sí	—	—	0,77
4	80,0	Sí	—	—	1,33
5	90,0	No	80,0	$1,0 > 0,016$	0,87
Ciclo 2: $\Delta l = 1,0$					
0	80,0	Sí	—	—	1,42
1	81,0	Sí	—	—	1,26
2	82,0	No	81,0	$0,1 > 0,016$	0,93
Ciclo 3: $\Delta l = 0,1$					
0	81,0	Sí	—	—	1,18
1	81,1	Sí	—	—	1,21
2	81,2	Sí	—	—	1,39
3	81,3	Sí	—	—	1,47
4	81,4	Sí	—	—	0,77
5	81,5	Sí	—	—	1,20
6	81,6	Sí	—	—	0,48
7	81,7	Sí	—	—	0,50
8	81,8	Sí	—	—	0,43
9	81,9	Sí	—	—	0,48
Tiempo total: 28,21[s]					

Tabla D.11: Valores de las variables en el límite superior de D_{P_2} .

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	10,85	22,26	[t/h]	$F_{T_1,S}$	59,99	46,26	[t/h]
F_{N,P_2}	49,14	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	32,75	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	17,25	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	39,59	50,60	[ppm]
F_{P_2,P_1}	39,15	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	42,75	26,00	[t/h]	C_{P_1}	50,00	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	38,32	57,50	[ppm]
F_{T_1,P_1}	-	13,74	[t/h]	y_{T_1}	1,00	1,00	[-]
F_{T_1,P_2}	0,01	-	[t/h]	y_{T_2}	-	-	[-]

Finalmente, se puede observar en la Figura D.2 que el comportamiento de las variables binarias concuerda con lo esperado al calcular el rango de estabilidad, siendo comprobado con 100 valores arbitrarios evaluados. Tal como ocurría con D_{P_1} , los valores negativos ocasionan que la solución sea infactible.

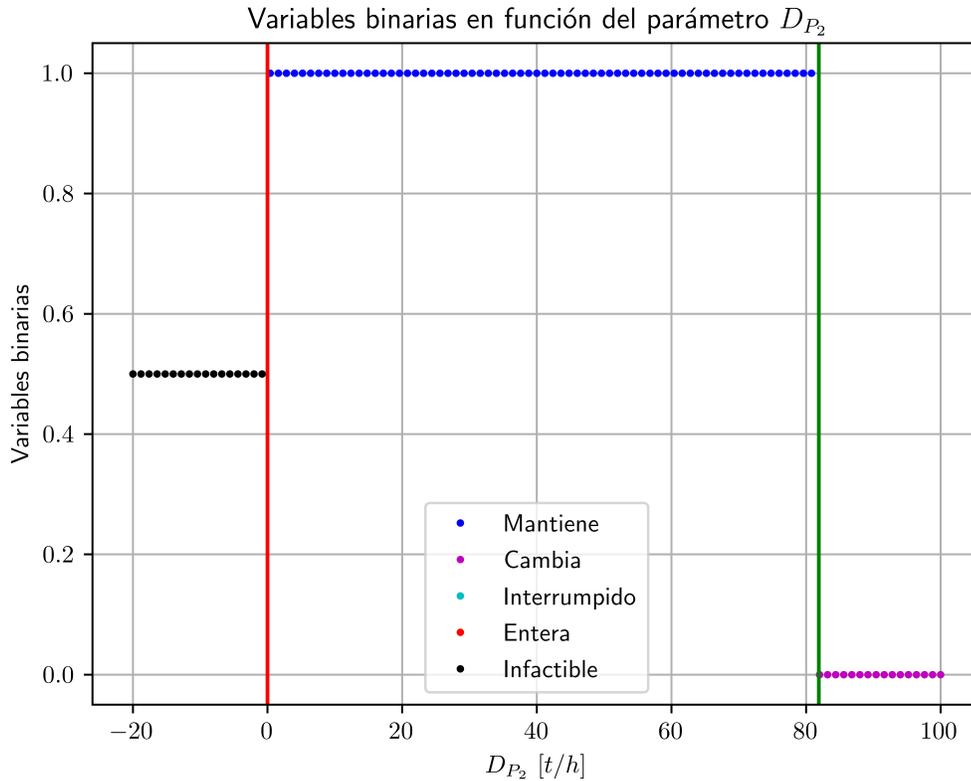


Figura D.2: Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro D_{P_2} . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

D.3.3. Parámetro $Cost_{T_1}$

Para calcular el límite inferior de $Cost_{T_1}$ se realizaron las iteraciones presentes en la Tabla D.12. Se observa que el valor obtenido es inferior a cero, lo que no sería posible al tratarse de costos. Desde otra perspectiva un costo negativo podría comprenderse como una ganancia, pero en este ejemplo solamente se utilizará la idea de costo asociado a lo gastado en la inversión. Comprendiendo que el valor cero mantiene las variables binarias, se analizará cómo se comportan las variables en ese escenario extremo ($Cost_{T_1} = 0$), obteniéndose como resultado los valores de la Tabla D.13, mientras que la función objetivo alcanzó un valor de 0,514.

Tabla D.12: Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_1}$.

i	$Cost_{T_1}$	$y' = y_0?$	$Cost_{T_1}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	1.000.000	Sí	—	—	0,82
1	0	Sí	—	—	0,48
2	-1.000.000	Sí	—	—	0,47
3	-2.000.000	No	-1.000.000	100.000 > 400	0,53
Ciclo 2: $\Delta l = 100.000$					
0	-1.000.000	Sí	—	—	0,40
1	-1.100.000	Sí	—	—	0,14
2	-1.200.000	No	-1.100.000	10.000 > 400	0,52
Ciclo 3: $\Delta l = 10.000$					
0	-1.100.000	Sí	—	—	0,13
1	-1.110.000	No	-1.100.000	1.000 > 400	0,53
Ciclo 4: $\Delta l = 1.000$					
0	-1.100.000	Sí	—	—	0,15
1	-1.101.000	No	-1.100.000	100 < 400	0,37
Tiempo total: 4,54[s]					

Para obtener el límite superior se necesitaron las iteraciones de la Tabla D.14, mientras que en ese valor límite la solución del problema se resume en la Tabla D.15, teniendo una función objetivo igual a 0,995. Tanto para el límite inferior como el superior del rango de estabilidad de $Cost_{T_1}$, las variables del problema mantienen su valor original (no solo las binarias). Pese a lo anterior, la función objetivo disminuye al decrecer el costo, lo que se debe a que solo uno de los costos está decreciendo, ocasionando un numerador con un valor inferior y generando con ello una función objetivo menor. Análogamente, el aumento en solo uno de los costos produce una función objetivo de mayor valor.

Tabla D.13: Valores de las variables en el límite inferior de $Cost_{T_1}$, aceptando solamente costos no negativos.

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	22,26	22,26	[t/h]	$F_{T_1,S}$	46,26	46,26	[t/h]
F_{N,P_2}	24,00	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	16,00	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	34,00	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	50,60	50,60	[ppm]
F_{P_2,P_1}	14,00	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	26,00	26,00	[t/h]	C_{P_1}	50,00	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	57,50	57,50	[ppm]
F_{T_1,P_1}	13,74	13,74	[t/h]	y_{T_1}	1,00	1,00	[-]
F_{T_1,P_2}	-	-	[t/h]	y_{T_2}	-	-	[-]

Tabla D.14: Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_1}$.

i	$Cost_{T_1}$	$y' = y_0?$	$Cost_{T_1}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	1.000.000	Sí	-	-	1,90
1	2.000.000	No	1.000.000	100.000 > 400	1,10
Ciclo 2: $\Delta l = 100.000$					
0	1.000.000	Sí	-	-	2,13
1	1.100.000	No	1.000.000	10.000 > 400	0,51
Ciclo 3: $\Delta l = 10.000$					
0	1.000.000	Sí	-	-	1,88
1	1.010.000	Sí	-	-	2,02
2	1.020.000	No	1.010.000	1.000 > 400	2,32
Ciclo 4: $\Delta l = 1.000$					
0	1.010.000	Sí	-	-	2,13
1	1.011.000	Sí	-	-	0,80
2	1.012.000	Sí	-	-	0,71
3	1.013.000	Sí	-	-	0,77
4	1.014.000	Sí	-	-	0,70
5	1.015.000	Sí	-	-	2,30
6	1.016.000	Sí	-	-	2,32
7	1.017.000	Sí	-	-	0,73
8	1.018.000	Sí	-	-	0,98
9	1.019.000	No	1.018.000	100 < 400	2,17
Tiempo total: 25,47[s]					

Tabla D.15: Valores de las variables en el límite superior de $Cost_{T_1}$.

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	22,26	22,26	[t/h]	$F_{T_1,S}$	46,26	46,26	[t/h]
F_{N,P_2}	24,00	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	16,00	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	34,00	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	50,60	50,60	[ppm]
F_{P_2,P_1}	14,00	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	26,00	26,00	[t/h]	C_{P_1}	50,00	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	57,50	57,50	[ppm]
F_{T_1,P_1}	13,74	13,74	[t/h]	y_{T_1}	1,00	1,00	[-]
F_{T_1,P_2}	-	-	[t/h]	y_{T_2}	-	-	[-]

Finalmente, tras 100 evaluaciones arbitrarias y equidistantes para $Cost_{T_1}$ se observa en la Figura D.3 que los resultados concuerdan con lo esperado desde la solución matemática. Sin embargo, es importante notar que se deben contextualizar los valores, por lo que la zona de interés corresponde a los valores mayores o iguales a cero.

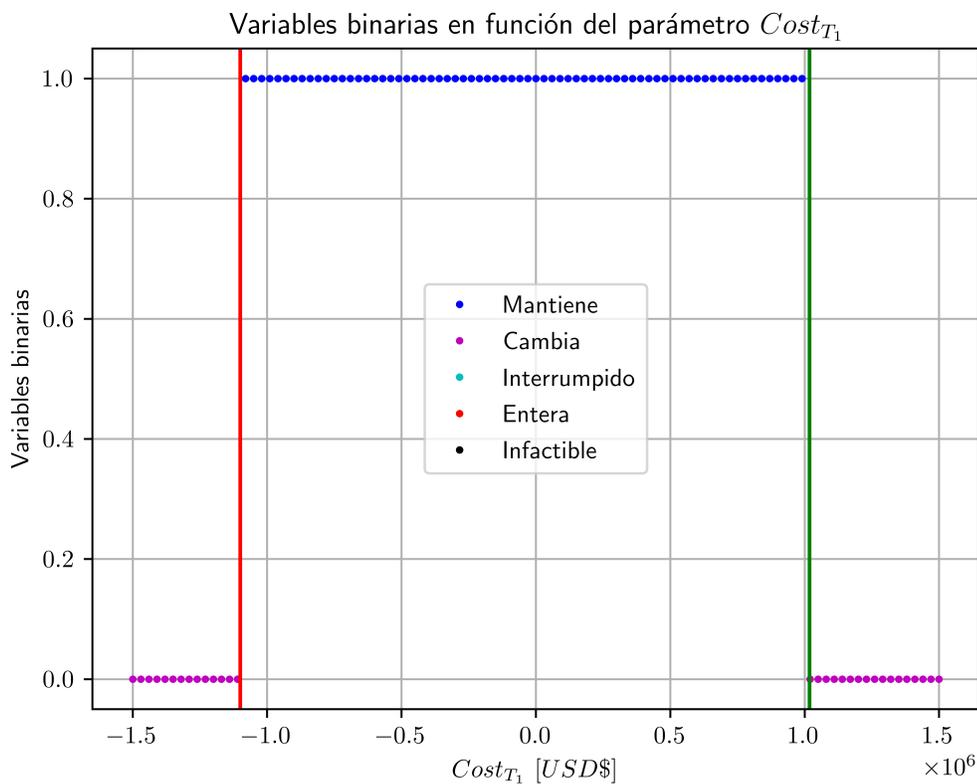


Figura D.3: Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro $Cost_{T_1}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

D.3.4. Parámetro $Cost_{T_2}$

Para obtener el límite inferior del rango de estabilidad de $Cost_{T_2}$ se realizaron las iteraciones que se encuentran en la Tabla D.16. En este caso se ha alcanzado el mayor valor de i , por lo que el algoritmo sugiere actualizar los parámetros de búsqueda si se necesitara una mayor precisión en la determinación del límite. Sin embargo, tal como ocurrió con el proceso de búsqueda del límite superior de $Cost_{T_1}$, una nueva disminución del paso en un orden de magnitud ocasionaría que este sea menor a la tolerancia establecida, por lo que el algoritmo se detendría y quedaría definido $Cost_{T_2} = 1.081.000$ como el límite inferior. Para este caso, las variables adquieren el valor que se observa en la Tabla D.17, mientras que la función objetivo alcanza un valor de 0,995.

Tabla D.16: Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_2}$.

i	$Cost_{T_2}$	$y' = y_0?$	$Cost_{T_2}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	1.100.000	Sí	–	–	1,90
1	100.000	No	1.100.000	$100.000 > 400$	0,38
Ciclo 2: $\Delta l = 100.000$					
0	1.100.000	Sí	–	–	1,96
1	1.000.000	No	1.100.000	$10.000 > 400$	0,49
Ciclo 3: $\Delta l = 10.000$					
0	1.100.000	Sí	–	–	2,16
1	1.090.000	Sí	–	–	0,75
2	1.080.000	No	1.090.000	$1.000 > 400$	2,26
Ciclo 4: $\Delta l = 1.000$					
0	1.090.000	Sí	–	–	0,72
1	1.089.000	Sí	–	–	0,75
2	1.088.000	Sí	–	–	0,85
3	1.087.000	Sí	–	–	0,83
4	1.086.000	Sí	–	–	0,85
5	1.085.000	Sí	–	–	0,87
6	1.084.000	Sí	–	–	0,78
7	1.083.000	Sí	–	–	2,33
8	1.082.000	Sí	–	–	0,84
9	1.081.000	Sí	–	–	2,27
Tiempo total:					20,99[s]

Tabla D.17: Valores de las variables en el límite inferior de $Cost_{T_2}$.

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	22,26	22,26	[t/h]	$F_{T_1,S}$	46,26	46,26	[t/h]
F_{N,P_2}	24,00	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	16,00	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	34,00	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	50,60	50,60	[ppm]
F_{P_2,P_1}	14,00	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	26,00	26,00	[t/h]	C_{P_1}	50,00	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	57,50	57,50	[ppm]
F_{T_1,P_1}	13,74	13,74	[t/h]	y_{T_1}	1,00	1,00	[-]
F_{T_1,P_2}	-	-	[t/h]	y_{T_2}	-	-	[-]

Las iteraciones realizadas en la búsqueda del límite superior del rango de estabilidad se encuentran en la Tabla D.18. El algoritmo sugiere actualizar los parámetros de búsqueda para continuar con la determinación del extremo del rango de estabilidad. Sin embargo, dada la estructura del problema, se concluye que no existe un límite superior. Particularmente, se emplea como ejemplo el último valor evaluado en las iteraciones para conocer así el comportamiento de las variables del problema en ese escenario mediante la información de la Tabla D.19. El valor de la función objetivo en ese caso es de 0,604.

Tabla D.18: Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_2}$.

i	$Cost_{T_2}$	$y' = y_0?$	$Cost_{T_2}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	1.100.000	Sí	—	—	1,93
1	2.100.000	Sí	—	—	1,32
2	3.100.000	Sí	—	—	1,17
3	4.100.000	Sí	—	—	1,19
4	5.100.000	Sí	—	—	1,29
5	6.100.000	Sí	—	—	1,45
6	7.100.000	Sí	—	—	1,34
7	8.100.000	Sí	—	—	1,35
8	9.100.000	Sí	—	—	2,36
9	10.100.000	Sí	—	—	1,97
Tiempo total: 15,37[s]					

Al disminuir el valor del parámetro hasta el límite inferior o aumentar a un valor aproximado de 9 veces el original, los valores de las variables no cambian, ocasionando que las conexiones de la red se mantengan pese a la variación mencionada en el costo de instalación de T_2 . La única diferencia en ambos casos es el valor de la función objetivo, debido a la influencia del parámetro en su estructura, situación similar a lo ocurrido con el costo de T_1 .

Tabla D.19: Valores de las variables en el límite superior de $Cost_{T_2}$.

Var.	Actual	Nominal	UN	Var.	Actual	Nominal	UN
F_{N,P_1}	22,26	22,26	[t/h]	$F_{T_1,S}$	46,26	46,26	[t/h]
F_{N,P_2}	24,00	24,00	[t/h]	F_{T_2,P_1}	-	-	[t/h]
F_{P_1,P_2}	16,00	16,00	[t/h]	F_{T_2,P_2}	-	-	[t/h]
F_{P_1,T_1}	34,00	34,00	[t/h]	$F_{T_2,S}$	-	-	[t/h]
F_{P_1,T_2}	-	-	[t/h]	C_{T_1}	50,60	50,60	[ppm]
F_{P_2,P_1}	14,00	14,00	[t/h]	C_{T_2}	-	-	[ppm]
F_{P_2,T_1}	26,00	26,00	[t/h]	C_{P_1}	50,00	50,00	[ppm]
F_{P_2,T_2}	-	-	[t/h]	C_{P_2}	57,50	57,50	[ppm]
F_{T_1,P_1}	13,74	13,74	[t/h]	y_{T_1}	1,00	1,00	[-]
F_{T_1,P_2}	-	-	[t/h]	y_{T_2}	-	-	[-]

Finalmente, se realizan 100 evaluaciones arbitrarias para el parámetro $Cost_{T_2}$ para comprobar el comportamiento de las variables binarias, observándose en la Figura D.4 que los resultados coinciden con lo esperado.

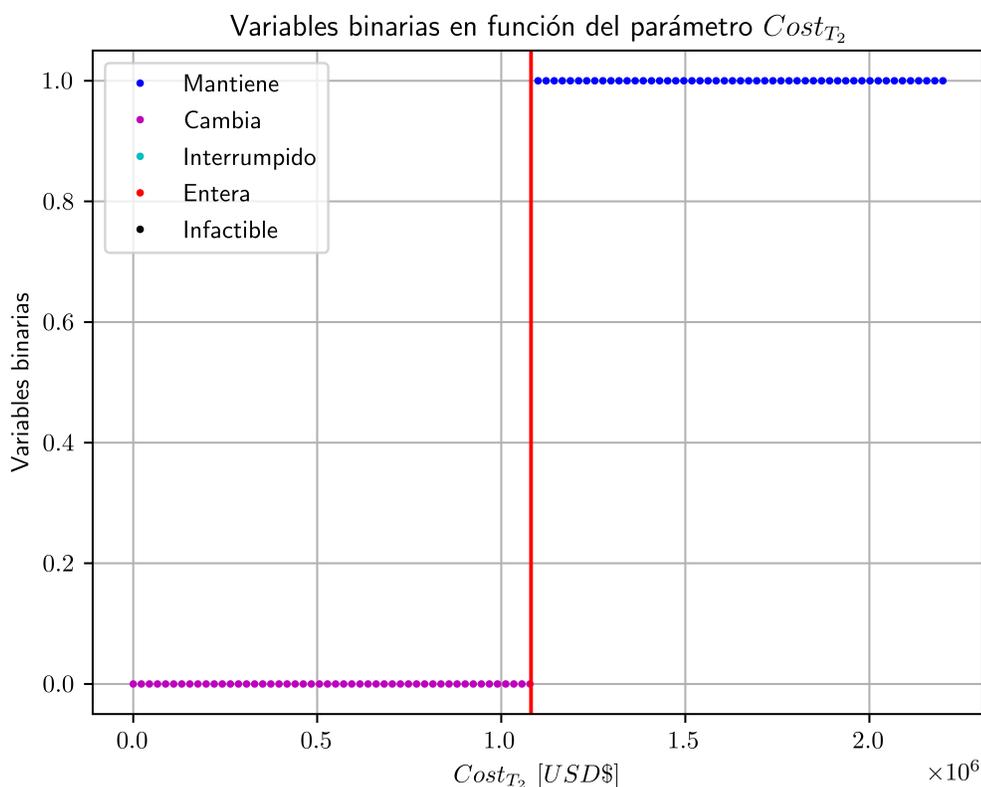


Figura D.4: Variación de las variables binarias y_{T_1} e y_{T_2} frente a cambios en el parámetro $Cost_{T_2}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

D.4. Efecto del tiempo de resolución

Como se ha mencionado en la sección 5.2.1.4, se estudia a continuación el tiempo requerido en las múltiples resoluciones efectuadas para los diferentes parámetros en estudio.

D.4.1. Parámetro D_{P_1}

Alrededor de 100 iteraciones se realizaron para analizar el efecto del parámetro D_{P_1} en las variables binarias del problema, resumiéndose en la Figura D.1. El tiempo requerido en estas, en relación al tiempo utilizado en el caso nominal, se representa en la Figura D.5. En este gráfico es posible notar que todas las ejecuciones necesitaron un tiempo inferior a 5 veces el nominal, aunque con un número aproximadamente cercano entre los casos que requieren menos tiempo que el original (53 %) y los que están por sobre ese valor (47 %), contrastando con la mayor diferencia que se observa en el ejemplo 1.

Número de iteraciones de D_{P_1} resueltas según el tiempo de ejecución (ponderado por τ)

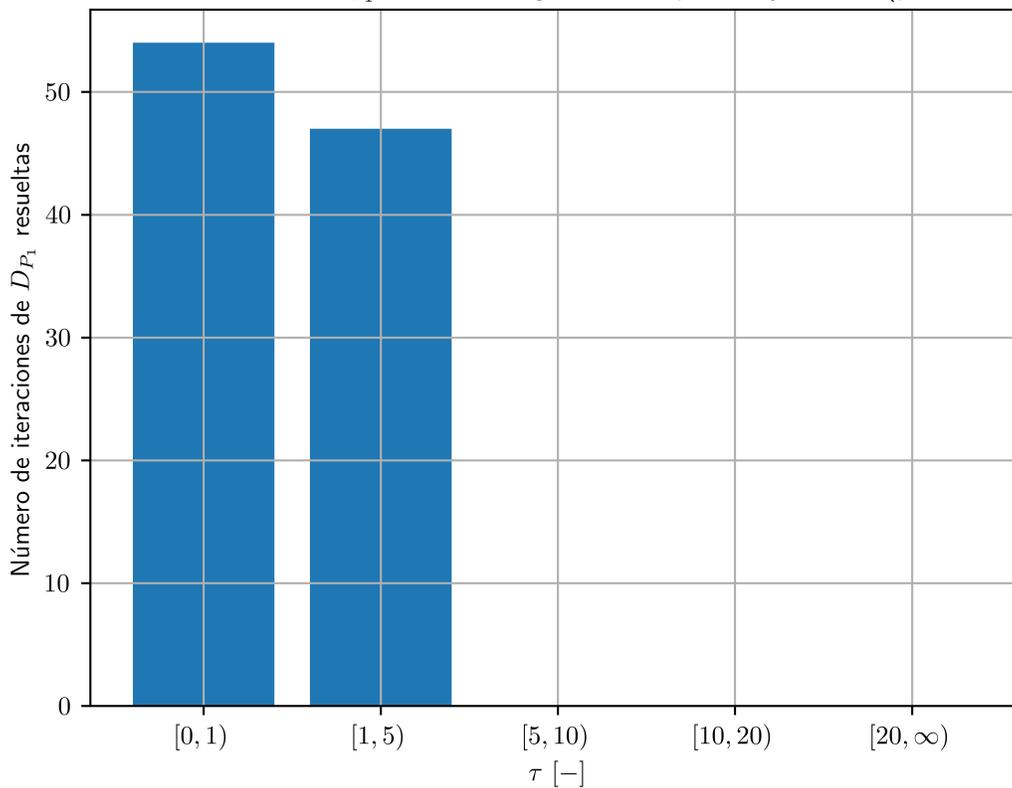


Figura D.5: Número de iteraciones de D_{P_1} resueltas según el tiempo de ejecución.

D.4.2. Parámetro D_{P_2}

Para este parámetro, las múltiples resoluciones resumidas en la Figura D.2 necesitaron diferentes tiempos de ejecución, clasificados en la Figura D.6. En este gráfico se observa, nuevamente, que se requiere un ponderador $\tau < 5$ para resolver las iteraciones sin que sean interrumpidas. Particularmente, la mayoría requiere un tiempo inferior al nominal, presentándose una mayor diferencia con el número de casos que necesitan más tiempo, en contraste con el comportamiento de las evaluaciones de D_{P_1} .

Número de iteraciones de D_{P_2} resueltas según el tiempo de ejecución (ponderado por τ)

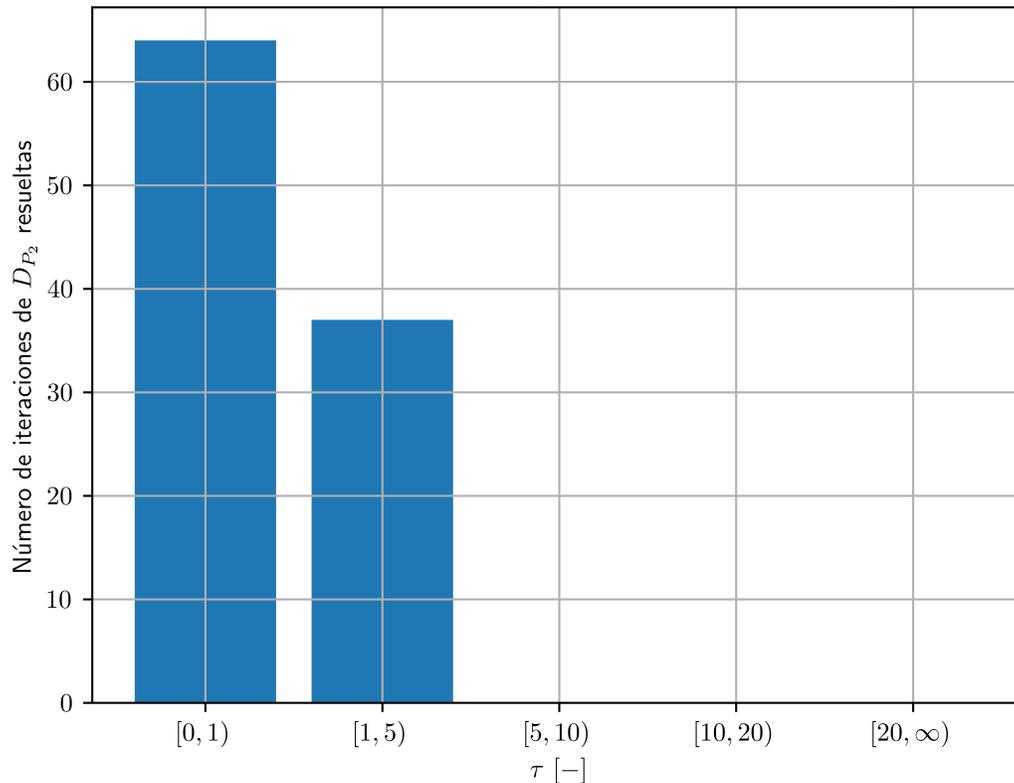


Figura D.6: Número de iteraciones de D_{P_2} resueltas según el tiempo de ejecución.

D.4.3. Parámetro $Cost_{T_1}$

Las múltiples resoluciones de la Figura D.3 presentaron un comportamiento similar a los parámetros anteriores en cuanto al tiempo de ejecución. Específicamente, la mayoría de las variaciones de $Cost_{T_1}$ requirieron un tiempo inferior al nominal, mientras que el resto se ejecuta en tiempos menores a 5 veces el nominal, lo que se aprecia en la Figura D.7.

Número de iteraciones de $Cost_{T_1}$ resueltas según el tiempo de ejecución (ponderado por τ)

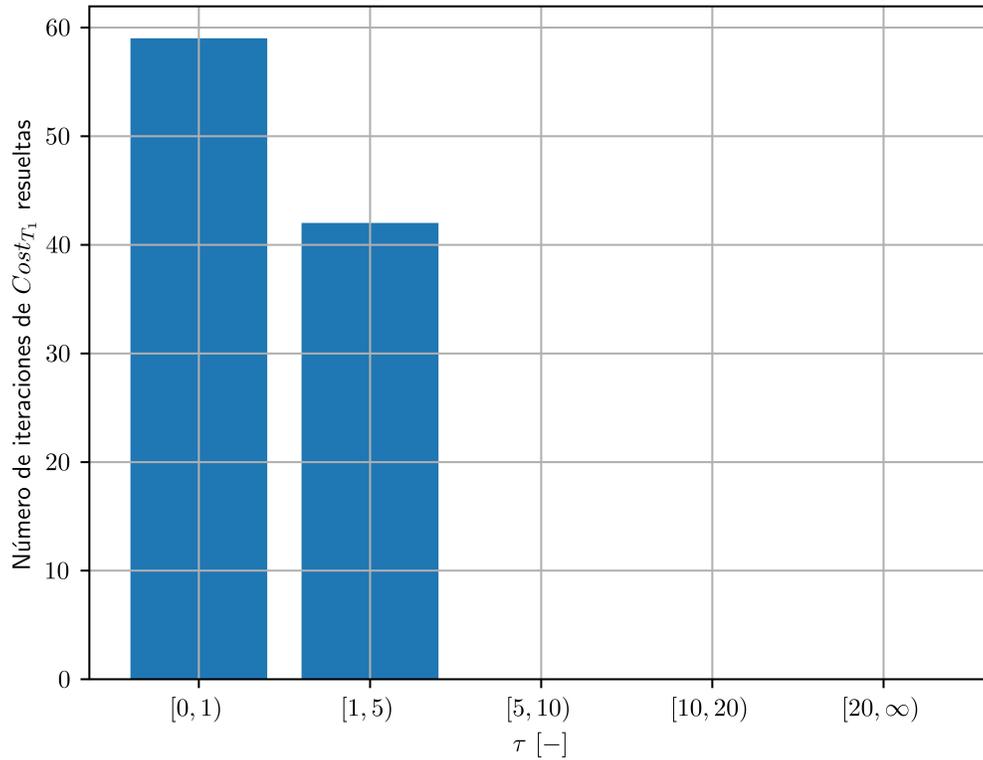


Figura D.7: Número de iteraciones de $Cost_{T_1}$ resueltas según el tiempo de ejecución.

D.4.4. Parámetro $Cost_{T_2}$

Las variaciones de $Cost_{T_2}$, representadas en la Figura D.4, presentan un comportamiento distinto a los parámetros anteriores, tanto del presente ejemplo como del primero. La diferencia radica en que la mayoría de las ejecuciones necesitaron un tiempo superior al nominal, por sobre los que requieren un tiempo inferior. De todas formas, concuerdan con los otros parámetros en la necesidad de un tiempo inferior a 5 veces el valor nominal ($\tau < 5$) para ejecutar sin interrupciones cada evaluación.

Número de iteraciones de $Cost_{T_2}$ resueltas según el tiempo de ejecución (ponderado por τ)

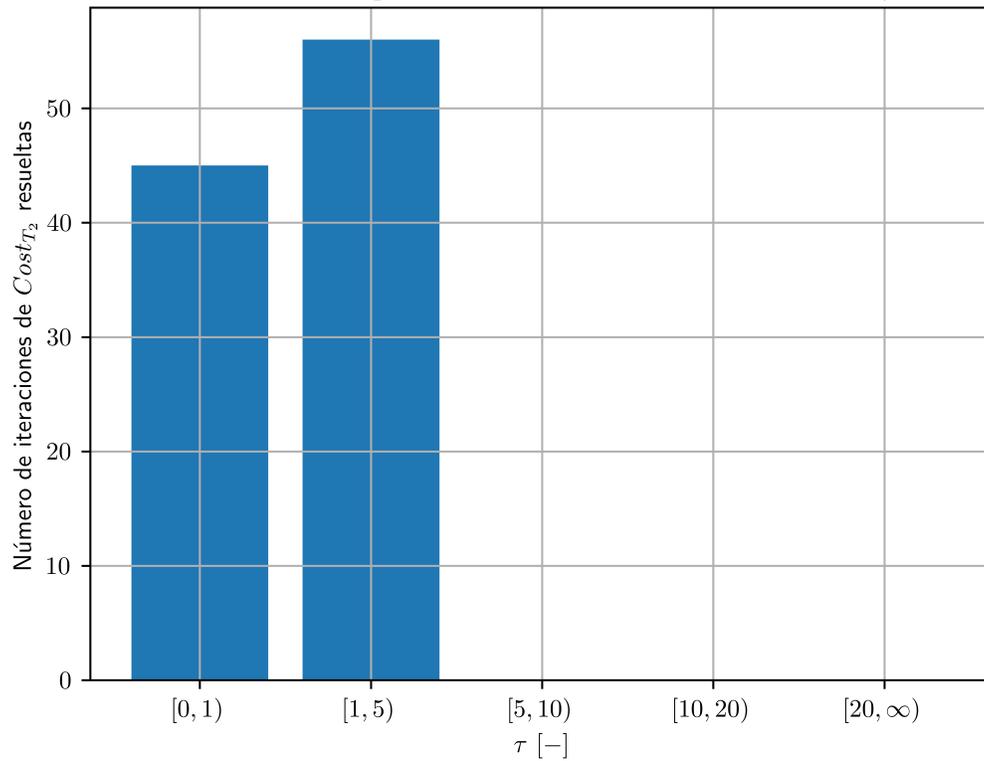


Figura D.8: Número de iteraciones de $Cost_{T_2}$ resueltas según el tiempo de ejecución.

D.5. Análisis de inecuaciones

En el primer ejemplo de este documento se reconoció que los cambios en las variables binarias se producían cuando existía un cambio en las restricciones activas del problema. Con el propósito de encontrar una tendencia similar en el segundo ejemplo, en esta sección se grafican las restricciones de desigualdad (D.11, D.12, D.13, D.14, D.15 y D.16) con respecto al valor de los parámetros analizados en diferentes iteraciones.

$$Inst_1 : \sum_{i \in I} F_{i,T_1} + \sum_{i \in I \cup \{S\}} F_{T_1,i} - y_{T_1} \cdot M \leq 0 \quad (D.11)$$

$$Inst_2 : \sum_{i \in I} F_{i,T_2} + \sum_{i \in I \cup \{S\}} F_{T_2,i} - y_{T_2} \cdot M \leq 0 \quad (D.12)$$

$$Cap_1 : \sum_{i \in I} F_{i,T_1} - Cap_{T_1} \leq 0 \quad (D.13)$$

$$Cap_2 : \sum_{i \in I} F_{i,T_2} - Cap_{T_2} \leq 0 \quad (D.14)$$

$$Cal_1 : \sum_{k \in K} F_{k,P_1} \cdot C_k - \sum_{k \in K} F_{k,P_1} \cdot Cal_{P_1} \leq 0 \quad (D.15)$$

$$Cal_2 : \sum_{k \in K} F_{k,P_2} \cdot C_k - \sum_{k \in K} F_{k,P_2} \cdot Cal_{P_2} \leq 0 \quad (D.16)$$

En relación a D_{P_1} , se resolvió el problema para 100 valores distintos de este parámetro, cuyo efecto en las restricciones del problema se observa en la Figura D.9. Recordando que el valor nominal de D_{P_1} es de 50 [t/h], se aprecia que en ese caso la planta T_1 opera a capacidad completa (restricción de capacidad Cap_1 está activa). Si se aumenta el valor de este parámetro, la planta sigue a capacidad completa hasta que se alcanza el límite superior del rango de estabilidad, dejando de operar y siendo instalada la planta T_2 , que pasa a funcionar a capacidad completa. De esta forma, son las restricciones de capacidad las que pasan o dejan de estar activas al momento de producirse un cambio en las variables binarias.

La disminución de D_{P_1} no permite obtener conclusiones al respecto, debido a que para valores de este parámetro inferiores a cero la solución es infactible.

Las restricciones $Inst_1$ y $Inst_2$ están formuladas para relacionar la instalación de las plantas de tratamiento con los flujos asociados a estas mediante una restricción del tipo *Big M*. Es por ello que se esperaba y ocurrió que estuviesen activas en los casos en que una planta no se instalara, es decir, para los casos en que la variable binaria correspondiente fuese nula.

El análisis de la demanda de la planta P_2 , D_{P_2} , se realizó también luego de 100 ejecuciones del problema, obteniéndose los resultados de la Figura D.10. Tal como ocurría con D_{P_1} , en este caso el aumento del valor del parámetro mantiene activa la restricción Cap_1 , mientras que al decidir ser instalada la planta T_2 , Cap_2 pasa a estar activa y Cap_1 deja de estarlo. Nuevamente la infactibilidad generada para valores negativos de D_{P_2} no permite obtener conclusiones al respecto. Adicionalmente, las restricciones del tipo *Big M* ($Inst_1$ e $Inst_2$) se comportan como se esperaba: Se encuentran activas o inactivas según el valor de las variables binarias.

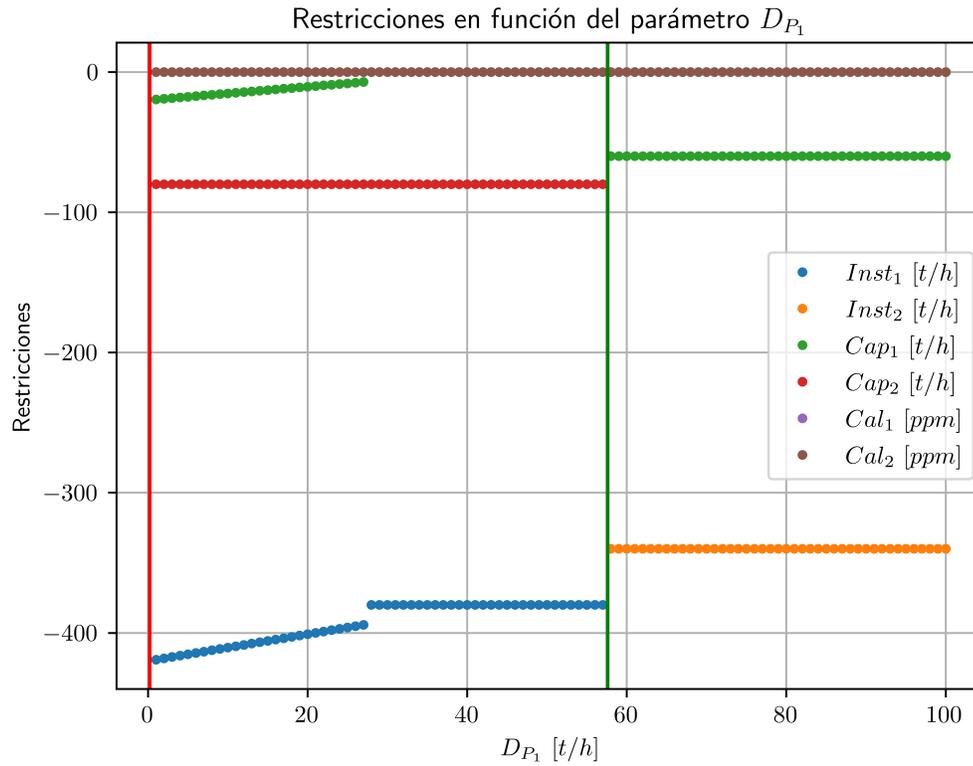


Figura D.9: Variación del valor de las restricciones frente a cambios en el parámetro D_{P_1} . Línea vertical roja: límite inferior calculado. Línea vertical verde: límite superior calculado.

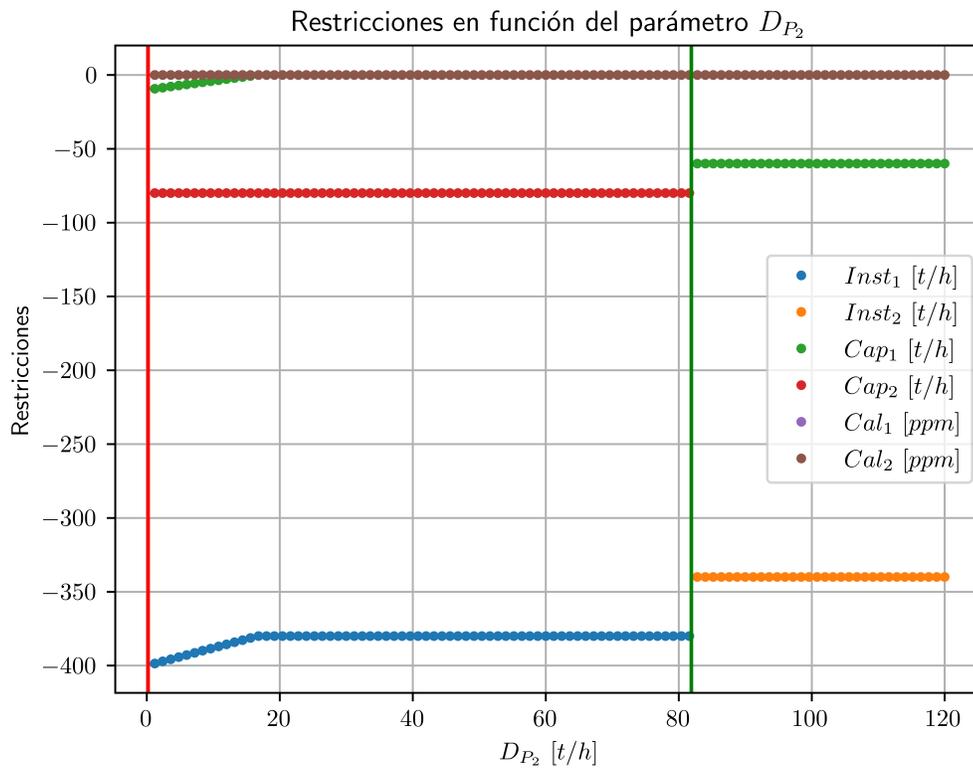


Figura D.10: Variación del valor de las restricciones frente a cambios en el parámetro D_{P_2} . Línea vertical roja: límite inferior calculado. Línea vertical verde: límite superior calculado.

Los cambios en $Cost_{T_1}$, según se observa en la Figura D.11, mantienen la tendencia que se originaba en los aumentos de la demanda de las plantas de procesos: Si se instala una planta de tratamiento, opera a capacidad completa en el rango analizado. Este efecto se produce también para la disminución del valor del parámetro. Específicamente, tras aumentar el valor del parámetro $Cost_{T_1}$ y alcanzar el límite superior del rango de estabilidad, dos restricciones que estaban activas dejan de estarlo (Cap_1 e $Inst_2$), mientras que dos que estaban no activas pasan a estarlo (Cap_2 e $Inst_1$), como consecuencia de operar a capacidad completa si se decide instalarlas.

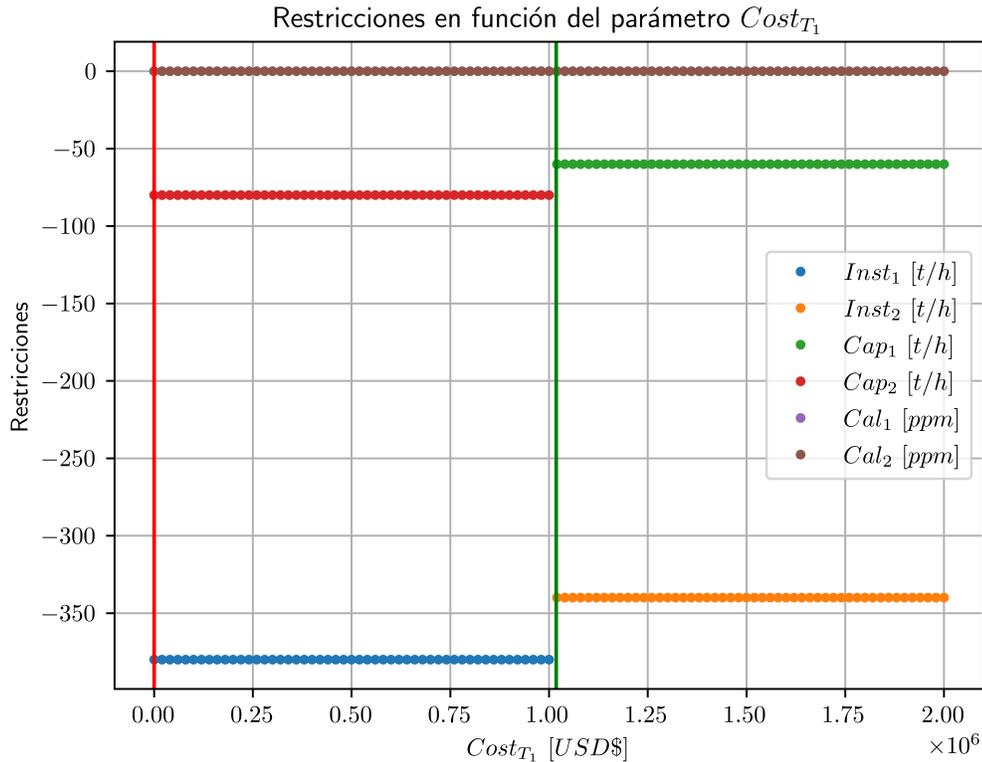


Figura D.11: Variación del valor de las restricciones frente a cambios en el parámetro $Cost_{T_1}$. Línea vertical roja: límite inferior calculado. Línea vertical verde: límite superior calculado.

Los cambios en las restricciones debido a modificaciones en $Cost_{T_2}$ se observan en la Figura D.12, notándose que los efectos son similares a lo ocurrido con cambios en $Cost_{T_1}$: Las plantas instaladas operan a capacidad completa y las restricciones activas del problema cambian al alcanzarse el límite del rango de estabilidad.

En términos generales, las restricciones Cal_1 y Cal_2 se encuentran activas (o con un valor aproximadamente nulo) en todos los casos analizados, por lo que su comportamiento no permitiría obtener conclusiones respecto a los rangos de estabilidad de los parámetros estudiados.

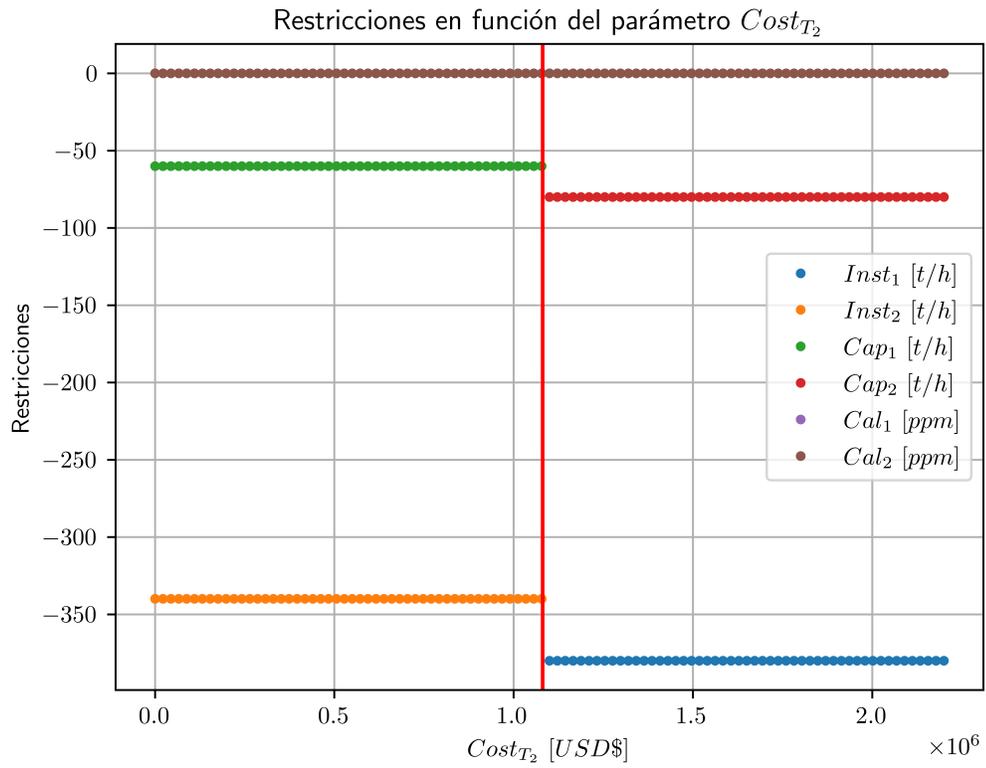


Figura D.12: Variación del valor de las restricciones frente a cambios en el parámetro $Cost_{T_2}$. Línea vertical roja: límite inferior calculado.

D.6. Optimización de parámetros

Para resolver el problema reformulado se ha intercambiado la naturaleza de algunos parámetros a variables y las variables binarias a parámetros. Específicamente, los parámetros D_{P_1} , D_{P_2} , $Cost_{T_1}$ y $Cost_{T_2}$ pasan a ser variables, uno a la vez. Por otro lado, las variables binarias y_{T_1} e y_{T_2} se consideran en esta sección como parámetros, adoptando los valores de la solución del problema original.

El resultado de los subproblemas para cada parámetro, el tiempo requerido para determinarlos, el utilizado en la evaluación de estos valores en el caso original y su factibilidad de uso como nuevo punto inicial se resume en la Tabla D.20:

Tabla D.20: Parámetros optimizados del ejemplo ilustrativo 2.

Parámetro optimizado [UN]	Valor obtenido	Tiempo ejec. reformulado [s]	Tiempo de evaluación [s]	¿Mantiene binaria?
$D_{P_1,op}^{min}$ [t/h]	0,0	1,24	0,13	Sí
$D_{P_1,op}^{max}$ [t/h]	79,2	0,56	0,38	No
$D_{P_2,op}^{min}$ [t/h]	0,0	0,29	0,20	Sí
$D_{P_2,op}^{max}$ [t/h]	81,9	0,33	0,40	Sí
$Cost_{T_1,op}^{min}$ [USD\$]	0,0	0,23	0,36	Sí
$Cost_{T_1,op}^{max}$ [USD\$]	10^{50}	0,96	0,32	No
$Cost_{T_2,op}^{min}$ [USD\$]	0,0	0,22	0,32	No
$Cost_{T_2,op}^{max}$ [USD\$]	10^{50}	0,47	0,35	Sí

A diferencia del problema 1, en este segundo ejemplo se conoce información previa. Específicamente, en los problemas de optimización en que se han empleado como variables los flujos y costos, se han definido como variables no negativas. Esto se ha efectuado considerando una de las mejoras propuestas en el Capítulo 5, permitiendo que se acote la región factible y así definir con mayor precisión los valores óptimos contextualizados al problema en cuestión. Su incorporación permitió determinar que los resultados de la Tabla D.20 sugieren que los valores optimizados pueden ser empleados en la determinación de cinco de los ocho límites a calcular (límite inferior y superior de los cuatro parámetros), cuya implementación se efectuará en el Anexo D.7.

Analizando los resultados obtenidos, se destaca que 10^{50} corresponde al mayor valor que es posible expresar en GAMS, pudiendo tener realmente un valor superior la solución. Dado que este valor máximo permite mantener los valores de las variables binarias en el caso del parámetro $Cost_{T_2}$ y no en $Cost_{T_1}$, entonces para el primer parámetro se podría inferir que no está acotado (no presenta un límite superior), mientras que el segundo posee un límite superior de su rango de estabilidad, a determinar mediante la implementación completa del algoritmo propuesto.

En cuanto al valor de $D_{P_2,op}^{max}$, se destaca que una solución más precisa corresponde a 81,9246, valor que ocasiona un cambio en las variables binarias. Para evitar errores de

redondeo que ocasionen que un valor esté fuera del límite del rango de estabilidad por aproximaciones efectuadas por los software empleados, se ha decidido truncar la solución hasta alcanzar el nivel de precisión de la tolerancia (t) correspondiente. Para el presente parámetro, solamente se analizará hasta el primer decimal, razón por la que se considera como estimación inicial el valor de 81,9. Análogamente, en la búsqueda de un límite inferior la solución sería redondeada al valor superior que presente el nivel de precisión requerido según la tolerancia establecida (t), evitando así que se encuentre fuera del rango de estabilidad por aproximaciones.

D.7. Rango de estabilidad (algoritmo propuesto)

D.7.1. Parámetro D_{P_1}

El límite inferior del rango de estabilidad de D_{P_1} se encuentra expresado en la Tabla D.21, notándose que no se necesitaron evaluaciones desde el valor 0 por corresponder al menor valor posible de la demanda D_{P_1} y que fue obtenido mediante la *Optimización de parámetros*. Por el contrario, para el límite superior de D_{P_1} no fue posible emplear una estimación inicial diferente al valor nominal, dado que la obtenida en el Anexo D.6 genera un cambio en las variables binarias, requiriéndose las iteraciones de la Tabla D.22.

Tabla D.21: Determinación del límite inferior del rango de estabilidad del parámetro D_{P_1} mediante el algoritmo propuesto.

i	D_{P_1}	$y' = y_0?$	D_{P_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	0,0	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 1,37[s]					

Los resultados de las Tablas D.21 y D.22 se contrastan con los obtenidos mediante el algoritmo referencial, resumiendo las diferencias más relevantes en la Tabla D.23. En esta es posible notar que ambos algoritmos alcanzan el mismo rango de estabilidad para D_{P_1} , lográndose en un tiempo similar la determinación del límite superior, mientras que la nueva estimación inicial del algoritmo propuesto ocasiona una reducción del 80 % del tiempo necesario para obtener el límite inferior.

Tabla D.22: Determinación del límite superior del rango de estabilidad del parámetro D_{P_1} mediante el algoritmo propuesto.

i	D_{P_1}	$y' = y_0?$	D_{P_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	50,0	Sí	—	—	1,00
1	60,0	No	50,0	$1,0 > 0,02$	1,29
Ciclo 2: $\Delta l = 1,0$					
0	50,0	Sí	—	—	0,90
1	51,0	Sí	—	—	0,84
2	52,0	Sí	—	—	1,60
3	53,0	Sí	—	—	1,63
4	54,0	Sí	—	—	0,98
5	55,0	Sí	—	—	0,91
6	56,0	Sí	—	—	1,35
7	57,0	Sí	—	—	1,02
8	58,0	No	57,0	$0,1 > 0,02$	1,15
Ciclo 3: $\Delta l = 0,1$					
0	57,0	Sí	—	—	1,01
1	57,1	Sí	—	—	1,30
2	57,2	Sí	—	—	1,38
3	57,3	Sí	—	—	1,23
4	57,4	Sí	—	—	1,34
5	57,5	Sí	—	—	1,35
6	57,6	Sí	—	—	1,11
7	57,7	No	57,6	$0,01 < 0,02$	1,22
Tiempo iteraciones: 22,61[s]					
Tiempo total (iteraciones + opt. de parámetros): 23,55[s]					

Tabla D.23: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de D_{P_1} .

Límite	Valor ref. [t/h]	Valor prop. [t/h]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	0,0	0,0	6,86	1,37	-80,0
Superior	57,6	57,6	23,48	23,55	0,3

D.7.2. Parámetro D_{P_2}

Similar a lo ocurrido con D_{P_1} , el límite inferior del rango de estabilidad de D_{P_2} no requirió iteraciones adicionales desde el valor obtenido mediante la *Optimización de parámetros*, alcanzando como límite el valor 0, según se ilustra en la Tabla D.24.

Tabla D.24: Determinación del límite inferior del rango de estabilidad del parámetro D_{P_2} mediante el algoritmo propuesto.

i	D_{P_2}	$y' = y_0?$	D_{P_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	0,0	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 0,49[s]					

Para la búsqueda del límite superior también se emplea el valor obtenido en el Anexo D.6 como punto inicial, aunque en este caso son requeridas iteraciones adicionales al no corresponder a un límite establecido para D_{P_2} con información previa. Para el límite inferior se conoce que D_{P_2} no puede ser negativo, pero no existe información sobre alguna cota para el límite superior previo a la ejecución del algoritmo. De esta forma, se efectúan las iteraciones presentes en la Tabla D.25 y que finalizan en el mismo valor usado inicialmente, por lo que la justificación de las variaciones tiene relación con alcanzar la tolerancia establecida y así concluir que corresponde efectivamente al límite del rango de estabilidad.

Tabla D.25: Determinación del límite superior del rango de estabilidad del parámetro D_{P_2} mediante el algoritmo propuesto.

i	D_{P_2}	$y' = y_0?$	D_{P_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	81,9	Sí	–	–	0,38
1	91,9	No	81,9	$1,0 > 0,016$	0,33
Ciclo 2: $\Delta l = 1,0$					
0	81,9	Sí	–	–	0,38
1	82,9	No	81,9	$0,1 > 0,016$	1,31
Ciclo 3: $\Delta l = 0,1$					
0	81,9	Sí	–	–	0,43
1	82,0	No	81,9	$0,01 < 0,016$	1,00
Tiempo iteraciones: 3,83[s]					
Tiempo total (iteraciones + opt. de parámetros): 4,56[s]					

Al comparar los resultados obtenidos mediante el algoritmo propuesto y el referencial, se observa que en ambos casos se alcanza el mismo rango de estabilidad, tal como ocurrió con D_{P_1} . Sin embargo, en la Tabla D.26 se destaca que el valor alcanzado como límite superior mediante el algoritmo propuesto corresponde a aquel que cumple con la tolerancia

definida, mientras que el uso del algoritmo referencial no lo asegura. Específicamente, la versión referencial sugiere que el usuario defina si desea continuar la búsqueda, por lo que el algoritmo propuesto consigue disminuir la interacción con el usuario y obtener una solución final para este caso, a diferencia del algoritmo referencial.

Tabla D.26: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de D_{P_2} .

Límite	Valor ref. [t/h]	Valor prop. [t/h]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	0,0	0,0	5,35	0,49	-90,8
Superior	81,9 ^a	81,9	28,21	4,56	-83,8

^a Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

Finalmente, se resalta que para D_{P_2} fue posible emplear una nueva estimación inicial para los dos límites del rango de estabilidad, razón por la que se consiguió una disminución del 90,8% en el tiempo requerido para determinar el límite inferior y del 83,8% en el caso del límite superior.

D.7.3. Parámetro $Cost_{T_1}$

El cálculo del límite inferior inicia en el valor obtenido en el Anexo D.6, siendo el menor posible para $Cost_{T_1}$. Como mantiene la solución, no se necesitan nuevas evaluaciones, tal como se resume en la Tabla D.27. Por el contrario, para el límite superior se itera desde el valor nominal tras el análisis del Anexo D.6, requiriéndose las iteraciones de la Tabla D.28.

Tabla D.27: Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_1}$ mediante el algoritmo propuesto.

i	$Cost_{T_1}$	$y' = y_0?$	$Cost_{T_1}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	0,0	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 0,59[s]					

Tabla D.28: Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_1}$ mediante el algoritmo propuesto.

i	$Cost_{T_1}$	$y' = y_0?$	$Cost_{T_1}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	1.000.000	Sí	–	–	0,98
1	2.000.000	No	1.000.000	100.000 > 400	0,60
Ciclo 2: $\Delta l = 100.000$					
0	1.000.000	Sí	–	–	0,99
1	1.100.000	No	1.000.000	10.000 > 400	0,43
Ciclo 3: $\Delta l = 10.000$					
0	1.000.000	Sí	–	–	0,86
1	1.010.000	Sí	–	–	0,80
2	1.020.000	No	1.010.000	1.000 > 400	2,18
Ciclo 4: $\Delta l = 1.000$					
0	1.010.000	Sí	–	–	0,87
1	1.011.000	Sí	–	–	0,71
2	1.012.000	Sí	–	–	0,88
3	1.013.000	Sí	–	–	0,91
4	1.014.000	Sí	–	–	1,01
5	1.015.000	Sí	–	–	2,49
6	1.016.000	Sí	–	–	3,40
7	1.017.000	Sí	–	–	2,73
8	1.018.000	Sí	–	–	2,29
9	1.019.000	No	1.018.000	100 < 400	2,17
Tiempo iteraciones: 24,30[s]					
Tiempo total (iteraciones + opt. de parámetros): 25,58[s]					

Contrastando los resultados del algoritmo referencial y el propuesto, se resalta en la Tabla D.29 que el uso de información previa basada en consideraciones físicas o lógicas favorece a la disminución del tiempo de determinación del rango de estabilidad. Específicamente, en el límite inferior se observa que el algoritmo referencial finaliza evaluando valores negativos, que no son posibles, mientras que el propuesto finaliza por aquella razón en 0. Esta información previa en conjunto con la *Optimización de parámetros* permiten que se disminuya en un 87% el tiempo necesario para determinar el límite inferior.

Tabla D.29: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $Cost_{T_1}$.

Límite	Valor ref. [USD\$]	Valor prop. [USD\$]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	-1.100.000	0	4,54	0,59	-87,0
Superior	1.018.000	1.018.000	25,47	25,58	0,4

Respecto al límite superior, se alcanza el mismo valor en un tiempo similar para ambos algoritmos, sin que se observe un beneficio por el uso del algoritmo propuesto a diferencia de lo ocurrido con el límite inferior. De todas formas, al observar en conjunto el procedimiento para determinar ambos límites, es decir, para obtener el rango de estabilidad, se logra una disminución en el tiempo total requerido empleando el algoritmo propuesto en contraste con el referencial.

D.7.4. Parámetro $Cost_{T_2}$

Para obtener el límite inferior del rango de estabilidad de $Cost_{T_2}$ se realizaron las iteraciones que se encuentran en la Tabla D.30, tras descartar el uso de una nueva estimación inicial proveniente de la *Optimización de parámetros*. En contraste, para la determinación del límite superior se emplea como valor inicial el obtenido en el Anexo D.6, correspondiente al valor máximo expresado en GAMS, originando que no se requieran iteraciones adicionales, tal como se refleja en la Tabla D.31.

Tabla D.30: Determinación del límite inferior del rango de estabilidad del parámetro $Cost_{T_2}$ mediante el algoritmo propuesto.

i	$Cost_{T_2}$	$y' = y_0?$	$Cost_{T_2}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	1.100.000	Sí	–	–	0,84
1	100.000	No	1.100.000	100.000 > 400	0,40
Ciclo 2: $\Delta l = 100.000$					
0	1.100.000	Sí	–	–	1,26
1	1.000.000	No	1.100.000	10.000 > 400	0,47
Ciclo 3: $\Delta l = 10.000$					
0	1.100.000	Sí	–	–	1,73
1	1.090.000	Sí	–	–	1,00
2	1.080.000	No	1.090.000	1.000 > 400	3,30
Ciclo 4: $\Delta l = 1.000$					
0	1.090.000	Sí	–	–	0,73
1	1.089.000	Sí	–	–	0,88
2	1.088.000	Sí	–	–	0,88
3	1.087.000	Sí	–	–	0,83
4	1.086.000	Sí	–	–	0,68
5	1.085.000	Sí	–	–	1,35
6	1.084.000	Sí	–	–	0,83
7	1.083.000	Sí	–	–	2,28
8	1.082.000	Sí	–	–	2,30
9	1.081.000	Sí	–	–	2,25
10	1.080.000	No	1.081.000	100 < 400	2,20
Tiempo iteraciones: 24,21[s]					
Tiempo total (iteraciones + opt. de parámetros): 24,75[s]					

Tabla D.31: Determinación del límite superior del rango de estabilidad del parámetro $Cost_{T_2}$ mediante el algoritmo propuesto.

i	$Cost_{T_2}$	$y' = y_0?$	$Cost_{T_2}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000.000$					
0	10^{50}	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 0,82[s]					

Efectuando una comparación entre los resultados obtenidos en esta sección respecto a lo determinado mediante el algoritmo referencial, que se resume en la Tabla D.32, es posible notar que se alcanza el mismo límite inferior pero un límite superior diferente.

Tabla D.32: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $Cost_{T_2}$.

Límite	Valor ref. [USD\$]	Valor prop. [USD\$]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	1.081.000 ^a	1.081.000	20,99	24,75	17,9
Superior	10.100.000 ^a	NA ^b	15,37	0,82	–94,7

^a Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

^b NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresado en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

En cuanto al tiempo requerido para obtener el límite inferior, es mayor con el uso del algoritmo propuesto, tras haber evaluado la posibilidad de emplear una nueva estimación y descartarse. Sin embargo, se destaca que el algoritmo propuesto consideró un mayor número de evaluaciones para finalizar debido a la tolerancia impuesta inicialmente, mientras que la versión referencial al detenerse sugiere actualizar los parámetros de búsqueda sin alcanzar el criterio de tolerancia, razón por la que la continuación por parte del usuario podría haber generado un mayor tiempo utilizado para las iteraciones.

Por otro lado, el nuevo punto inicial determinado para la búsqueda del límite superior logra una disminución del tiempo al utilizar el algoritmo propuesto en vez del referencial. Específicamente, se logra debido a que alcanza el mayor valor posible en GAMS, mientras que el algoritmo referencial efectuó múltiples iteraciones hasta detenerse y sugerir que se actualicen los parámetros de búsqueda en caso de requerir continuar. De esta forma, con la versión propuesta se disminuye el tiempo total y la interacción con el usuario.

D.8. Índice de estabilidad

La propuesta efectuada en el Capítulo 5 para determinar los índices de estabilidad inferior y superior se aplica en el presente ejemplo, empleando los valores nominales y los rangos calculados en la sección D.7, obteniéndose los resultados de la Tabla D.33:

Tabla D.33: Índice de estabilidad inferior y superior de los parámetros del ejemplo 2.

Par. [UN]	Nominal	Rango de estabilidad	IE^{inf} [%]	IE^{sup} [%]
D_{P_1} [t/h]	50	$0 \leq D_{P_1} \leq 57,6$	100,0	15,2
D_{P_2} [t/h]	40	$0 \leq D_{P_2} \leq 81,9$	100,0	104,8
$Cost_{T_1}$ [USD\$]	1.000.000	$0 \leq Cost_{T_1} \leq 1.018.000$	100,0	1,8
$Cost_{T_2}$ [USD\$]	1.100.000	$1.081.000 \leq Cost_{T_2}$	1,7	NA ^a

^a NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresable en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

Los resultados anteriores reflejan una situación similar a lo ocurrido en el ejemplo 1: $Cost_{T_2}$ no está acotado superiormente, pero es el parámetro con mayor cercanía entre su valor nominal y uno de los extremos del rango de estabilidad, ocasionando que sea el parámetro crítico a analizar para evitar que las estimaciones efectuadas generaran que la solución óptima fuese otra. Específicamente, un error superior al 1,7% en aquel parámetro ocasionaría que el layout óptimo fuese diferente.

Se observa que el uso del índice de estabilidad a partir de variaciones relativas permite comparar parámetros de diferentes órdenes de magnitud, ya que $Cost_{T_1}$ y $Cost_{T_2}$ presentan rangos de mayor amplitud, pero los cambios relativos en estos generan que sean los parámetros críticos en el contexto dado, considerando que $Cost_{T_1}$ presenta un índice de estabilidad cercano al otro costo, con un valor de 1,8%.

Anexo E

Ejemplo ilustrativo 3

E.1. Formulación

E.1.1. Conjuntos

- $J = \{A, B, C, D\}$: Conjunto de contaminantes.
- $PU = \{P_1, P_2\}$: Conjunto de unidades de procesos.
- $TU = \{T_1, T_2, T_3\}$: Conjunto de unidades de tratamiento.

E.1.2. Parámetros

Parámetros operacionales

- $C_{des,j}$: Concentración máxima de $j \in J$ permitida en la descarga.
- $C_{p,j}^{max}$: Concentración máxima permitida de $j \in J$ en la entrada del proceso $p \in PU$.
- $C_{FW,j}$: Concentración de contaminante $j \in J$ de la fuente.
- $CTU_{t,j}^{in,max}$: Concentración máxima de contaminante $j \in J$ permitida en la entrada de la unidad de tratamiento $t = \{T_1, T_2\} \in TU$.
- $CTU_{t,j}^{out,max}$: Concentración máxima de contaminante $j \in J$ en la salida de la unidad de tratamiento $t = \{T_1, T_2\} \in TU$.
- FPU_p^{in} : Flujo de entrada al proceso $p \in PU$.
- $L_{p,j}$: Carga de contaminante $j \in J$ en el proceso $p \in PU$.
- r_{T_3} : Porcentaje de remoción de contaminantes en la unidad de tratamiento $T_3 \in TU$.

Parámetros económicos

- AR : Factor anualizado de inversión para unidades de tratamiento.
- $Cost_{FW}$: Costo de agua de la fuente.
- CP : Costo fijo para cada tubería en la red.
- H : Horas de operación de las plantas por año.

- IC_t : Coeficiente de costo de inversión de la unidad de tratamiento $t \in TU$.
- IP : Costo variable de cada tubería individual en la red.
- OC_t : Coeficiente de costo de operación de la unidad de tratamiento $t \in TU$.
- PM : Coeficiente de costo de operación de bombeo de aguas en cada tubería de la red.
- α : Exponente del costo de las unidades de tratamiento.
- γ : Exponente del costo de las tuberías.

Cotas

- $CPU_{p,j}^{out,L}$: Cota inferior de la concentración de $j \in J$ en la salida de la unidad $p \in PU$.
- $CPU_{p,j}^{out,U}$: Cota superior de la concentración de $j \in J$ en la salida de la unidad $p \in PU$.
- $CTU_{T_3,j}^{in,U}$: Cota superior de la concentración de $j \in J$ en la entrada de la unidad $T_3 \in TU$.
- $CTU_{T_3,j}^{out,U}$: Cota superior de la concentración de $j \in J$ en la salida de la unidad $T_3 \in TU$.
- FIF^L : Cota inferior del flujo FIF .
- FIF^U : Cota superior del flujo FIF .
- FIP_p^L : Cota inferior del flujo FIP_p , con $p \in PU$.
- FIP_p^U : Cota superior del flujo FIP_p , con $p \in PU$.
- FIT_t^L : Cota inferior del flujo FIT_t , con $t \in TU$.
- FIT_t^U : Cota superior del flujo FIT_t , con $t \in TU$.
- FPO_p^L : Cota inferior del flujo FPO_p , con $p \in PU$.
- FTO_t^L : Cota inferior del flujo FTO_t , con $t \in TU$.
- $FP_{p,p'}^L$: Cota inferior del flujo $FP_{p,p'}$, con $p, p' \in PU$.
- $FPT_{p,t}^L$: Cota inferior del flujo $FPT_{p,t}$, con $p \in PU$ y $t \in TU$.
- $FT_{t,t'}^L$: Cota inferior del flujo $FT_{t,t'}$, con $t, t' \in TU$.
- $FTP_{t,p}^L$: Cota inferior del flujo $FTP_{t,p}$, con $t \in TU$ y $p \in PU$.
- FW^L : Cota inferior del flujo proveniente de la fuente.
- FW^U : Cota superior del flujo proveniente de la fuente.

Otros

- ϵ : Tolerancia para la función de costos de inversión de tuberías y unidades de tratamiento.

Los valores nominales de los parámetros se encuentran en las Tablas E.1 y E.2:

Tabla E.1: Valores nominales de los parámetros operacionales del ejemplo ilustrativo 3.

Parámetro	Valor	Unidad	Parámetro	Valor	Unidad
$C_{des,A}$	10	[ppm]	$CTU_{T_2,C}^{in,max}$	10	[ppm]
$C_{des,B}$	10	[ppm]	$CTU_{T_2,D}^{in,max}$	10	[ppm]
$C_{des,C}$	10	[ppm]	$CTU_{T_1,A}^{out,max}$	10	[ppm]
$C_{des,D}$	10	[ppm]	$CTU_{T_1,B}^{out,max}$	10	[ppm]
$C_{P_1,A}^{max}$	0	[ppm]	$CTU_{T_1,C}^{out,max}$	10	[ppm]
$C_{P_1,B}^{max}$	0	[ppm]	$CTU_{T_1,D}^{out,max}$	10	[ppm]
$C_{P_1,C}^{max}$	0	[ppm]	$CTU_{T_2,A}^{out,max}$	0	[ppm]
$C_{P_1,D}^{max}$	0	[ppm]	$CTU_{T_2,B}^{out,max}$	0	[ppm]
$C_{P_2,A}^{max}$	50	[ppm]	$CTU_{T_2,C}^{out,max}$	0	[ppm]
$C_{P_2,B}^{max}$	50	[ppm]	$CTU_{T_2,D}^{out,max}$	0	[ppm]
$C_{P_2,C}^{max}$	50	[ppm]	$FPU_{P_1}^{in}$	40	[t/h]
$C_{P_2,D}^{max}$	50	[ppm]	$FPU_{P_2}^{in}$	50	[t/h]
$C_{FW,A}$	100	[ppm]	$L_{P_1,A}^{in}$	1.000	[g/h]
$C_{FW,B}$	100	[ppm]	$L_{P_1,B}^{in}$	1.500	[g/h]
$C_{FW,C}$	100	[ppm]	$L_{P_1,C}^{in}$	1.000	[g/h]
$C_{FW,D}$	100	[ppm]	$L_{P_1,D}^{in}$	1.000	[g/h]
$CTU_{T_1,A}^{in,max}$	100	[ppm]	$L_{P_2,A}^{in}$	1.000	[g/h]
$CTU_{T_1,B}^{in,max}$	100	[ppm]	$L_{P_2,B}^{in}$	1.000	[g/h]
$CTU_{T_1,C}^{in,max}$	100	[ppm]	$L_{P_2,C}^{in}$	1.000	[g/h]
$CTU_{T_1,D}^{in,max}$	100	[ppm]	$L_{P_2,D}^{in}$	1.000	[g/h]
$CTU_{T_2,A}^{in,max}$	10	[ppm]	r_{T_3}	95	[%]
$CTU_{T_2,B}^{in,max}$	10	[ppm]			

Tabla E.2: Valores nominales de parámetros económicos y otros del ejemplo ilustrativo 3.

Parámetro	Valor	Unidad	Parámetro	Valor	Unidad
AR	0,1	[-]	OC_{T_1}	0,1	[\$/t]
$Cost_{FW}$	0,1	[\$/t]	OC_{T_2}	1,15	[\$/t]
CP	6	[\$]	OC_{T_3}	1,80	[\$/t]
H	8.600	[h]	PM	0,006	[\$/t]
IC_{T_1}	10.000	[] ^a	α	0,7	[-]
IC_{T_2}	25.000	[] ^a	γ	0,7	[-]
IC_{T_3}	30.000	[] ^a	ϵ	0,001	[t/h]
IP	100	[] ^b			

^a El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [\$]

^b El componente variable del costo de inversión en tuberías, $IP \cdot (Flujos)^\gamma$, posee unidades en [\$]. En la expresión anterior, *Flujos* representa todos aquellos caudales de posibles tuberías a instalar.

E.1.3. Variables

Continuas y positivas

- $C_{out,j}$: Concentración de contaminante $j \in J$ en la salida del mezclador de flujo final (descarga).
- CSI_j : Concentración de contaminante $j \in J$ en la salida del separador de flujo inicial (fuente de agua).
- $CPU_{p,j}^{in}$: Concentración de contaminante $j \in J$ en la entrada del proceso $p \in PU$.
- $CPU_{p,j}^{out}$: Concentración de contaminante $j \in J$ en la salida del proceso $p \in PU$.
- $CSPU_{p,j}^{out}$: Concentración de contaminante $j \in J$ en la salida del separador de flujo del proceso $p \in PU$.
- $CSTU_{t,j}^{out}$: Concentración de contaminante $j \in J$ en la salida del separador de flujo del tratamiento $t \in TU$.
- $CTU_{t,j}^{in}$: Concentración de contaminante $j \in J$ en la entrada del tratamiento $t \in TU$.
- $CTU_{t,j}^{out}$: Concentración de contaminante $j \in J$ en la salida del tratamiento $t \in TU$.
- F_{out} : Caudal de agua residual del mezclador de flujo final (descarga).
- FIF : Caudal de la fuente al mezclador de flujo final.
- FIP_p : Caudal de la fuente al proceso $p \in PU$.
- FIT_t : Caudal de la fuente al tratamiento $t \in TU$.
- $FP_{p,p'}$: Caudal del proceso $p \in PU$ enviado a $p' \in PU$.
- FPO_p : Caudal del proceso $p \in PU$ enviado al mezclador de flujo final.
- $FPT_{p,t}$: Caudal del proceso $p \in PU$ enviado al tratamiento $t \in TU$.
- FPU_p^{out} : Caudal de salida del proceso $p \in PU$.
- $FT_{t,t'}$: Caudal del tratamiento $t \in TU$ enviado al tratamiento $t' \in TU$.
- FTO_t : Caudal del tratamiento $t \in TU$ enviado al mezclador de flujo final.
- $FTP_{t,p}$: Caudal del tratamiento $t \in TU$ enviado al proceso $p \in PU$.
- FTU_t^{in} : Caudal de entrada al tratamiento $t \in TU$.
- FTU_t^{out} : Caudal de salida del tratamiento $t \in TU$.
- FW : Caudal de salida de la fuente de agua.

Binarias

- y_{FIF} : Existencia de conexión entre la fuente y el mezclador de flujo final.
- y_{FIP_p} : Existencia de conexión entre la fuente y la unidad de proceso $p \in PU$.
- y_{FIT_t} : Existencia de conexión entre la fuente y la unidad de tratamiento $t \in TU$.
- $y_{FP_{p,p'}}$: Existencia de conexión entre el proceso $p \in PU$ y el proceso $p' \in PU$.
- y_{FPO_p} : Existencia de conexión entre el proceso $p \in PU$ y el mezclador de flujo final.
- $y_{FPT_{p,t}}$: Existencia de conexión entre el proceso $p \in PU$ y el tratamiento $t \in TU$.
- $y_{FT_{t,t'}}$: Existencia de conexión entre el tratamiento $t \in TU$ y el tratamiento $t' \in TU$.
- y_{FTO_t} : Existencia de conexión entre el tratamiento $t \in TU$ y el mezclador de flujo final.
- $y_{FTP_{t,p}}$: Existencia de conexión entre el tratamiento $t \in TU$ y el proceso $p \in PU$.

E.1.4. Restricciones

Balances de masa

- Balance global:

Como parte de la estrategia de resolución realizada por Ahmetovic y Grossmann [73], basada en las cotas propuestas por Karuppiah y Grossmann [72], se incluye el balance global por especies:

$$FW \cdot C_{FW} + \sum_{p \in PU} L_{p,j} = \sum_{t \in TU} (F_{t,j} \cdot CTU_{t,j}^{in} - FTU_{t,j}^{out} \cdot CTU_{t,j}^{out}) + F_{out} \cdot C_{out,j} \quad \forall j \in J \quad (E.1)$$

- Balance por unidad/equipo:

Respecto a la extracción de agua de la fuente, posee tres tipos de posibles destinos según se plantea en la siguiente ecuación:

$$FW = FIF + \sum_{p \in PU} FIP_p + \sum_{t \in TU} FIT_t \quad (E.2)$$

Para el caso de las unidades de procesos, se consideran los balances de masa del mezclador previo a su entrada, la unidad de proceso particular y el separador de flujo que lo sigue.

$$FPU_p^{in} = FIP_p + \sum_{p' \in PU} FP_{p',p} + \sum_{t \in TU} FPT_{t,p} \quad \forall p \in PU \quad (E.3)$$

$$FPU_p^{in} = FPU_p^{out} \quad \forall p \in PU \quad (E.4)$$

$$FPU_p^{out} = FPO_p + \sum_{p' \in PU} FP_{p,p'} + \sum_{t \in TU} FPT_{p,t} \quad \forall p \in PU \quad (E.5)$$

En cuanto al tratamiento del agua, las etapas involucradas corresponden a un mezclador de flujo, la unidad de tratamiento y un posterior separador de flujo. En estos tres tipos

de nodos de la red son planteados los siguientes balances:

$$FTU_t^{in} = FIT_t + \sum_{p \in PU} FPT_{p,t} + \sum_{t' \in TU} FT_{t',t} \quad \forall t \in TU \quad (E.6)$$

$$FTU_t^{in} = FTU_t^{out} \quad \forall t \in TU \quad (E.7)$$

$$FTU_t^{out} = FTO_t + \sum_{t' \in TU} FT_{t,t'} + \sum_{p \in PU} FTP_{t,p} \quad \forall t \in TU \quad (E.8)$$

Finalmente, para determinar el flujo de descarga al ambiente se consideran aquellos que ingresan al mezclador de flujo final:

$$F_{out} = FIF + \sum_{p \in PU} FPO_p + \sum_{t \in TU} FTO_t \quad (E.9)$$

- Balance por especies:

El balance por especies de la fuente de agua se plantea en el separador de flujo inicial, donde cada salida presenta la misma concentración que la entrada:

$$CSI_j = C_{FW,j} \quad \forall j \in J \quad (E.10)$$

Respecto a las unidades de procesos, nuevamente se consideran los balances en el mezclador de flujo, la unidad de procesos y el separador de flujo correspondientes. Particularmente, para este último equipo se conoce que la concentración en cada salida es igual a la entrada, razón por la que el balance de masa es reemplazado por aquella consideración.

$$FPU_p^{in} \cdot CPU_{p,j}^{in} = FIP_p \cdot CSI_j + \sum_{p' \in PU} FP_{p,p'} \cdot CSPU_{p',j}^{out} + \sum_{t \in TU} FTP_{t,p} \cdot CSTU_{t,j}^{out} \quad \forall j \in J, \quad \forall p \in PU \quad (E.11)$$

$$FPU_p^{in} \cdot CPU_{p,j}^{in} + L_{p,j} = FPU_p^{out} \cdot CPU_{p,j}^{out} \quad \forall j \in J, \quad \forall p \in PU \quad (E.12)$$

$$CSPU_{p,j}^{out} = CPU_{p,j}^{out} \quad \forall j \in J, \quad \forall p \in PU \quad (E.13)$$

Para el caso del tratamiento de agua en la red, nuevamente se consideran tres unidades: mezclador, tratamiento y separador de flujo. Los balances por especies se plantean a continuación, incluyendo nuevamente la condición sobre las concentraciones de los flujos involucrados en el separador. Adicionalmente, para las etapas de tratamiento los balances de masa por especies son reemplazados por las condiciones impuestas en las concentraciones de salida luego de ser tratados los flujos.

$$FTU_t^{in} \cdot CTU_{t,j}^{in} = FIT_t \cdot CSI_j + \sum_{p \in PU} FPT_{p,t} \cdot CSPU_{p,j}^{out} + \sum_{t' \in TU} FT_{t',t} \cdot CSTU_{t',j}^{out} \quad \forall j \in J, \quad \forall t \in TU \quad (E.14)$$

Para los pretratamientos T_1 y T_2 , se asume que son capaces de reducir la concentración de cada contaminante $j \in J$ a $10[ppm]$ y $0[ppm]$, respectivamente. Para el tratamiento T_3 , se incluye el porcentaje de remoción (r_{T_3}) para determinar la concentración de salida.

$$CTU_{t,j}^{out} = CTU_{t,j}^{out,max} \quad \forall j \in J, \quad t = \{T_1, T_2\} \in TU \quad (E.15)$$

$$CTU_{t,j}^{out} = \left(1 - \frac{r_t}{100}\right) \cdot CTU_{t,j}^{in} \quad \forall j \in J, \quad t = \{T_3\} \in TU \quad (E.16)$$

$$CSTU_{t,j}^{out} = CTU_{t,j}^{out} \quad \forall j \in J, \quad \forall t \in TU \quad (E.17)$$

Finalmente, el balance por especies para la descarga al ambiente se plantea de la siguiente forma:

$$F_{out} \cdot C_{out,j} = FIF \cdot CSI_j + \sum_{p \in PU} FPO_p \cdot CSPU_{p,j}^{out} + \sum_{t \in TU} FTO_t \cdot CSTU_{t,j}^{out} \quad \forall j \in J \quad (E.18)$$

Operacionales

- Calidad de entrada:

Existen limitaciones sobre la concentración de contaminantes de entrada a las unidades de proceso (P_1 y P_2) y a dos plantas de tratamiento (T_1 y T_2), que se observan en las siguientes expresiones:

$$CPU_{p,j}^{in} \leq C_{p,j}^{max} \quad \forall j \in J, \quad \forall p \in PU \quad (E.19)$$

$$CTU_{t,j}^{in} \leq CTU_{t,j}^{in,max} \quad \forall j \in J, \quad t = \{T_1, T_2\} \in TU \quad (E.20)$$

- Calidad de salida:

La descarga al ambiente está restringida por una concentración máxima permitida:

$$C_{out,j} \leq C_{des,j} \quad \forall j \in J \quad (E.21)$$

Lógicas

- Existencia de conexiones:

Para que se pueda realizar una transferencia del flujo desde un nodo de la red a otro, debe existir una tubería que los conecte. Es por ello que se incorporan las siguientes relaciones lógicas:

$$FIF^L \cdot y_{FIF} \leq FIF \quad (E.22)$$

$$FIF \leq FIF^U \cdot y_{FIF} \quad (E.23)$$

$$FIP_p^L \cdot y_{FIP_p} \leq FIP_p \quad \forall p \in PU \quad (E.24)$$

$$FIP_p \leq FIP_p^U \cdot y_{FIP_p} \quad \forall p \in PU \quad (E.25)$$

$$FIT_t^L \cdot y_{FIT_t} \leq FIT_t \quad \forall t \in TU \quad (E.26)$$

$$FIT_t \leq FIT_t^U \cdot y_{FIT_t} \quad \forall t \in TU \quad (E.27)$$

$$FPO_p^L \cdot y_{FPO_p} \leq FPO_p \quad \forall p \in PU \quad (E.28)$$

$$FPO_p \leq FPO_p^U \cdot y_{FPO_p} \quad \forall p \in PU \quad (E.29)$$

$$FPT_{p,t}^L \cdot y_{FPT_{p,t}} \leq FPT_{p,t} \quad \forall p \in PU, \quad \forall t \in TU \quad (E.30)$$

$$FPT_{p,t} \leq FPT_{p,t}^U \cdot y_{FPT_{p,t}} \quad \forall p \in PU, \quad \forall t \in TU \quad (E.31)$$

$$FP_{p,p'}^L \cdot y_{FP_{p,p'}} \leq FP_{p,p'} \quad \forall p, p' \in PU \quad (E.32)$$

$$FP_{p,p'} \leq FP_{p,p'}^U \cdot y_{FP_{p,p'}} \quad \forall p, p' \in PU \quad (E.33)$$

$$FTP_{t,p}^L \cdot y_{FTP_{t,p}} \leq FTP_{t,p} \quad \forall t \in TU, \quad \forall p \in PU \quad (E.34)$$

$$FTP_{t,p} \leq FTP_{t,p}^U \cdot y_{FTP_{t,p}} \quad \forall t \in TU, \quad \forall p \in PU \quad (E.35)$$

$$FTO_t^L \cdot y_{FTO_t} \leq FTO_t \quad \forall t \in TU \quad (E.36)$$

$$FTO_t \leq FTO_t^U \cdot y_{FTO_t} \quad \forall t \in TU \quad (E.37)$$

$$FT_{t,t'}^L \cdot y_{FT_{t,t'}} \leq FT_{t,t'} \quad \forall t, t' \in TU \quad (E.38)$$

$$FT_{t,t'} \leq FT_{t,t'}^U \cdot y_{FT_{t,t'}} \quad \forall t, t' \in TU \quad (E.39)$$

- Cotas adicionales:

Utilizando especificaciones lógicas e inspección física, Ahmetovic y Grossmann [73] incluyeron nuevas restricciones para acelerar la búsqueda de la solución óptima. Las cotas superiores e inferiores sobre las variables han sido incorporadas como parámetros, siendo estos calculados previamente a la resolución del problema. Con ello, se plantean las expresiones que se observan a continuación, mientras que las ecuaciones con las que se determinan los valores de las cotas y su explicación se encuentran en la sección E.1.6.

$$FW \leq FW^U \quad (E.40)$$

$$FIT_t \leq FIT_t^U \quad \forall t \in TU \quad (E.41)$$

$$CPU_{p,j}^{out,L} \leq CPU_{p,j}^{out} \quad \forall j \in J, \quad \forall p \in PU \quad (E.42)$$

$$CPU_{p,j}^{out} \leq CPU_{p,j}^{out,U} \quad \forall j \in J, \quad \forall p \in PU \quad (E.43)$$

$$FP_{p,p'} \leq FP_{p,p'}^U \quad \forall p, p' \in PU \quad (E.44)$$

$$FPT_{p,t} \leq FPT_{p,t}^U \quad \forall p \in PU, \quad \forall t \in TU \quad (E.45)$$

$$FPO_p \leq FPO_p^U \quad \forall p \in PU \quad (E.46)$$

$$FTU_t^{in} \leq FTU_t^{in,U} \quad \forall t \in TU \quad (E.47)$$

$$FTU_t^{out} \leq FTU_t^{out,U} \quad \forall t \in TU \quad (E.48)$$

$$CTU_{t,j}^{in} \leq CTU_{t,j}^{in,U} \quad \forall j \in J, \quad \forall t \in TU \quad (E.49)$$

$$CTU_{t,j}^{out} \leq CTU_{t,j}^{out,U} \quad \forall j \in J, \quad t = \{T_3\} \in TU \quad (E.50)$$

$$FT_{t,t'} \leq FT_{t,t'}^U \quad \forall t, t' \in TU \quad (E.51)$$

$$FTP_{t,p} \leq FTP_{t,p}^U \quad \forall t \in TU, \quad \forall p \in PU \quad (E.52)$$

$$FTO_t \leq FTO_t^U \quad \forall t \in TU \quad (E.53)$$

$$F_{out} \leq F_{out}^U \quad (E.54)$$

E.1.5. Función objetivo

La función objetivo del problema corresponde a la minimización del costo total anualizado de la red, considerando el costo del agua obtenida de la fuente (C_{agua}), la instalación y operación de las unidades de tratamiento (IC_{trat} y OC_{trat} , respectivamente), y la instalación de tuberías (IC_{tub}) con el respectivo costo de bombeo de agua a través de estas (OC_{bomb}). La expresión matemática de la función objetivo se observa a continuación, incluyendo una tolerancia ϵ en aquellos flujos que podrían generar errores si se evalúa el valor nulo:

$$F.O. = C_{agua} + IC_{trat} + OC_{trat} + OC_{bomb} + IC_{tub} \quad (E.55)$$

$$C_{agua} = H \cdot FW \cdot Cost_{FW} \quad (E.55a)$$

$$IC_{trat} = AR \cdot \sum_{t \in TU} IC_t \cdot (FTU_t^{out} + \epsilon)^\alpha \quad (E.55b)$$

$$OC_{trat} = H \cdot \sum_{t \in TU} OC_t \cdot FTU_t^{out} \quad (E.55c)$$

$$OC_{bomb} = H \cdot \left[\sum_{p \in PU} FIP_p \cdot PM + \sum_{t \in TU} FIT_t \cdot PM + \sum_{t \in TU} \sum_{p \in PU} FTP_{t,p} \cdot PM \right. \\ \left. + \sum_{p \in PU} \sum_{p' \in PU, p \neq p'} FP_{p,p'} \cdot PM + \sum_{p \in PU} \sum_{t \in TU} FPT_{p,t} \cdot PM \right. \quad (E.55d)$$

$$\left. + \sum_{p \in PU} FPO_p \cdot PM + \sum_{t \in TU} FTO_t \cdot PM + \sum_{t \in TU} \sum_{t' \in TU, t \neq t'} FT_{t,t'} \cdot PM \right] \\ IC_{tub} = AR \cdot \left[\sum_{t \in TU} \sum_{p \in PU} (CP \cdot y_{FTP_{t,p}} + IP \cdot (FTP_{t,p} + \epsilon)^\gamma) \right. \\ \left. + \sum_{p \in PU} \sum_{p' \in PU, p \neq p'} (CP \cdot y_{FP_{p,p'}} + IP \cdot (FP_{p,p'} + \epsilon)^\gamma) \right. \\ \left. + \sum_{p \in PU} \sum_{t \in TU} (CP \cdot y_{FPT_{p,t}} + IP \cdot (FPT_{p,t} + \epsilon)^\gamma) \right. \\ \left. + \sum_{p \in PU} (CP \cdot y_{FPO_p} + IP \cdot (FPO_p + \epsilon)^\gamma) + \sum_{t \in TU} (CP \cdot y_{FTO_t} + IP \cdot (FTO_t + \epsilon)^\gamma) \right. \\ \left. + \sum_{t \in TU} \sum_{t' \in TU, t \neq t'} (CP \cdot y_{FT_{t,t'}} + IP \cdot (FT_{t,t'} + \epsilon)^\gamma) \right. \\ \left. + \sum_{p \in PU} (CP \cdot y_{FIP_p} + IP \cdot (FIP_p + \epsilon)^\gamma) + \sum_{t \in TU} (CP \cdot y_{FIT_t} + IP \cdot (FIT_t + \epsilon)^\gamma) \right] \quad (E.55e)$$

E.1.6. Ecuaciones adicionales

La determinación de las cotas empleadas para las variables, ingresadas como parámetros en el problema de optimización, se logra mediante las expresiones que se plantean a continuación. Particularmente, se han separado del resto del planteamiento debido a que, por eficiencia computacional, es conveniente que sean calculadas previamente a la resolución del problema de optimización, ingresándose sus resultados directamente como parámetros del problema.

Respecto al flujo a extraer desde la fuente, este debe ser menor o igual a lo requerido en los procesos. Al existir la posibilidad de recircular flujos de agua, no presenta un mínimo (se asume $FW^L = 0,0001$ para las restricciones de existencia de conexiones, supuesto que se aplica para todos los casos en que se planteen posibles conexiones y no exista otra cota inferior).

$$FW^U = \sum_{p \in PU} FPU_p^{in} \quad (E.56)$$

Considerando el flujo máximo a extraer, el mayor caudal que se puede enviar a alguna de las etapas de tratamiento o al mezclador de flujo final corresponde a este mismo valor máximo, dado que está relacionado con el mayor flujo a tratar luego de su paso por las unidades de proceso. En cuanto al flujo a enviar a cada proceso, se limita a lo que requiere cada uno. Adicionalmente, se destaca que no existe un mínimo, considerando que podría no existir extracción de agua de la fuente.

$$FIF^U = \sum_{p \in PU} FPU_p^{in} \quad (E.57)$$

$$FIP_p^U = FPU_p^{in} \quad \forall p \in PU \quad (E.58)$$

$$FIT_t^U = \sum_{p \in PU} FPU_p^{in} \quad \forall t \in TU \quad (E.59)$$

En el caso de las unidades de procesos, el flujo que ingresa y sale de estos equipos son constantes y conocidos, razón por la que no se requieren cotas. Sin embargo, es posible restringir el rango de concentraciones de los flujos de estas unidades. Particularmente, la cota superior para la concentración de entrada se encuentra definida por la restricción relacionada con la concentración máxima que permite cada proceso, no siendo necesario agregar otra restricción. Por el contrario, es posible añadir cotas para la concentración de salida, debido a que se conoce la carga de contaminante en cada proceso ($L_{p,j}$) y la concentración máxima de entrada ($C_{p,j}^{max}$). En las ecuaciones E.60 y E.61 se ha incluido una holgura $Z = 1$ que no estaba considerada en el problema original, siendo añadida debido a infactibilidades que se producen al redondear las fracciones que componen ambas expresiones. En el problema original no se requiere, pero las variaciones de los flujos en distintas iteraciones del algoritmo para determinar el rango de estabilidad produce aquel inconveniente, que es cubierto con la opción de flexibilizar el valor inferior y superior de las concentraciones de salida.

$$CPU_{p,j}^{out,L} = \frac{LPU_{p,j}}{FPU_p^{in}} - Z \quad \forall j \in J, \quad \forall p \in PU \quad (E.60)$$

$$CPU_{p,j}^{out,U} = \frac{C_{p,j}^{max} \cdot FPU_p^{in} + LPU_{p,j}}{FPU_p^{in}} + Z \quad \forall j \in J, \quad \forall p \in PU \quad (E.61)$$

Respecto al separador de flujo de las unidades de procesos, los caudales involucrados están limitados por las capacidades de cada proceso. En el caso del flujo que se envía a otra unidad de proceso, este no puede ser superior a la capacidad del proceso que recibe (al ser solo dos unidades en este ejemplo, podría limitarse directamente por el mínimo de las capacidades). Para el flujo a enviar a tratamiento o disposición, estos se limitan por el caudal del proceso correspondiente.

$$FP_{p,p'}^U = FPU_{p'}^{in} \quad si \quad FPU_{p'}^{in} \leq FPU_p^{in} \quad \forall p, p' \in PU \quad (E.62)$$

$$FP_{p,p'}^U = FPU_p^{in} \quad si \quad FPU_p^{in} \leq FPU_{p'}^{in} \quad \forall p, p' \in PU \quad (E.63)$$

$$FPT_{p,t}^U = FPU_p^{in} \quad \forall p \in PU, \quad \forall t \in TU \quad (E.64)$$

$$FPO_p^U = FPU_p^{in} \quad \forall p \in PU \quad (E.65)$$

Para el caso de las unidades de tratamiento, el flujo máximo de entrada y de salida de estos equipos está restringido por el flujo total de agua a tratar en la red. Respecto a la calidad de estos flujos, se ha limitado la concentración máxima de entrada a dos unidades (T_1 y T_2), que también poseen una concentración de salida preestablecida. Es por ello que no requerirían cotas nuevas, mientras que es necesaria una cota para el tercer tratamiento (T_3), la que está condicionada por la concentración máxima de salida de los procesos (es por ello que se incluye también la holgura Z):

$$FTU_t^{in,U} = \sum_{p \in PU} FPU_p^{in} \quad \forall t \in TU \quad (E.66)$$

$$FTU_t^{out,U} = \sum_{p \in PU} FPU_p^{in} \quad \forall t \in TU \quad (E.67)$$

$$CTU_{t,j}^{in,U} = \max_{p \in PU} \left(\frac{C_{p,j}^{max} \cdot FPU_p^{in} + LPU_{p,j}}{FPU_p^{in}} + Z \right) \quad \forall j \in J, \quad t = \{T_3\} \in TU \quad (E.68)$$

$$CTU_{t,j}^{out,U} = \left(1 - \frac{r_t}{100} \right) \cdot CTU_{t,j}^{in,U} \quad \forall j \in J, \quad t = \{T_3\} \in TU \quad (E.69)$$

Los caudales de los separadores de flujo de las unidades de tratamiento, tal como los equipos que los preceden, están limitados por los flujos de las unidades de procesos:

$$FT_{t,t'}^U = \sum_{p \in PU} FPU_p^{in} \quad \forall t, t' \in TU \quad (E.70)$$

$$FTP_{t,p}^U = FPU_p^{in} \quad \forall t \in TU, \quad \forall p \in PU \quad (E.71)$$

$$FTO_t^U = \sum_{p \in PU} FPU_p^{in} \quad \forall t \in TU \quad (E.72)$$

Finalmente, el flujo de descarga se encuentra limitado por los flujos de los procesos, tal como otros mencionados previamente, debido a la relación que existe entre los diversos flujos de la red. Respecto a las concentraciones, existe una limitación para la calidad de descarga, razón por la que no se adiciona una nueva cota.

$$F_{out}^U = \sum_{p \in PU} FPU_p^{in} \quad (E.73)$$

E.2. Solución óptima

El resultado obtenido para el modelo planteado por Ahmetovic y Grossmann [73] se observa de forma completa en las Tablas E.3, E.4 y E.5. La función objetivo para este caso, correspondiente al costo de la red, es de \$1.156.059 al año.

Tabla E.3: Flujos óptimos del ejemplo 3, empleando los valores nominales de los parámetros.

Variable	Valor	Unidad	Variable	Valor	Unidad
F_{out}	-	[t/h]	FT_{T_1,T_3}	-	[t/h]
FIF	-	[t/h]	FT_{T_2,T_1}	-	[t/h]
FIP_{P_1}	-	[t/h]	FT_{T_2,T_2}	-	[t/h]
FIP_{P_2}	-	[t/h]	FT_{T_2,T_3}	-	[t/h]
FIT_{t_1}	-	[t/h]	FT_{T_3,T_1}	-	[t/h]
FIT_{t_2}	-	[t/h]	FT_{T_3,T_2}	33,77	[t/h]
FIT_{t_3}	-	[t/h]	FT_{T_3,T_3}	-	[t/h]
FP_{P_1,P_1}	-	[t/h]	FTO_{T_1}	-	[t/h]
FP_{P_1,P_2}	40,00	[t/h]	FTO_{T_2}	-	[t/h]
FP_{P_2,P_1}	-	[t/h]	FTO_{T_3}	-	[t/h]
FP_{P_2,P_2}	-	[t/h]	FTP_{T_1,P_1}	-	[t/h]
FPO_{P_1}	-	[t/h]	FTP_{T_1,P_2}	-	[t/h]
FPO_{P_2}	-	[t/h]	FTP_{T_2,P_1}	40,00	[t/h]
FPT_{P_1,T_1}	-	[t/h]	FTP_{T_2,P_2}	-	[t/h]
FPT_{P_1,T_2}	-	[t/h]	FTP_{T_3,P_1}	-	[t/h]
FPT_{P_1,T_3}	-	[t/h]	FTP_{T_3,P_2}	10,00	[t/h]
FPT_{P_2,T_1}	-	[t/h]	$FTU_{T_1}^{in}$	-	[t/h]
FPT_{P_2,T_2}	6,23	[t/h]	$FTU_{T_2}^{in}$	40,00	[t/h]
FPT_{P_2,T_3}	43,77	[t/h]	$FTU_{T_3}^{in}$	43,77	[t/h]
$FPU_{P_1}^{out}$	40,00	[t/h]	$FTU_{T_1}^{out}$	-	[t/h]
$FPU_{P_2}^{out}$	50,00	[t/h]	$FTU_{T_2}^{out}$	40,00	[t/h]
FT_{T_1,T_1}	-	[t/h]	$FTU_{T_3}^{out}$	43,77	[t/h]
FT_{T_1,T_2}	-	[t/h]	FW	-	[t/h]

Tabla E.4: Concentraciones óptimas del ejemplo ilustrativo 3, empleando los valores nominales de los parámetros.

Variable	Valor	Unidad	Variable	Valor	Unidad
$C_{out,A}$	-	[ppm]	$CSTU_{T_1,C}^{out}$	10,00	[ppm]
$C_{out,B}$	-	[ppm]	$CSTU_{T_1,D}^{out}$	10,00	[ppm]
$C_{out,C}$	-	[ppm]	$CSTU_{T_2,A}^{out}$	-	[ppm]
$C_{out,D}$	-	[ppm]	$CSTU_{T_2,B}^{out}$	-	[ppm]
CSI_A	100,00	[ppm]	$CSTU_{T_2,C}^{out}$	-	[ppm]
CSI_B	100,00	[ppm]	$CSTU_{T_2,D}^{out}$	-	[ppm]
CSI_C	100,00	[ppm]	$CSTU_{T_3,A}^{out}$	2,02	[ppm]
CSI_D	100,00	[ppm]	$CSTU_{T_3,B}^{out}$	2,53	[ppm]
$CPU_{P_1,A}^{in}$	-	[ppm]	$CSTU_{T_3,C}^{out}$	2,02	[ppm]
$CPU_{P_1,B}^{in}$	-	[ppm]	$CSTU_{T_3,D}^{out}$	2,02	[ppm]
$CPU_{P_1,C}^{in}$	-	[ppm]	$CTU_{T_1,A}^{in}$	-	[ppm]
$CPU_{P_1,D}^{in}$	-	[ppm]	$CTU_{T_1,B}^{in}$	-	[ppm]
$CPU_{P_2,A}^{in}$	20,40	[ppm]	$CTU_{T_1,C}^{in}$	-	[ppm]
$CPU_{P_2,B}^{in}$	30,51	[ppm]	$CTU_{T_1,D}^{in}$	-	[ppm]
$CPU_{P_2,C}^{in}$	20,40	[ppm]	$CTU_{T_2,A}^{in}$	8,00	[ppm]
$CPU_{P_2,D}^{in}$	20,40	[ppm]	$CTU_{T_2,B}^{in}$	10,00	[ppm]
$CPU_{P_1,A}^{out}$	25,00	[ppm]	$CTU_{T_2,C}^{in}$	8,00	[ppm]
$CPU_{P_1,B}^{out}$	37,50	[ppm]	$CTU_{T_2,D}^{in}$	8,00	[ppm]
$CPU_{P_1,C}^{out}$	25,00	[ppm]	$CTU_{T_3,A}^{in}$	40,40	[ppm]
$CPU_{P_1,D}^{out}$	25,00	[ppm]	$CTU_{T_3,B}^{in}$	50,51	[ppm]
$CPU_{P_2,A}^{out}$	40,40	[ppm]	$CTU_{T_3,C}^{in}$	40,40	[ppm]
$CPU_{P_2,B}^{out}$	50,51	[ppm]	$CTU_{T_3,D}^{in}$	40,40	[ppm]
$CPU_{P_2,C}^{out}$	40,40	[ppm]	$CTU_{T_1,A}^{out}$	10,00	[ppm]
$CPU_{P_2,D}^{out}$	40,40	[ppm]	$CTU_{T_1,B}^{out}$	10,00	[ppm]
$CSPU_{P_1,A}^{out}$	25,00	[ppm]	$CTU_{T_1,C}^{out}$	10,00	[ppm]
$CSPU_{P_1,B}^{out}$	37,50	[ppm]	$CTU_{T_1,D}^{out}$	10,00	[ppm]
$CSPU_{P_1,C}^{out}$	25,00	[ppm]	$CTU_{T_2,A}^{out}$	-	[ppm]
$CSPU_{P_1,D}^{out}$	25,00	[ppm]	$CTU_{T_2,B}^{out}$	-	[ppm]
$CSPU_{P_2,A}^{out}$	40,40	[ppm]	$CTU_{T_2,C}^{out}$	-	[ppm]
$CSPU_{P_2,B}^{out}$	50,51	[ppm]	$CTU_{T_2,D}^{out}$	-	[ppm]
$CSPU_{P_2,C}^{out}$	40,40	[ppm]	$CTU_{T_3,A}^{out}$	2,02	[ppm]
$CSPU_{P_2,D}^{out}$	40,40	[ppm]	$CTU_{T_3,B}^{out}$	2,53	[ppm]
$CSTU_{T_1,A}^{out}$	10,00	[ppm]	$CTU_{T_3,C}^{out}$	2,02	[ppm]
$CSTU_{T_1,A}^{out}$	10,00	[ppm]	$CTU_{T_3,D}^{out}$	2,02	[ppm]

Tabla E.5: Instalaciones óptimas del ejemplo 3, empleando los valores nominales de los parámetros.

Variable	Valor	Unidad	Variable	Valor	Unidad
y_{FIF}	-	$[-]$	$y_{FT_{T_1, T_1}}$	-	$[-]$
$y_{FIP_{P_1}}$	-	$[-]$	$y_{FT_{T_1, T_2}}$	-	$[-]$
$y_{FIP_{P_2}}$	-	$[-]$	$y_{FT_{T_1, T_3}}$	-	$[-]$
$y_{FIT_{t_1}}$	-	$[-]$	$y_{FT_{T_2, T_1}}$	-	$[-]$
$y_{FIT_{t_2}}$	-	$[-]$	$y_{FT_{T_2, T_2}}$	-	$[-]$
$y_{FIT_{t_3}}$	-	$[-]$	$y_{FT_{T_2, T_3}}$	-	$[-]$
$y_{FP_{P_1, P_1}}$	-	$[-]$	$y_{FT_{T_3, T_1}}$	-	$[-]$
$y_{FP_{P_1, P_2}}$	1	$[-]$	$y_{FT_{T_3, T_2}}$	1	$[-]$
$y_{FP_{P_2, P_1}}$	-	$[-]$	$y_{FT_{T_3, T_3}}$	-	$[-]$
$y_{FP_{P_2, P_2}}$	-	$[-]$	$y_{FTO_{T_1}}$	-	$[-]$
$y_{FPO_{P_1}}$	-	$[-]$	$y_{FTO_{T_2}}$	-	$[-]$
$y_{FPO_{P_2}}$	-	$[-]$	$y_{FTO_{T_3}}$	-	$[-]$
$y_{FPT_{P_1, T_1}}$	-	$[-]$	$y_{FPT_{T_1, P_1}}$	-	$[-]$
$y_{FPT_{P_1, T_2}}$	-	$[-]$	$y_{FPT_{T_1, P_2}}$	-	$[-]$
$y_{FPT_{P_1, T_3}}$	-	$[-]$	$y_{FPT_{T_2, P_1}}$	1	$[-]$
$y_{FPT_{P_2, T_1}}$	-	$[-]$	$y_{FPT_{T_2, P_2}}$	-	$[-]$
$y_{FPT_{P_2, T_2}}$	1	$[-]$	$y_{FPT_{T_3, P_1}}$	-	$[-]$
$y_{FPT_{P_2, T_3}}$	1	$[-]$	$y_{FPT_{T_3, P_2}}$	1	$[-]$

Se han presentado en esta sección las concentraciones óptimas en caso de que se requiera ser consultadas, aunque para las comparaciones posteriores se enfatizará en los flujos y conexiones de la red, ya que corresponden a variables de mayor interés para el estudio realizado. Adicionalmente, se han incluido en las tablas los flujos de reciclos locales ($FP_{p,p}$ y $FT_{t,t}$) y sus respectivas decisiones de instalación ($y_{FP_{p,p}}$ e $y_{FT_{t,t}}$) para explicitar que sus valores son nulos en el caso estudiado. Si se desean incluir en nuevas resoluciones u otro tipo de análisis, se deben eliminar las siguientes condiciones que fueron incluidas particularmente para este ejemplo, en conjunto con la limitación de la conexión con el primer pretratamiento:

$$y_{FP_{p,p'}} = 0 \quad \forall p, p' \in PU \quad / \quad p = p' \quad (\text{E.74})$$

$$y_{FPT_{p,t}} = 0 \quad \forall p \in PU, \quad t = T_1 \in TU \quad (\text{E.75})$$

$$y_{FT_{t,t'}} = 0 \quad \forall t, t' \in PU \quad / \quad t = t' \quad (\text{E.76})$$

E.3. Rango de estabilidad

E.3.1. Parámetro $FPU_{P_1}^{in}$

La determinación del límite inferior del rango de estabilidad de $FPU_{P_1}^{in}$ se logró mediante las iteraciones que se encuentran en la Tabla E.6. Para cinco valores se alcanzó el tiempo máximo de ejecución (aprox. 1.000 [s]), presentándose una solución entera en tres de estos.

Tabla E.6: Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$.

i	$FPU_{P_1}^{in}$	$y' = y_0?$	$FPU_{P_1}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	40,0	Sí	—	—	1,74
1	30,0	Sí	—	—	1,09
2	20,0	Sí	—	—	2,31
3	10,0	No	20,0	$1,0 > 0,016$	17,51
Ciclo 2: $\Delta l = 1$					
0	20,0	Sí	—	—	2,20
1	19,0	Sí	—	—	2,08
2	18,0	Sí	—	—	1.019,52
3	17,0	Sí	—	—	1.030,71
4	16,0	Sí	—	—	3,19
5	15,0	Sí	—	—	2,85
6	14,0	Sí	—	—	4,27
7	13,0	Sí	—	—	4,24
8	12,0	Sí	—	—	4,33
9	11,0	No	12,0	$0,1 > 0,016$	17,34
Ciclo 3: $\Delta l = 0,1$					
0	12,0	Sí	—	—	4,39
1	11,9	Sí	—	—	1.025,57
2	11,8	Sí	—	—	5,46
3	11,7	Sí	—	—	1.018,65
4	11,6	Sí	—	—	7,36
5	11,5	Sí	—	—	6,53
6	11,4	No	11,5	$0,01 < 0,016$	1.014,49
Tiempo total: 5.195,83[s]					

Respecto al límite superior, las iteraciones involucradas se resumen en la Tabla E.7. Para $FPU_{P_1}^{in} = 49,0$ el algoritmo sugiere actualizar los parámetros de búsqueda. Debido a que la detención se produjo por esta situación, en la representación gráfica de la Figura E.1 se ha destacado que el límite estimado se debe al número de iteraciones máximo por paso. Notando que se evaluó $FPU_{P_1}^{in} = 50,0$ y la solución cambiaba, se puede concluir que el límite efectivo se encuentra entre 49,0 y 50,0, corroborado en la Figura E.1.

Tabla E.7: Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$.

i	$FPU_{P_1}^{in}$	$y' = y_0?$	$FPU_{P_1}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	40,0	Sí	—	—	1,51
1	50,0	No	40,0	$1,0 > 0,016$	1,44
Ciclo 2: $\Delta l = 1$					
0	40,0	Sí	—	—	1,57
1	41,0	Sí	—	—	1,20
2	42,0	Sí	—	—	1,18
3	43,0	Sí	—	—	1,35
4	44,0	Sí	—	—	1.022,65
5	45,0	Sí	—	—	1,68
6	46,0	Sí	—	—	1.027,16
7	47,0	Sí	—	—	1,10
8	48,0	Sí	—	—	1,16
9	49,0	Sí	—	—	1,19
Tiempo total: 2.063,19[s]					

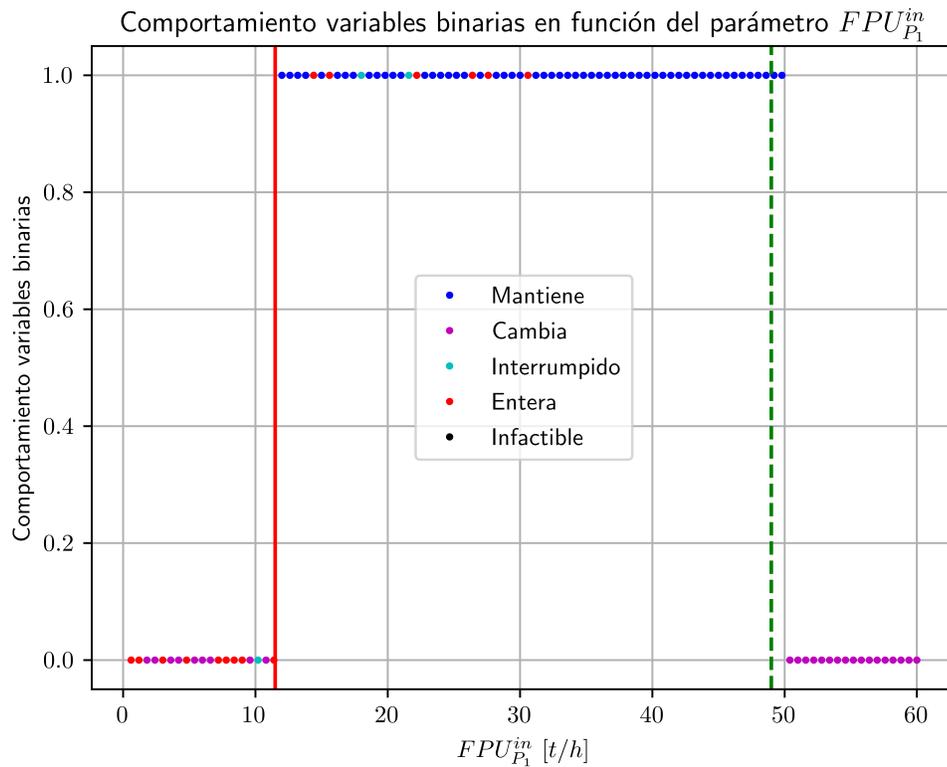


Figura E.1: Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_1}^{in}$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.

E.3.2. Parámetro $FPU_{P_2}^{in}$

Las iteraciones realizadas en el cálculo del límite inferior de $FPU_{P_2}^{in}$ se encuentran en la Tabla E.8. En este caso el algoritmo sugiere actualizar el paso de búsqueda si se requiere una solución más precisa, conociendo que el límite buscado se encuentra entre 40,0 y 41,0.

Tabla E.8: Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$.

i	$FPU_{P_2}^{in}$	$y' = y_0?$	$FPU_{P_2}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	50,0	Sí	—	—	1,64
1	40,0	No	50,0	$1,0 > 0,02$	1,22
Ciclo 2: $\Delta l = 1$					
0	50,0	Sí	—	—	1,97
1	49,0	Sí	—	—	1,48
2	48,0	Sí	—	—	1,49
3	47,0	Sí	—	—	1,42
4	46,0	Sí	—	—	1,27
5	45,0	Sí	—	—	1,41
6	44,0	Sí	—	—	1,29
7	43,0	Sí	—	—	1.032,53
8	42,0	Sí	—	—	693,69
9	41,0	Sí	—	—	1,13
Tiempo total: 1.740,54[s]					

Para la búsqueda del límite superior se ejecutaron las iteraciones de la Tabla E.9. El algoritmo se detiene tras alcanzar la tolerancia previamente definida, sin presentarse casos en que el tiempo máximo de ejecución sea alcanzado.

Realizando 100 evaluaciones para valores distintos de $FPU_{P_2}^{in}$ se obtienen los resultados de la Figura E.2, recordando que el valor 1 representa que las variables binarias mantienen su valor y 0 en el caso en que cambia alguna de estas. Se observa que para la determinación del límite inferior de $FPU_{P_2}^{in}$ se genera un efecto similar a lo acontecido en la búsqueda del límite superior de $FPU_{P_1}^{in}$: la detención se produjo por alcanzar el límite de iteraciones para un mismo paso ($i < 10$), existiendo la posibilidad de que se presenten valores inferiores que podrían formar parte del rango de estabilidad. Precisamente, en la Figura E.2 se aprecia aquella situación, donde el límite está entre 40,0 y 41,0, tal como se podía inferir de las evaluaciones efectuadas.

Tabla E.9: Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$.

i	$FPU_{P_2}^{in}$	$y' = y_0?$	$FPU_{P_2}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	50,0	Sí	—	—	1,79
1	60,0	No	50,0	$1,0 > 0,02$	2,46
Ciclo 2: $\Delta l = 1$					
0	50,0	Sí	—	—	1,70
1	51,0	Sí	—	—	1,42
2	52,0	Sí	—	—	1,39
3	53,0	Sí	—	—	3,91
4	54,0	Sí	—	—	3,82
5	55,0	Sí	—	—	3,77
6	56,0	No	55,0	$0,1 > 0,02$	3,46
Ciclo 3: $\Delta l = 0,1$					
0	55,0	Sí	—	—	4,04
1	55,1	Sí	—	—	2,89
2	55,2	Sí	—	—	4,12
3	55,3	Sí	—	—	4,19
4	55,4	No	55,3	$0,01 < 0,02$	4,44
Tiempo total: 43,40[s]					

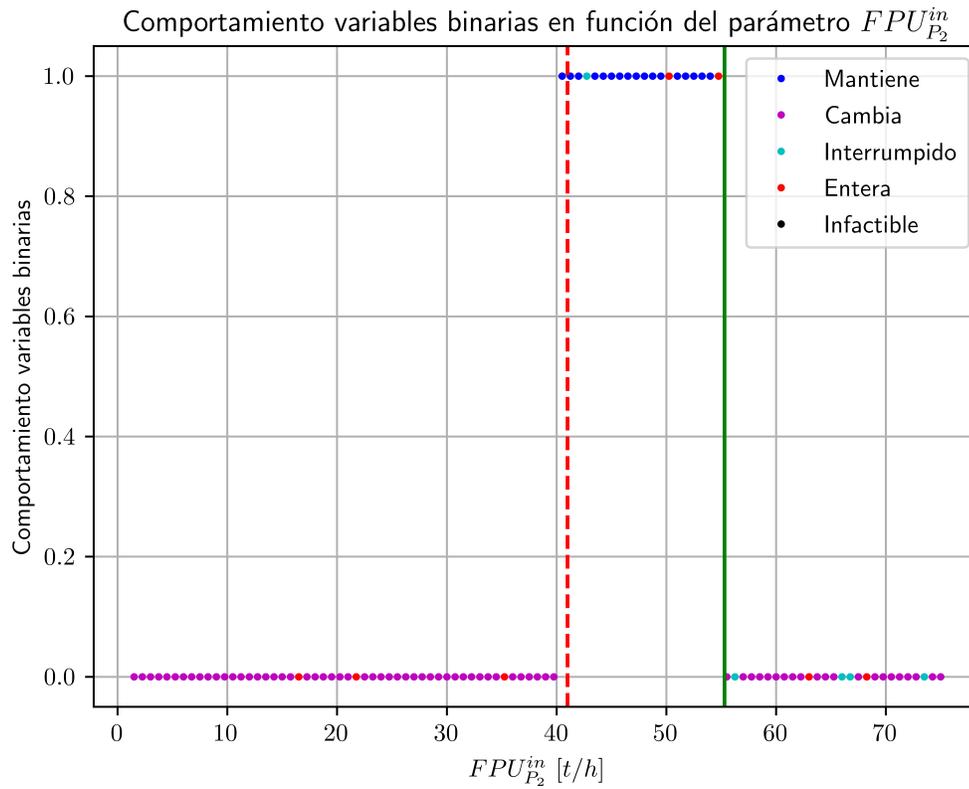


Figura E.2: Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.

E.3.3. Parámetro r_{T_3}

Para la búsqueda del límite inferior de r_{T_3} se requirieron las iteraciones de la Tabla E.10, finalizando tras alcanzar la tolerancia establecida. En estas evaluaciones no se presentaron casos que se aproximaran al tiempo máximo de ejecución (1.000[s]).

Tabla E.10: Determinación del límite inferior del rango de estabilidad del parámetro r_{T_3} .

i	r_{T_3}	$y' = y_0?$	r_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	95,0	Sí	–	–	1,50
1	85,0	Sí	–	–	1,25
2	75,0	No	85,0	$1,0 > 0,038$	2,31
Ciclo 2: $\Delta l = 1$					
0	85,0	Sí	–	–	1,08
1	84,0	Sí	–	–	1,14
2	83,0	Sí	–	–	0,78
3	82,0	Sí	–	–	1,08
4	81,0	Sí	–	–	0,97
5	80,0	No	81,0	$0,1 > 0,038$	91,57
Ciclo 3: $\Delta l = 0,1$					
0	81,0	Sí	–	–	0,98
1	80,9	Sí	–	–	0,99
2	80,8	Sí	–	–	1,06
3	80,7	No	80,8	$0,01 < 0,038$	19,71
Tiempo total: 124,42[s]					

El límite superior de r_{T_3} fue obtenido mediante las iteraciones de la Tabla E.11. Entre los valores evaluados, la estructura del algoritmo referencial considera la resolución del problema con $r_{T_3} = 105$. Para ese valor la solución es infactible, siendo un resultado esperado ya que se trata del porcentaje de remoción de contaminantes, que debiese estar acotado por un valor máximo de 100%. La detención de la búsqueda del límite se debe a que fue alcanzada la tolerancia establecida.

Las múltiples resoluciones para observar el comportamiento de las variables binarias se encuentran en la Figura E.3. Se analizaron valores entre 0 y 100, teniendo presente la información previa respecto a la interpretación del parámetro. Se destaca que no existen valores que ocasionen cambios en las variables binarias dentro del rango calculado, aunque existen algunos que provocan que la solución tarde más en obtenerse.

Tabla E.11: Determinación del límite superior del rango de estabilidad del parámetro r_{T_3} .

i	r_{T_3}	$y' = y_0?$	r_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	95,0	Sí	—	—	1,36
1	105,0	No	95,0	$1,0 > 0,038$	0,08
Ciclo 2: $\Delta l = 1$					
0	95,0	Sí	—	—	1,40
1	96,0	Sí	—	—	1,34
2	97,0	Sí	—	—	1,98
3	98,0	No	97,0	$0,1 > 0,038$	1,48
Ciclo 3: $\Delta l = 0,1$					
0	97,0	Sí	—	—	1,89
1	97,1	Sí	—	—	2,68
2	97,2	Sí	—	—	2,12
3	97,3	Sí	—	—	2,05
4	97,4	Sí	—	—	2,54
5	97,5	No	97,4	$0,01 < 0,038$	2,72
Tiempo total: 21,64[s]					

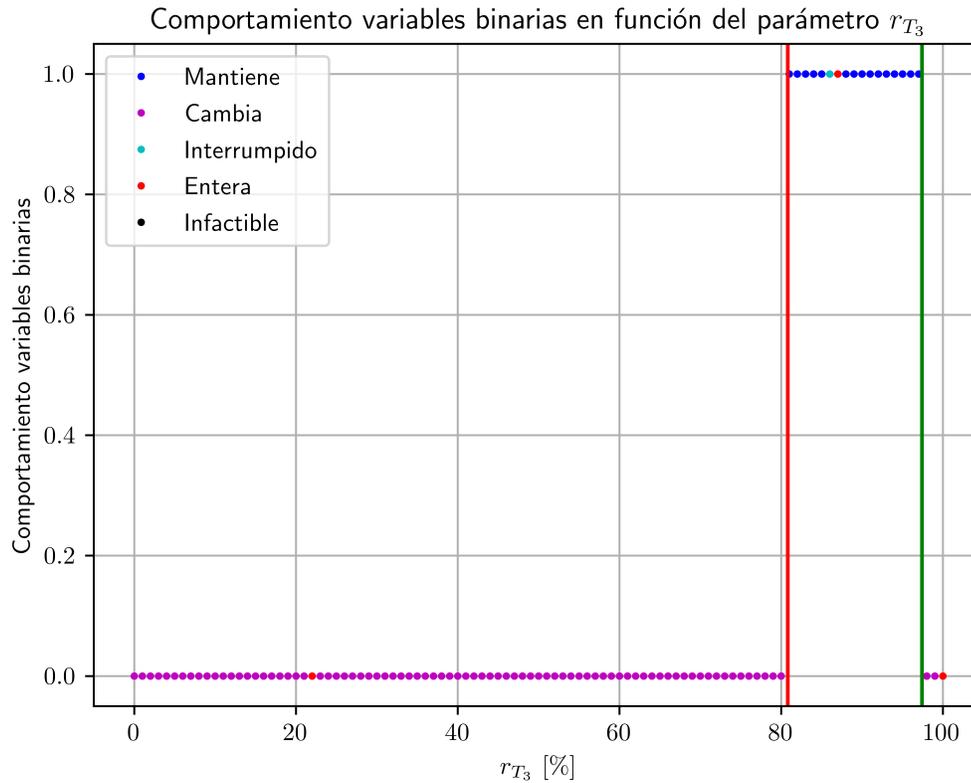


Figura E.3: Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} . Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

E.3.4. Parámetro IC_{T_1}

Para encontrar el límite inferior de IC_{T_1} se necesitaron las iteraciones que se observan en la Tabla E.12. Para el caso en que $IC_{T_1} = 4.600$ el tiempo de resolución máximo fue alcanzado. En cuanto al límite obtenido, este se conoce luego de que el algoritmo finalice la búsqueda por lograr la tolerancia establecida.

Tabla E.12: Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_1} .

i	IC_{T_1}	$y' = y_0?$	IC_{T_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	10.000	Sí	–	–	1,74
1	0	No	10.000	$1.000 > 4,0$	1,95
Ciclo 2: $\Delta l = 1.000$					
0	10.000	Sí	–	–	1,95
1	9.000	Sí	–	–	2,07
2	8.000	Sí	–	–	2,16
3	7.000	Sí	–	–	2,20
4	6.000	Sí	–	–	7,05
5	5.000	Sí	–	–	2,67
6	4.000	No	5.000	$100 > 4,0$	30,01
Ciclo 3: $\Delta l = 100$					
0	5.000	Sí	–	–	2,66
1	4.900	Sí	–	–	2,60
2	4.800	Sí	–	–	2,57
3	4.700	Sí	–	–	2,68
4	4.600	Sí	–	–	1021,64
5	4.500	Sí	–	–	3,72
6	4.400	Sí	–	–	3,73
7	4.300	Sí	–	–	3,25
8	4.200	Sí	–	–	3,58
9	4.100	No	4.200	$10 > 4,0$	3,90
Ciclo 4: $\Delta l = 10$					
0	4.200	Sí	–	–	3,74
1	4.190	Sí	–	–	2,81
2	4.180	Sí	–	–	3,88
3	4.170	Sí	–	–	3,35
4	4.160	Sí	–	–	4,20
5	4.150	Sí	–	–	4,12
6	4.140	No	4.150	$1,0 < 4,0$	5,91
Tiempo total: 1.131,07[s]					

Respecto al límite superior, fueron necesarias las iteraciones de la Tabla E.13 para determinar que no existe un cambio en las variables binarias, al menos, hasta el valor de $IC_{T_1} = 100.000$. Ante esta situación, el algoritmo sugiere actualizar los parámetros de búsqueda para continuar su ejecución. Para contrastar los resultados gráficamente empleando diferentes iteraciones, es posible analizar lo obtenido en la Figura E.4, donde se observa que no existen cambios en las variables binarias hacia valores superiores en el rango estudiado.

Tabla E.13: Determinación del límite superior del rango de estabilidad del parámetro IC_{T_1} .

i	IC_{T_1}	$y' = y_0?$	IC_{T_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	10.000	Sí	–	–	1,84
1	20.000	Sí	–	–	1,64
2	30.000	Sí	–	–	1,18
3	40.000	Sí	–	–	1,37
4	50.000	Sí	–	–	1,57
5	60.000	Sí	–	–	1,31
6	70.000	Sí	–	–	1,41
7	80.000	Sí	–	–	1,31
8	90.000	Sí	–	–	1.015,30
9	100.000	Sí	–	–	1.015,24
Tiempo total: 2.042,17[s]					

Se esperaría que la solución no cambiara en valores mayores a los representados en la Figura E.4, es decir, que el rango de estabilidad no estuviese acotado superiormente. Para corroborar lo anterior se podrían ejecutar más resoluciones para valores superiores, aunque teniendo presente que podrían existir casos en que el tiempo de resolución sea alto (orden de magnitud cercano al límite de tiempo por defecto de GAMS), ocasionando un alto tiempo computacional o interrupciones en las ejecuciones sin poder concluir sobre las variables binarias. Esta situación genera la siguiente reflexión: ¿hasta qué valor es necesario analizar que no existan cambios en las variables binarias?

Observando que la solución obtenida como límite superior es 1.000 % el valor nominal, sería conveniente considerar la interpretación que se realizaría con el rango de estabilidad. Particularmente, al estar estudiando las decisiones de instalación de plantas o tuberías en redes de agua, es importante que se contextualice el parámetro estudiado y analizar si podría ocurrir que el costo de inversión aumente a un valor mayor a 10 veces el nominal. En caso de que se estime que no ocurrirán cambios de esa magnitud, no sería necesario determinar efectivamente qué ocurre en valores superiores al determinado.

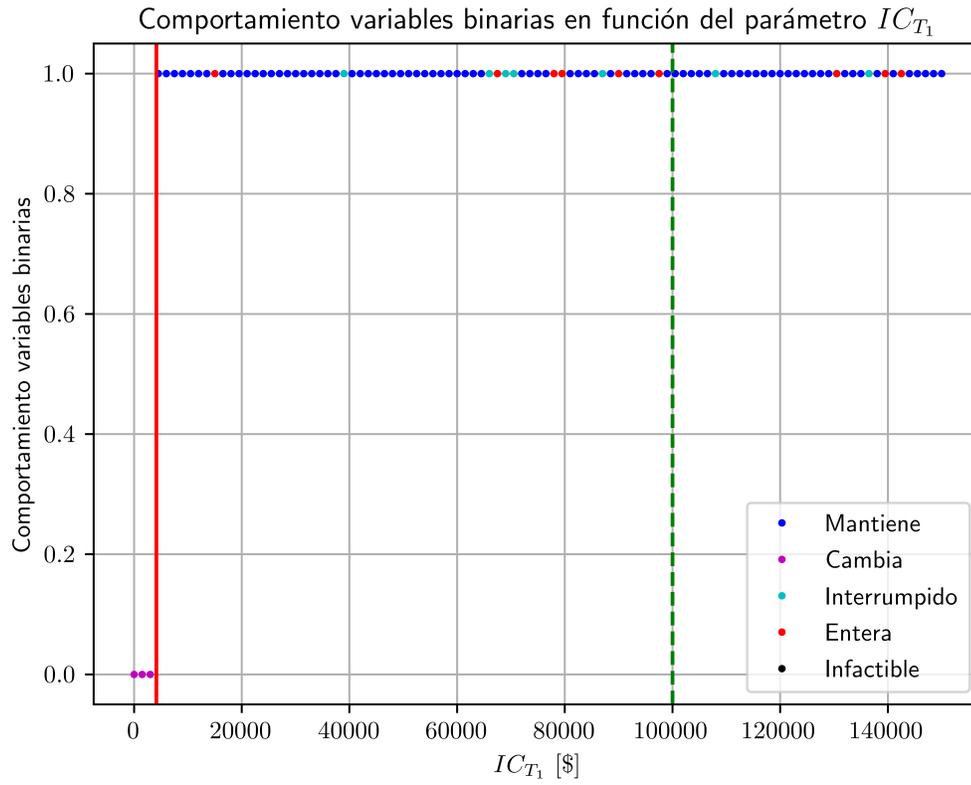


Figura E.4: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} . Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.

E.3.5. Parámetro IC_{T_2}

La determinación del límite inferior del rango de estabilidad de IC_{T_2} se efectuó mediante las iteraciones de la Tabla E.14. Se observa que, siguiendo los pasos del algoritmo referencial, se evalúan valores inferiores a cero. Comprendiendo la naturaleza del parámetro escogido, se considera que el límite debiese encontrarse en el valor 0 y haber evaluado valores negativos se emplea como antecedente para eventuales mejoras, proyectando el nuevo algoritmo a desarrollar. Se señala también que al evaluar $IC_{T_2} = -25.000$ se alcanza el tiempo máximo de resolución, razón por la que se eleva el tiempo empleado. Finalmente, el algoritmo termina la búsqueda sugiriendo asignar un nuevo paso y tolerancia en caso de requerir continuar su implementación.

Tabla E.14: Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_2} .

i	IC_{T_2}	$y' = y_0?$	IC_{T_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	25.000	Sí	—	—	1,42
1	15.000	Sí	—	—	1,22
2	5.000	Sí	—	—	1,15
3	-5.000	Sí	—	—	1,06
4	-15.000	Sí	—	—	1,17
5	-25.000	Sí	—	—	1.015,31
6	-35.000	Sí	—	—	1,27
7	-45.000	Sí	—	—	1,14
8	-55.000	Sí	—	—	1,23
9	-65.000	Sí	—	—	1,68
Tiempo total: 1.026,65[s]					

La búsqueda del límite superior del rango de estabilidad de IC_{T_2} se efectuó con las iteraciones de la Tabla E.15. Tal como se observa, no existió cambio en las variables binarias para los valores analizados, deteniéndose la búsqueda y sugiriendo actualizar el paso y la tolerancia del algoritmo, tal como para la determinación del límite inferior. Gráficamente, los valores mínimos y máximos evaluados se representan por líneas discontinuas en la Figura E.5, donde también se observa que no se registra un cambio en las variables binarias dentro del rango estudiado, aunque aparecen valores que provocan que se alcance el tiempo límite de resolución.

Tabla E.15: Determinación del límite superior del rango de estabilidad del parámetro IC_{T_2} .

i	IC_{T_2}	$y' = y_0?$	IC_{T_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	25.000	Sí	—	—	1,45
1	35.000	Sí	—	—	1,59
2	45.000	Sí	—	—	1,44
3	55.000	Sí	—	—	0,76
4	65.000	Sí	—	—	1,25
5	75.000	Sí	—	—	1,12
6	85.000	Sí	—	—	1,20
7	95.000	Sí	—	—	1,09
8	105.000	Sí	—	—	1,32
9	115.000	Sí	—	—	1,54
Tiempo total: 12,76[s]					

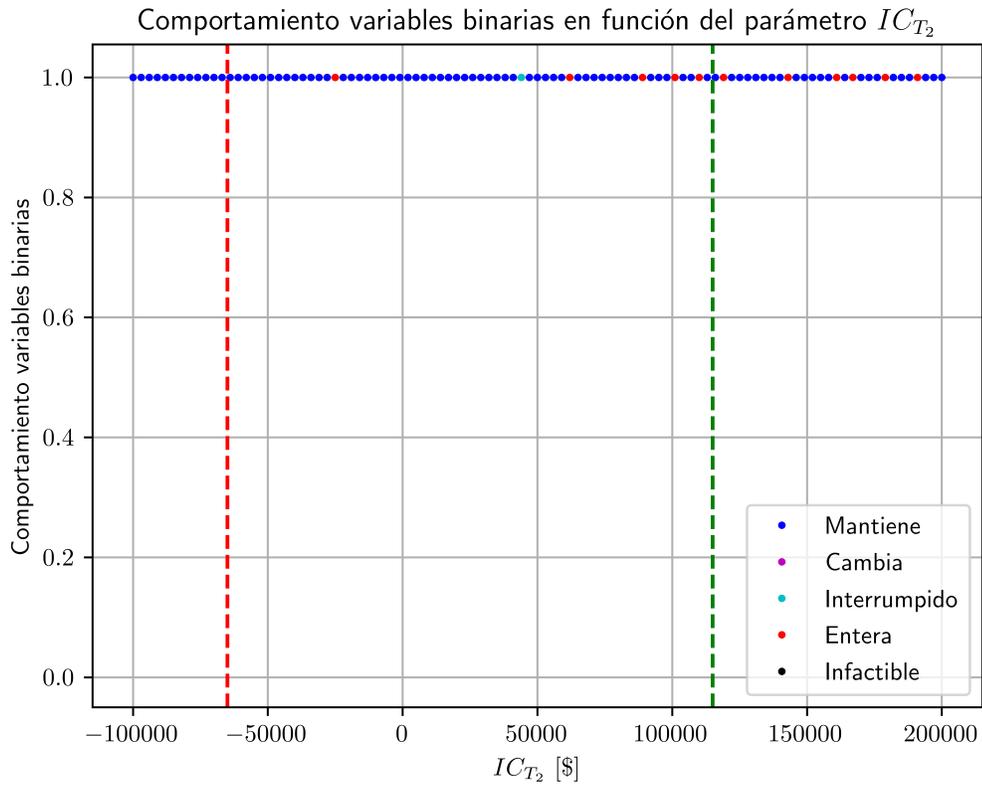


Figura E.5: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_2} . Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.

E.3.6. Parámetro IC_{T_3}

Las iteraciones observadas en la Tabla E.16 permitieron determinar el límite inferior del rango de estabilidad de IC_{T_3} . Al igual que con IC_{T_2} , se destaca que el algoritmo continúa evaluando valores menores a cero al no haber un cambio en las variables binarias. Nuevamente, por la naturaleza e interpretación del parámetro, el límite debiese ser cero. Particularmente, la detención de la búsqueda del límite inferior se produce luego de que el algoritmo sugiera actualizar el paso y la tolerancia, tras no detectar cambios en las variables binarias.

Tabla E.16: Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_3} .

i	IC_{T_3}	$y' = y_0?$	IC_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	30.000	Sí	—	—	1,56
1	20.000	Sí	—	—	1,41
2	10.000	Sí	—	—	1,23
3	0	Sí	—	—	1,09
4	-10,000	Sí	—	—	1,59
5	-20,000	Sí	—	—	1,10
6	-30,000	Sí	—	—	1,34
7	-40,000	Sí	—	—	0,79
8	-50,000	Sí	—	—	1,30
9	-60,000	Sí	—	—	1,19
Tiempo total: 12,60[s]					

En cuanto al límite superior, las iteraciones efectuadas para calcularlo se encuentran en la Tabla E.17. Para los valores evaluados no se detectaron cambios en las variables binarias, razón por la que el algoritmo sugiere actualizar el paso y la tolerancia de búsqueda.

En la Figura E.6 se observan 100 evaluaciones realizadas para distintos valores de IC_{T_3} , notando que no hay cambios en las variables binarias dentro del rango determinado, tal como se esperaba. Sin embargo, existen valores en que el tiempo de resolución máximo es alcanzado. Adicionalmente, se aprecia que para valores superiores al rango determinado se produce una modificación en los valores de las variables binarias, lo que se podría detectar en caso de que se hubiera considerado necesario continuar la búsqueda del límite del rango de estabilidad.

Tabla E.17: Determinación del límite superior del rango de estabilidad del parámetro IC_{T_3} .

i	IC_{T_3}	$y' = y_0?$	IC_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	30.000	Sí	—	—	1,56
1	40.000	Sí	—	—	1,19
2	50.000	Sí	—	—	1,31
3	60.000	Sí	—	—	1,31
4	70.000	Sí	—	—	1,64
5	80.000	Sí	—	—	1,45
6	90.000	Sí	—	—	1,71
7	100.000	Sí	—	—	2,63
8	110.000	Sí	—	—	3,68
9	120.000	Sí	—	—	2,94
Tiempo total: 19,42[s]					

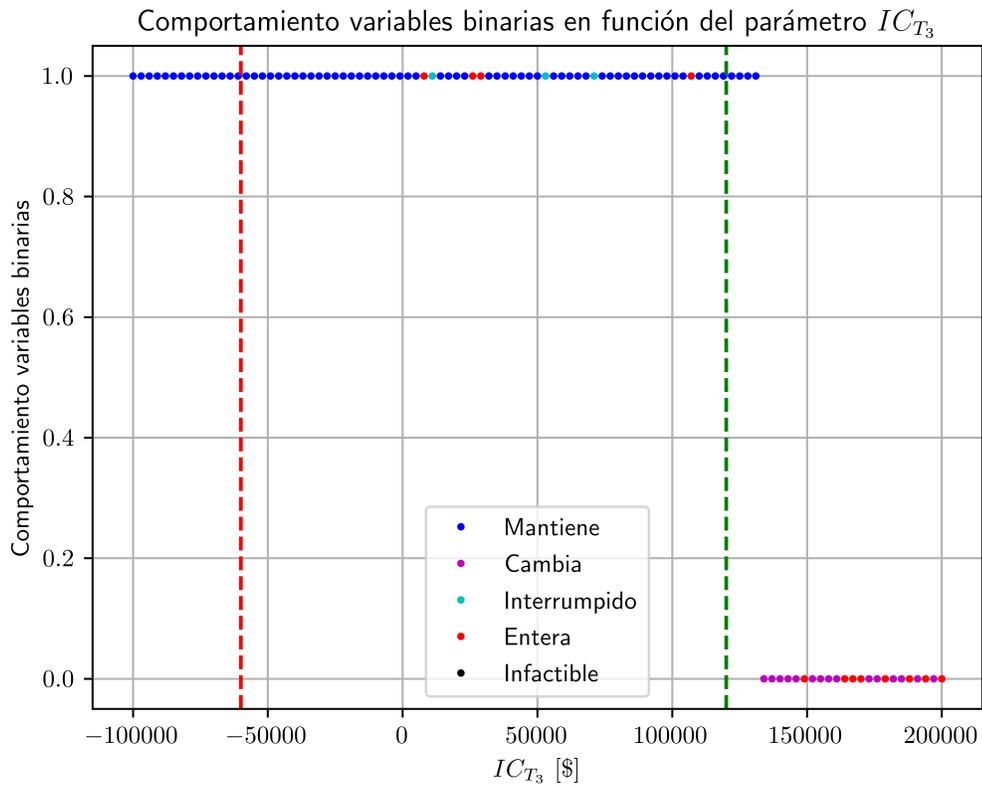


Figura E.6: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_3} . Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.

E.4. Efecto del tiempo de resolución

El tiempo requerido para la ejecución del problema original (con valores nominales de los parámetros) fue de 1,71[s]. A continuación, se analizará cuánto tiempo se necesitó para diferentes resoluciones de los parámetros en estudio, representándose gráficamente en relación al tiempo nominal, según lo planteado en la sección 5.2.1.4.

E.4.1. Parámetro $FPU_{P_1}^{in}$

El efecto que producen las variaciones del parámetro $FPU_{P_1}^{in}$ en la solución ha sido previamente representado en la Figura E.1. El tiempo requerido en cada una de las correspondientes resoluciones ha sido clasificado y representado en la Figura E.7, observándose que alrededor de la mitad de los casos necesitaron un tiempo inferior al nominal para ejecutarse. Sin embargo, y a diferencia de los parámetros de los ejemplos 1 y 2, se observa que aproximadamente un 21 % de las evaluaciones requirieron un tiempo superior a 5 veces el nominal. Específicamente, un 19 % necesitó un período superior a 100 veces aquel valor.

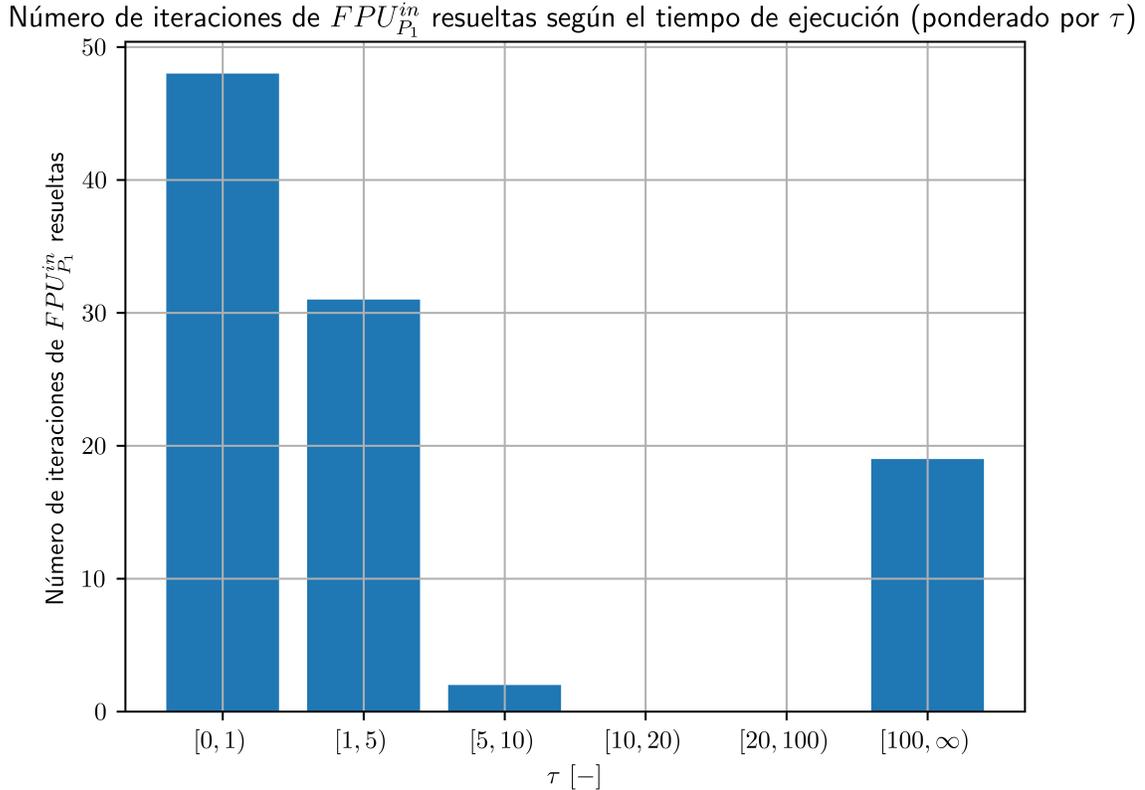


Figura E.7: Número de iteraciones de $FPU_{P_1}^{in}$ resueltas según el tiempo de ejecución.

Los ponderadores anteriores (τ) se utilizan para resolver nuevamente las iteraciones previamente analizadas, detallándose a continuación el efecto que producen en la solución estas limitaciones de tiempo:

- $\tau = 5, 10, 20, 100$: Se obtuvieron soluciones con el mismo comportamiento que lo observado en la Figura E.1, es decir, el número de iteraciones clasificadas como

Interrumpido y *Entera* se mantiene, tal como se ejemplifica en la Figura E.8.

- $\tau = 1$: El número de iteraciones clasificadas como *Entera* aumenta en contraste con lo obtenido en la Figura E.8 y en el caso en que el límite de tiempo es el incluido por defecto en GAMS, según se aprecia en la Figura E.9. Particularmente, se observa que la mayoría de las soluciones binarias se mantienen pese a que se interrumpió la búsqueda por el tiempo acotado, excepto para un valor: $FPU_{P_1}^{in} = 11,4$.

Los casos anteriores permiten considerar que existe un tiempo que ocasiona que la solución varíe respecto a lo obtenido con un mayor tiempo de ejecución. Esto se genera si se acota el tiempo máximo con el requerido en el caso original (sin variar parámetros, es decir, con valores nominales), aunque si se utiliza una holgura que permita resolver cada caso con un tiempo de 5 veces el requerido originalmente, se obtienen las mismas soluciones para las variables binarias.

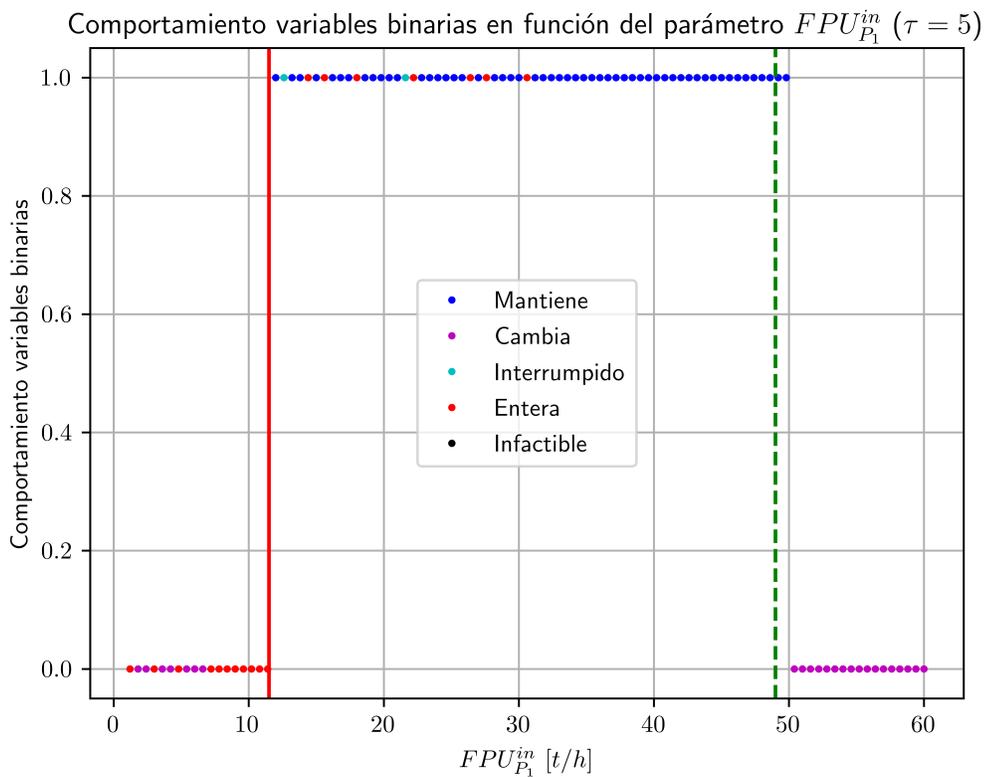


Figura E.8: Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_1}^{in}$ con $\tau = 5$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.

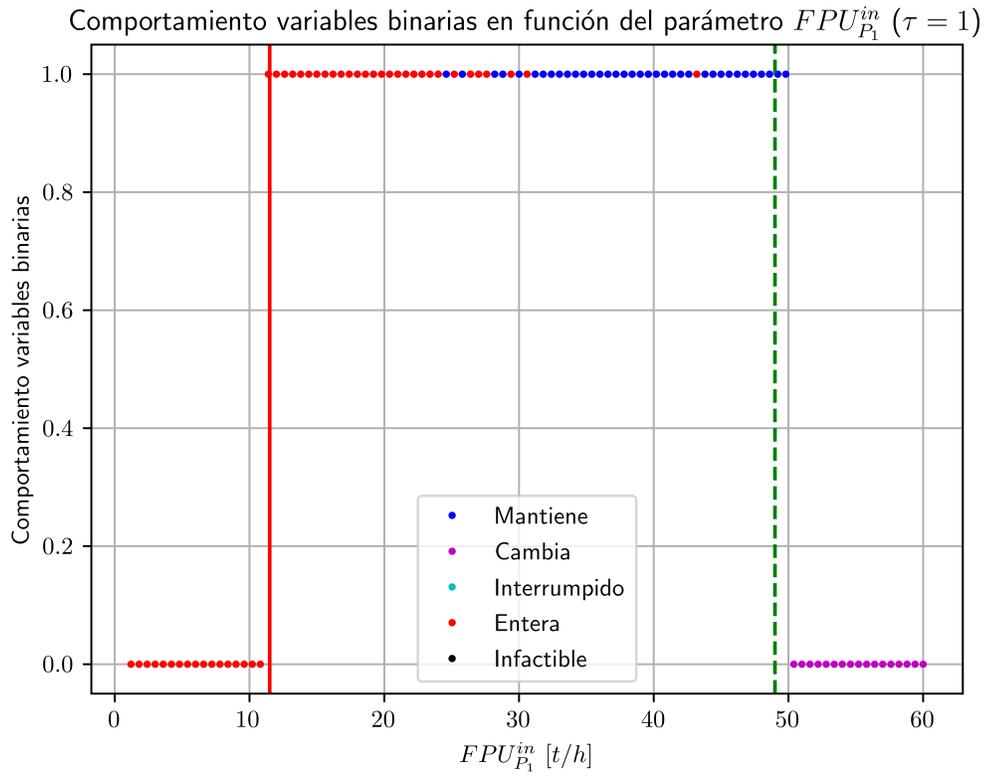


Figura E.9: Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_1}^{in}$ con $\tau = 1$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.

E.4.2. Parámetro $FPU_{P_2}^{in}$

Al resolver múltiples veces el problema modificando $FPU_{P_2}^{in}$, se tiene el comportamiento analizado y presentado en la Figura E.2. El tiempo empleado para estas resoluciones presenta una tendencia similar a lo ocurrido para $FPU_{P_1}^{in}$, según lo expresado en la Figura E.10: Alrededor de la mitad de los casos necesita un tiempo inferior al nominal, aunque se presentan evaluaciones que requieren tiempos superiores a 5 veces el nominal, existiendo también casos en que se necesita una ponderación superior a 100 veces este tiempo, debido a la complejidad de las expresiones involucradas en el ejemplo.

Número de iteraciones de $FPU_{P_2}^{in}$ resueltas según el tiempo de ejecución (ponderado por τ)

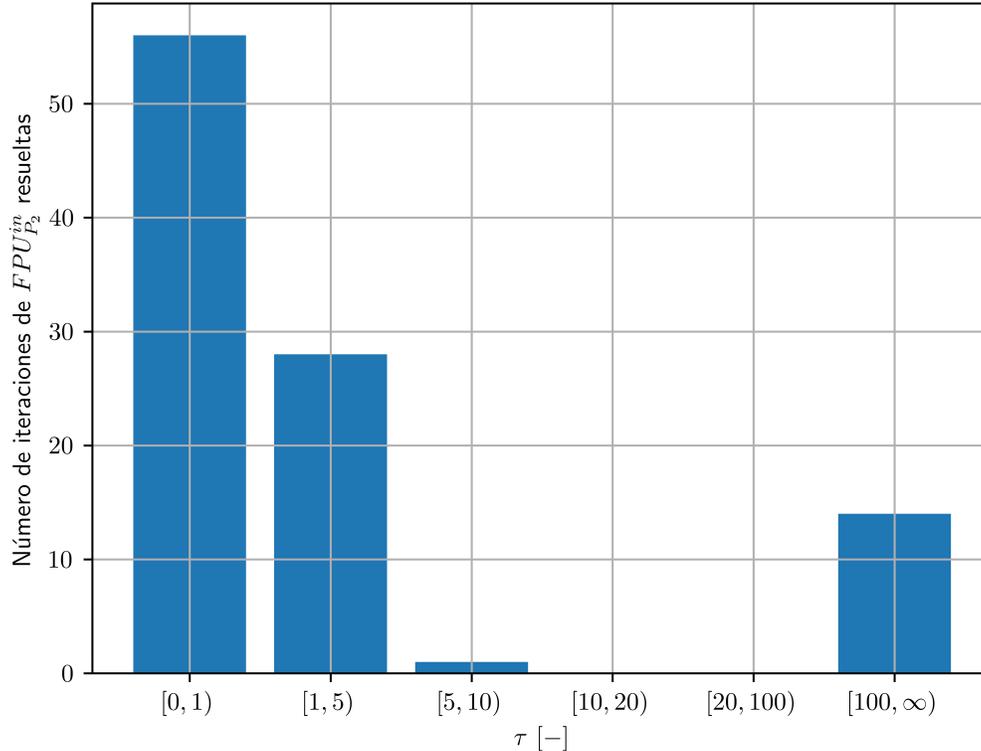


Figura E.10: Número de iteraciones de $FPU_{P_2}^{in}$ resueltas según el tiempo de ejecución.

Existiendo más del 80 % de los casos requiriendo $\tau \leq 10$, es necesario analizar la utilidad de un ponderador igual a 100 o mayor para poder concluir sin interrupciones las iteraciones del presente ejemplo. Si se emplean varias iteraciones en el algoritmo de búsqueda del rango de estabilidad, aumentar en 100 veces el tiempo por iteración podría significar un alto costo en términos de tiempo para el análisis completo. El caso contrario, en que se limita la ejecución para diferentes valores de τ , se analiza a continuación:

- $\tau = 20, 100$: Los valores obtenidos en las variables binarias son iguales a los registrados en la Figura E.2. La única diferencia radica en la aparición de un valor adicional catalogado como solución *Entera* y otro como *Interrumpido*, que anteriormente fueron resueltos sin interrupción, según lo reflejado en la Figura E.11.
- $\tau = 5, 10$: En estos casos también se mantienen los valores de las variables binarias. Sin embargo, se producen dos cambios adicionales a lo mencionado para $\tau = 20$ y $\tau = 100$,

ya que un punto *Interrumpido* se cambia a *Entera* y otro óptimo (*Cambia*) se reemplaza por *Interrumpido*, según se observa en la Figura E.12.

- $\tau = 1$: Como se esperaba, el número de iteraciones clasificadas como *Entera* aumenta en contraste con lo obtenido en los casos anteriores, según se resume en la Figura E.13. Es posible detectar que las variables binarias cambian respecto a los escenarios anteriores y también se aprecia un menor número de puntos ya que en algunos casos no fue posible obtener una solución en el período de tiempo establecido.

Los escenarios anteriores tienen similitudes y diferencias con lo ocurrido para $FPU_{P_1}^{in}$. Tal como ocurrió con aquel parámetro, para los valores de τ mayores a 5 es posible obtener la misma solución que la adquirida mediante el límite de tiempo por defecto de GAMS (1.000[s]). Sin embargo, se diferencian $FPU_{P_1}^{in}$ y $FPU_{P_2}^{in}$ para el caso en que $\tau = 1$, debido a que para $FPU_{P_2}^{in}$ se presenta un mayor número de valores en que las variables binarias no concuerdan con lo obtenido al emplear una mayor holgura de tiempo de resolución y también por la existencia de soluciones que no es posible determinar por el solver durante el acotado tiempo de resolución.

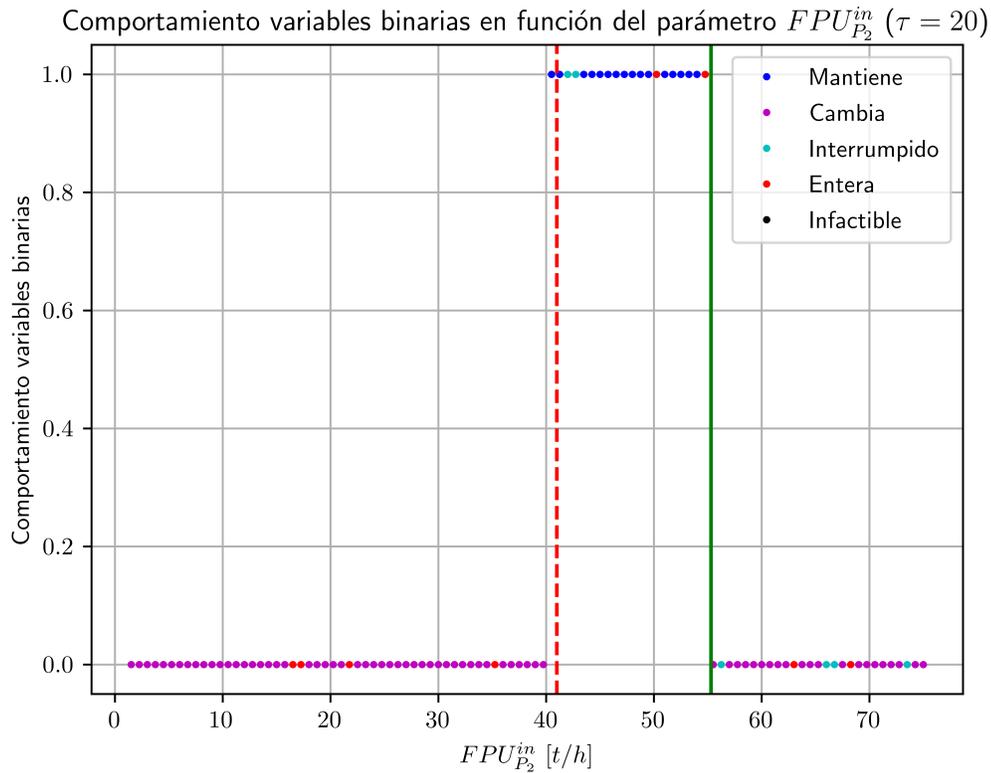


Figura E.11: Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$ con $\tau = 20$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.

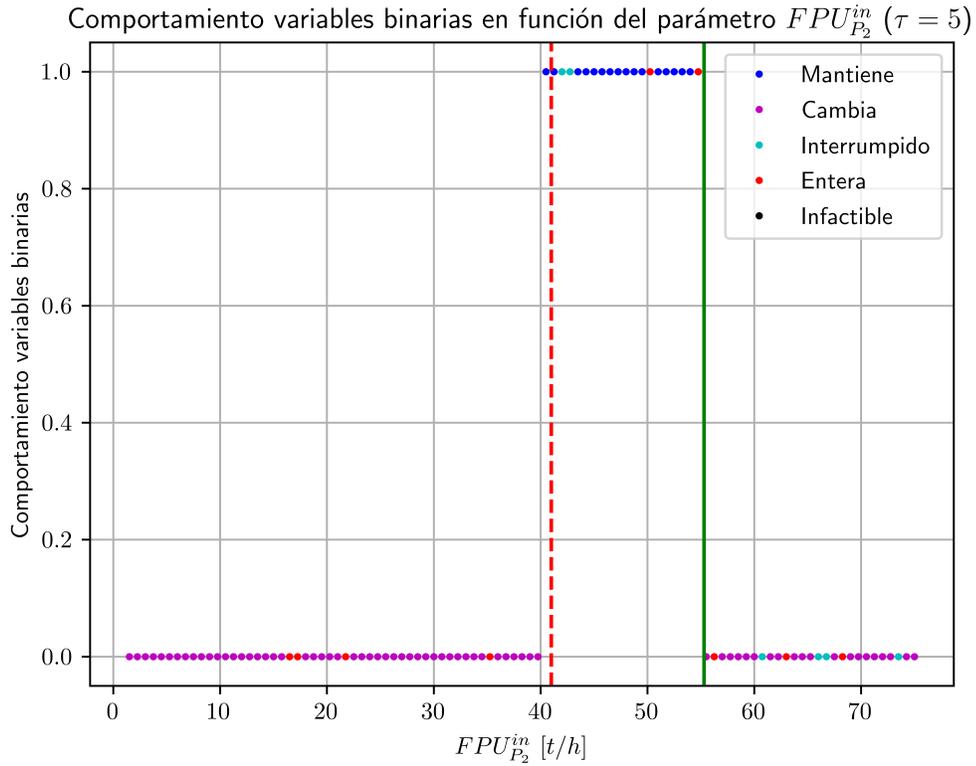


Figura E.12: Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$ con $\tau = 5$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.

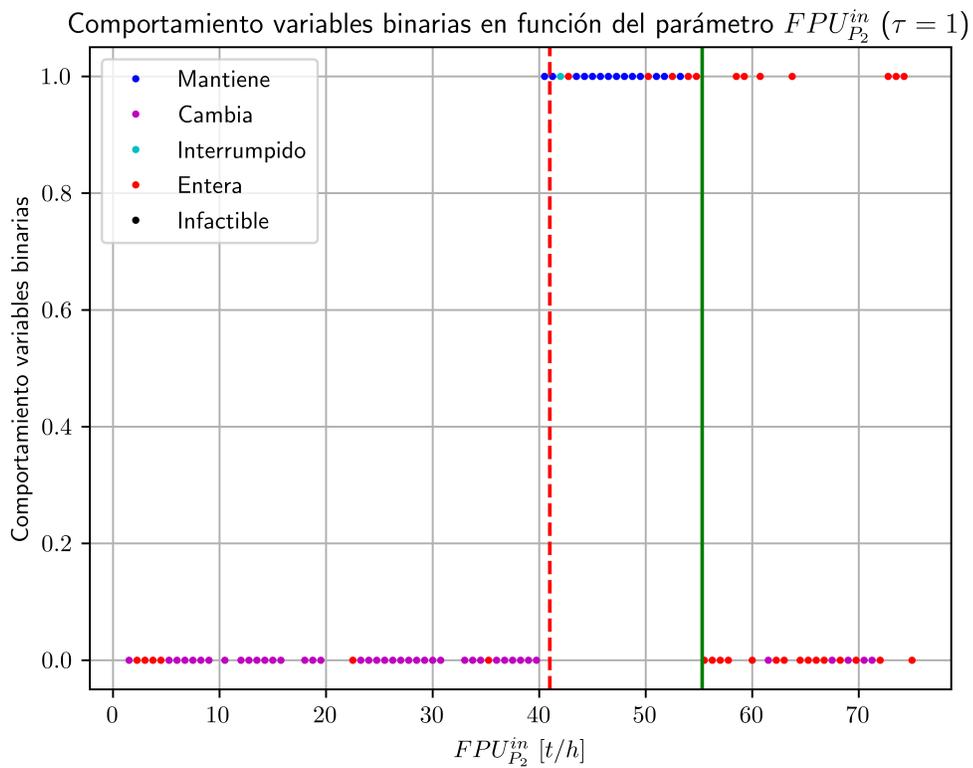


Figura E.13: Comportamiento de las variables binarias frente a cambios en el parámetro $FPU_{P_2}^{in}$ con $\tau = 1$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior calculado.

E.4.3. Parámetro r_{T_3}

El comportamiento de las variables binarias frente a modificaciones en r_{T_3} ha sido presentado en la Figura E.3, mientras que el tiempo requerido para cada una de estas se ha agrupado en la Figura E.14. Para este parámetro se observa que cerca del 80 % necesita un tiempo inferior al nominal. Existen también casos que requieren una ponderación del tiempo superior a 5 veces, tal como para las variaciones en los flujos $FPU_{P_1}^{in}$ y $FPU_{P_2}^{in}$, aunque en un menor porcentaje que para aquellos parámetros. Una de las diferencias entre r_{T_3} y los flujos mencionados radica en el número y complejidad de expresiones en que está involucrado el porcentaje de remoción en la formulación del problema, considerando que los flujos incluso afectan a las cotas adicionales planteadas por Ahmetovic y Grossmann [73].

Número de iteraciones de r_{T_3} resueltas según el tiempo de ejecución (ponderado por τ)

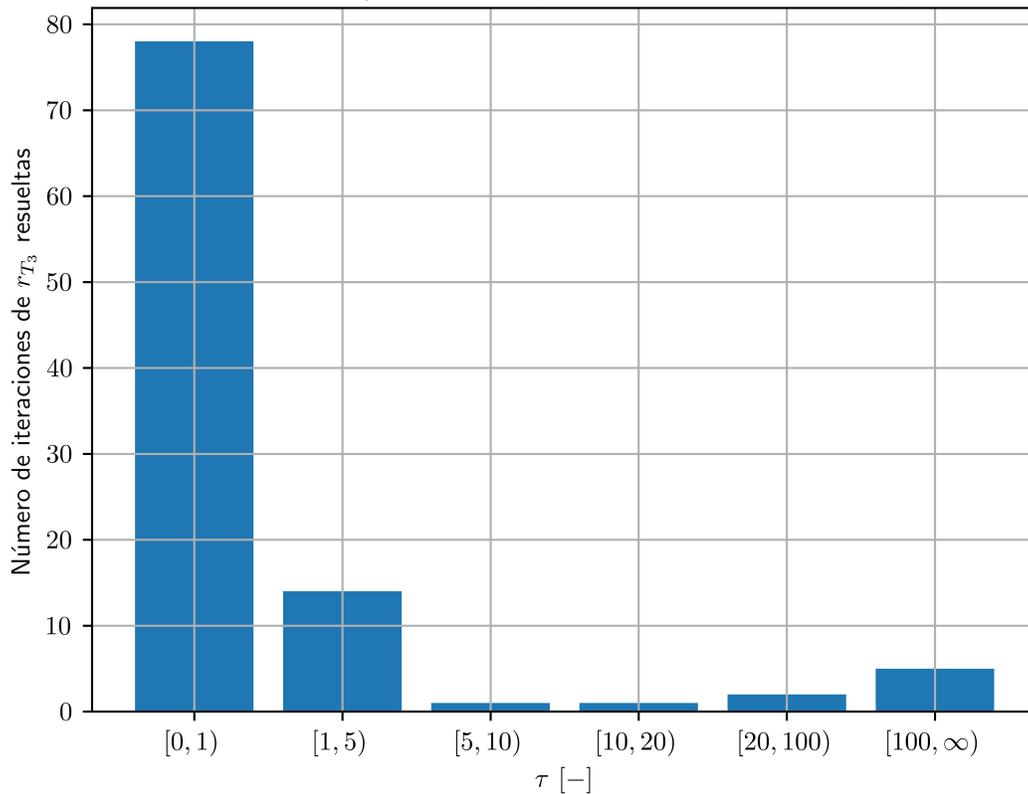


Figura E.14: Número de iteraciones de r_{T_3} resueltas según el tiempo de ejecución.

Considerando el alto porcentaje de escenarios que es posible resolver con un tiempo igual o inferior al nominal, se analiza a continuación el efecto que tendría la modificación del tiempo de resolución:

- $\tau = 100$: Los valores obtenidos en las variables binarias son iguales a los registrados en la Figura E.2. Solamente difieren en la aparición de un nuevo valor en que la solución óptima no fue alcanzada y se etiqueta como *Entera*, según lo observado en la Figura E.15.
- $\tau = 5, 10, 20$: Para estos ponderadores se mantienen los valores determinados originalmente en las variables binarias, aunque para $\tau = 20$ aumenta en dos casos

el número de puntos clasificados como *Entera*, apareciendo uno más con $\tau = 10$ y se adiciona otro con $\tau = 5$. Ilustrativamente, se señala en la Figura E.16 el escenario en que se alcanza el mayor número de puntos catalogados como solución *Entera* en contraste a lo indicado para $\tau = 100$.

- $\tau = 1$: Como se esperaba, el número de iteraciones en que la solución se clasifica como *Entera* aumenta en comparación con lo obtenido en los casos anteriores, según se observa en la Figura E.17. No se presentan cambios en las variables binarias, aunque existieron puntos en que no fue posible obtener una solución en el tiempo delimitado.

Tal como se esperaba por lo obtenido en la Figura E.14, la limitación del tiempo de resolución generó un menor número de casos en que se produjera una interrupción en la búsqueda de la solución óptima, en contraste con los flujos $FPU_{P_1}^{in}$ y $FPU_{P_2}^{in}$. De todas formas, se observa que la solución se mantiene para valores de τ mayores o iguales a 5. Particularmente, también para $\tau = 1$, aunque existen valores en que no se obtuvo solución, por lo que no es posible asegurar qué valores hubiesen adoptado.

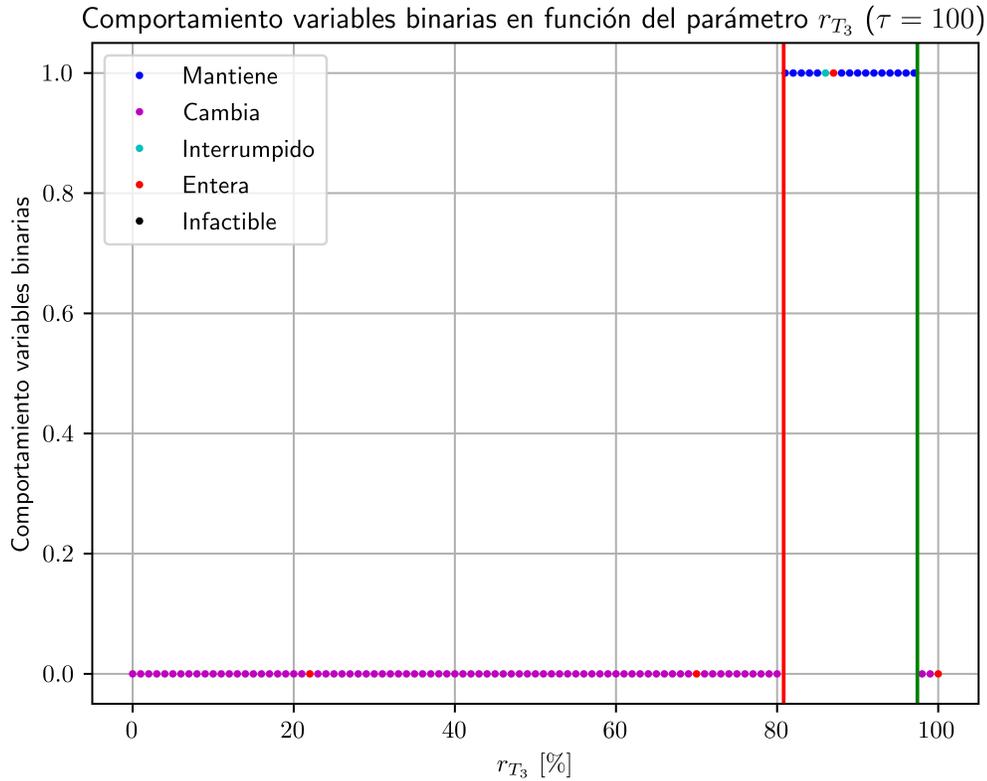


Figura E.15: Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} con $\tau = 100$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

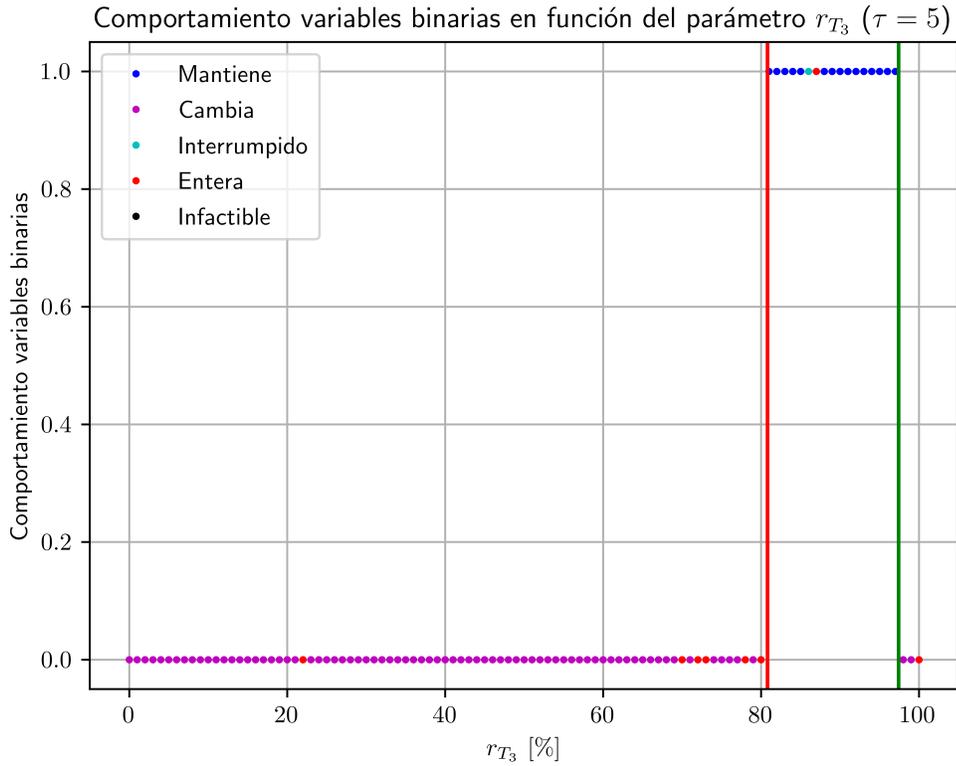


Figura E.16: Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} con $\tau = 5$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

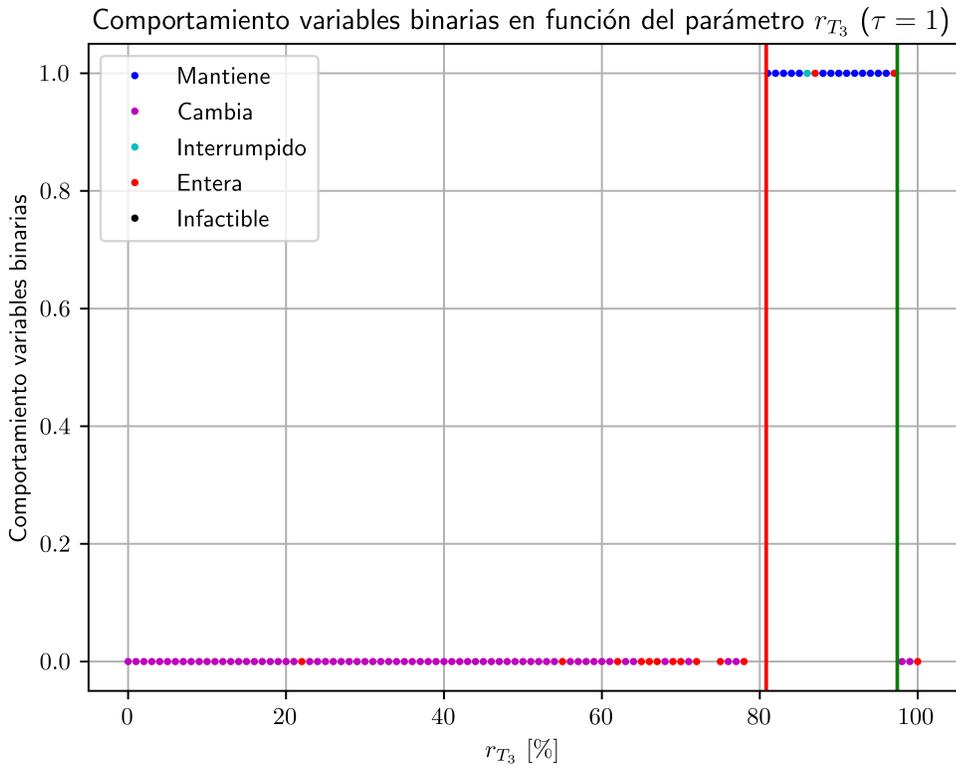


Figura E.17: Comportamiento de las variables binarias frente a cambios en el parámetro r_{T_3} con $\tau = 1$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado.

E.4.4. Parámetro IC_{T_1}

Las distintas evaluaciones efectuadas modificando IC_{T_1} se resumen en la Figura E.4, mientras que el tiempo requerido para aquellos casos se agrupa y representa en la Figura E.18. En este gráfico se aprecia que alrededor del 70% de las ejecuciones necesitan un tiempo inferior al nominal, ascendiendo hasta el 80% aproximadamente los casos que se pueden resolver con un tiempo igual a 5 veces el nominal.

Número de iteraciones de IC_{T_1} resueltas según el tiempo de ejecución (ponderado por τ)

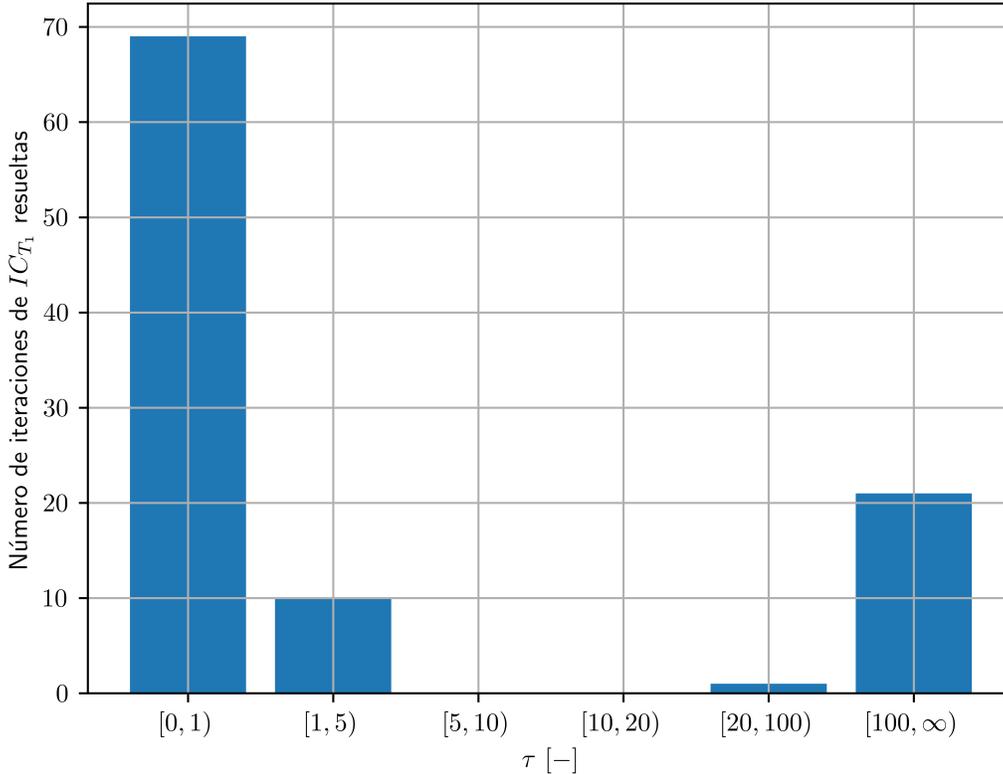


Figura E.18: Número de iteraciones de IC_{T_1} resueltas según el tiempo de ejecución.

Para las modificaciones en IC_{T_1} , es posible añadir una iteración más sin interrupciones si aumenta el tiempo de resolución entre 20 y 100 veces el original. Para agregar el 21% faltante es necesario un tiempo superior a 100 veces el nominal, por lo que se requiere evaluar si ese aumento es justificado o si se pueden aplicar otras alternativas para disminuir el tiempo necesario. El efecto de acotar el tiempo de resolución se estudia a partir de la resolución de los casos mencionados con diferentes tiempos máximos de ejecución, obteniéndose lo siguiente:

- $\tau = 10, 20, 100$: Los valores determinados en las variables binarias son iguales a los señalados en la Figura E.4. El número de soluciones con la denominación *Entera* también se mantiene, aunque aparecieron puntos óptimos con la característica de *Interrumpido*, tal como se observa en la Figura E.19.
- $\tau = 5$: Para este ponderador se mantiene la solución original de las variables binarias, aunque difiere de lo obtenido anteriormente (con $\tau \geq 10$) por la aparición de un nuevo valor cuya solución es *Entera*, que originalmente era *Interrumpido*, según lo reflejado en la Figura E.20.

- $\tau = 1$: Como se esperaba, el número de iteraciones en que la solución se clasifica como *Entera* aumenta en comparación con lo determinado en los casos anteriores, según se observa en la Figura E.21. No se presentan cambios en los valores de las variables binarias.

A diferencia de lo obtenido con los parámetros anteriores, para IC_{T_1} todos los valores de τ permitieron obtener valores iguales en las variables binarias. Entonces, para este parámetro se podría haber acotado el tiempo empleando $\tau = 1$, aunque el aumento en el número de soluciones enteras podría generar incertidumbre sobre la solución, siendo un efecto que se produce en menor medida con $\tau \geq 5$.

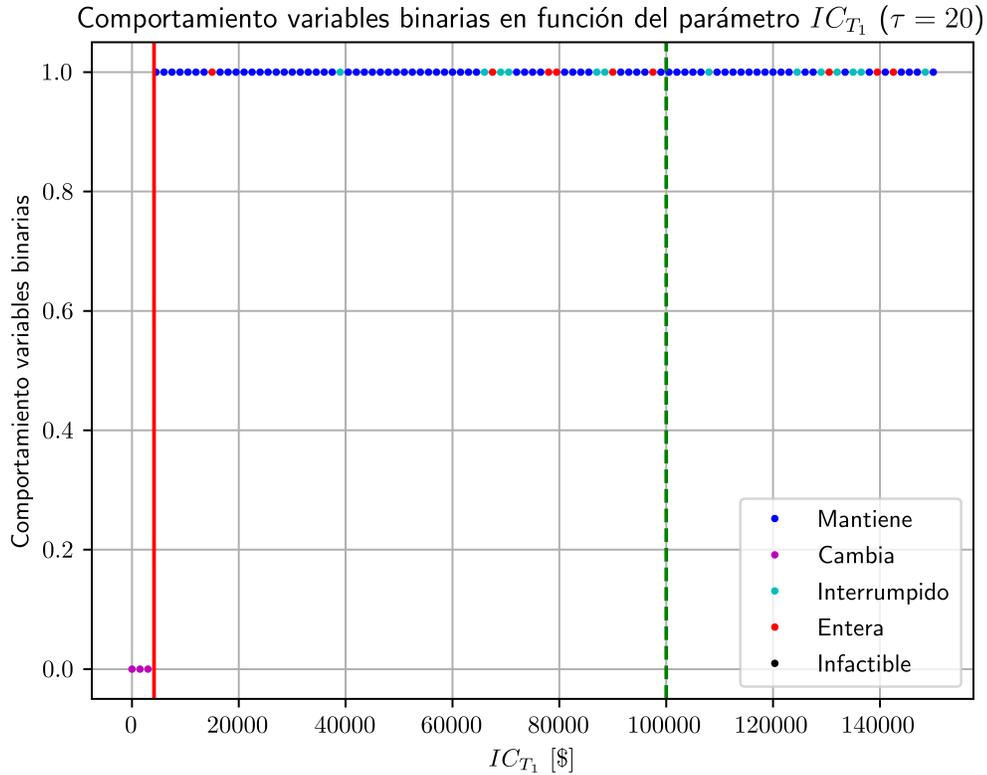


Figura E.19: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} con $\tau = 20$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.

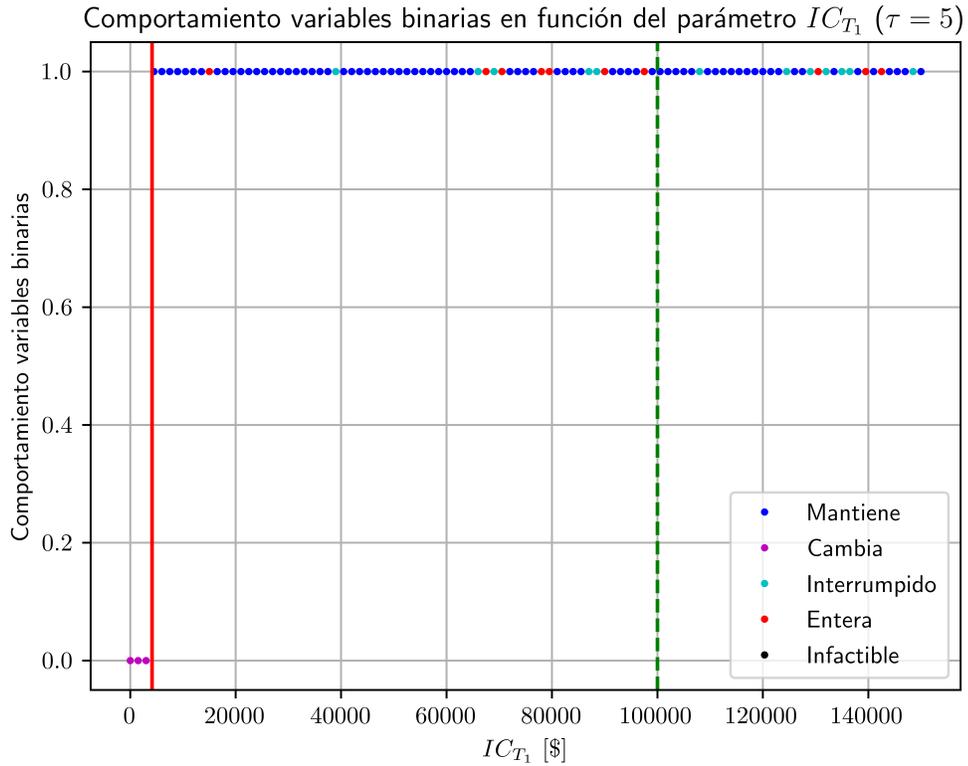


Figura E.20: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} con $\tau = 5$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.

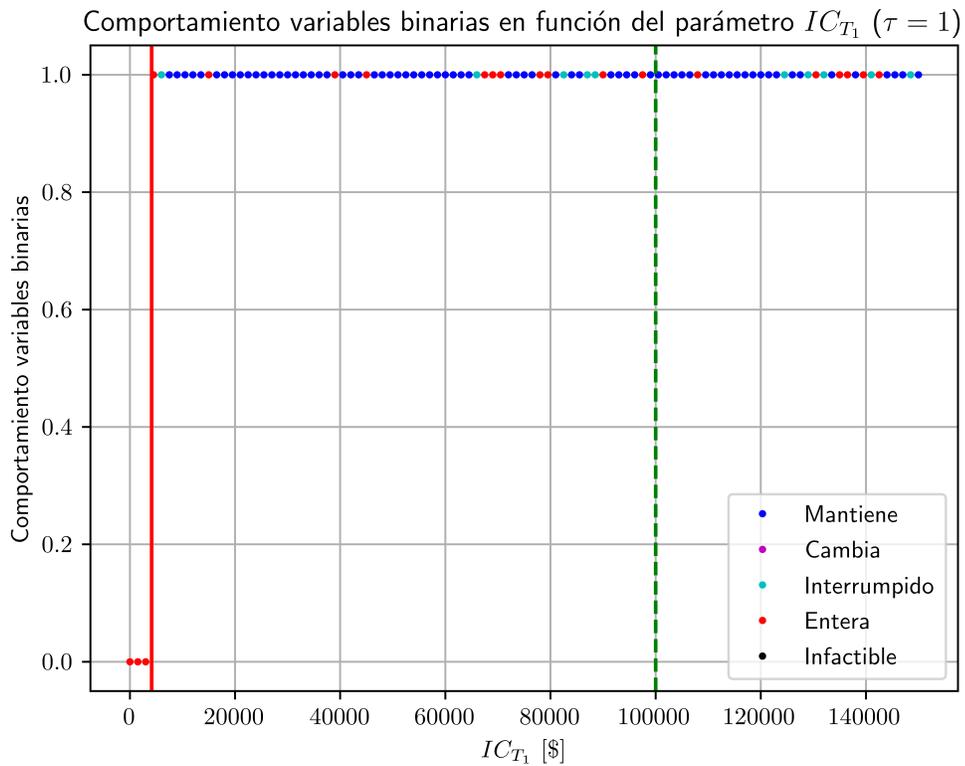


Figura E.21: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_1} con $\tau = 1$. Línea roja: límite inferior calculado. Línea verde: límite superior estimado por número de iteraciones.

E.4.5. Parámetro IC_{T_2}

Las diferentes ejecuciones del problema con cambios en IC_{T_2} , resumidas en la Figura E.5, requieren tiempos que presentan un comportamiento similar a lo obtenido con el parámetro IC_{T_1} , según se observa en la Figura E.22. Específicamente, el 88 % requiere un tiempo de 5 veces el nominal para ejecutarse sin que se interrumpa la resolución, mientras que el restante 12 % de los casos no se obtendrían sin interrupción para tiempos inferiores a 100 veces el original.

Número de iteraciones de IC_{T_2} resueltas según el tiempo de ejecución (ponderado por τ)

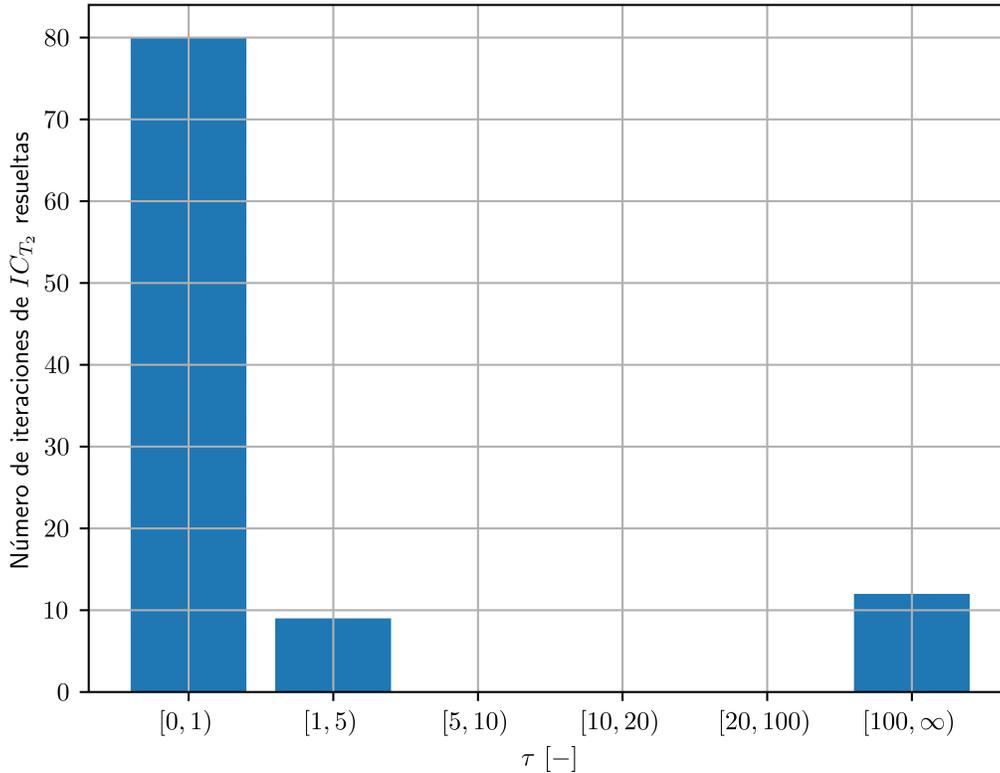


Figura E.22: Número de iteraciones de IC_{T_2} resueltas según el tiempo de ejecución.

Para analizar cómo se comportarían las múltiples resoluciones de la Figura E.5 si se acotara el tiempo de ejecución por defecto, se generaron escenarios con diferentes valores de τ y sus resultados son descritos a continuación:

- $\tau = 5, 10, 20, 100$: Los valores en las variables binarias son iguales a los obtenidos en la Figura E.5, presentando también la misma clasificación en cuanto al tipo de solución, según se aprecia en la Figura E.23.
- $\tau = 1$: Para este escenario los valores de las variables binarias se mantienen en comparación con los casos anteriores, cuya única diferencia corresponde a un valor que anteriormente presentaba la denominación de *Interrumpido* y con la disminución en el tiempo de resolución pasó a ser una solución *Entera*, según lo reflejado en la Figura E.24.

IC_{T_2} es el parámetro del presente ejemplo que presenta un menor efecto en las variables binarias al acotarse su tiempo de resolución con valores bajo el límite por defecto de GAMS.

Los resultados de la Figura E.23 concuerdan con lo esperado, ya que la distribución de tiempo de la Figura E.22 mostraba que la mayoría de los casos es posible resolverlos con $\tau \leq 5$. Particularmente, para IC_{T_2} el uso de $\tau = 1$ permitiría obtener también las soluciones originales y con solamente un caso cuyo tipo de solución difiere.

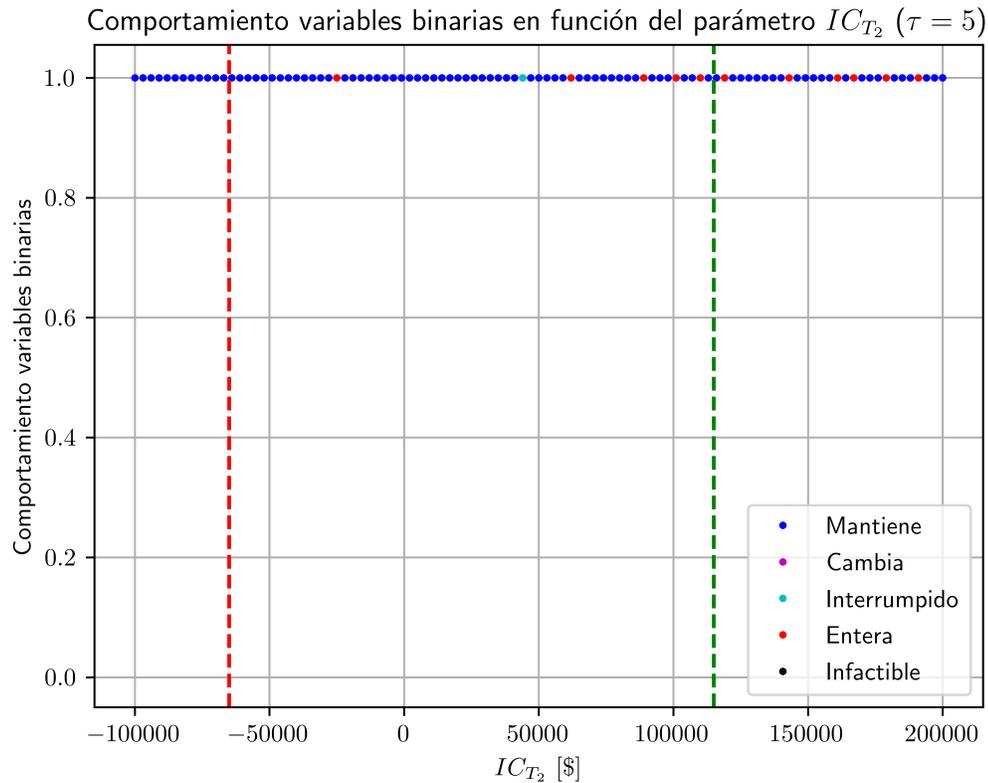


Figura E.23: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_2} con $\tau = 5$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.

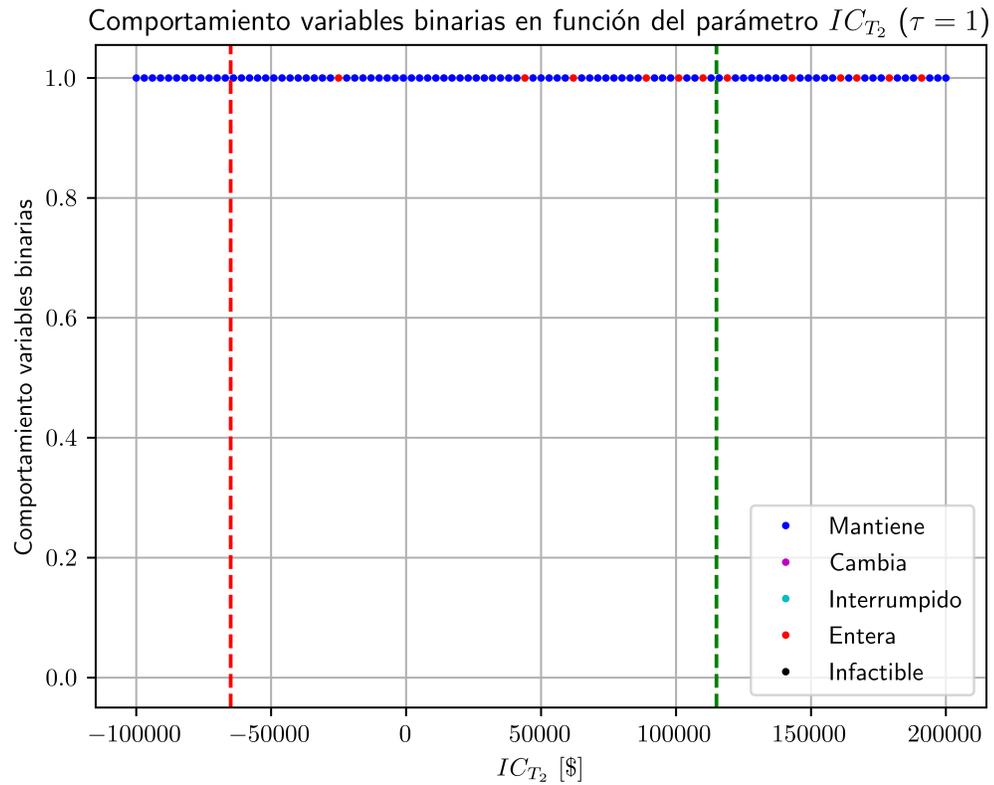


Figura E.24: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_2} con $\tau = 1$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.

E.4.6. Parámetro IC_{T_3}

En contraste con los dos costos anteriores, IC_{T_1} y IC_{T_2} , IC_{T_3} presenta una distribución de tiempos diferente. Aproximadamente la mitad de las múltiples resoluciones de este parámetro, que se encuentran representadas en la Figura E.6, requieren un tiempo inferior al nominal para obtener la solución del problema, tal como se aprecia en la Figura E.25. Adicionalmente, se observa que un 32% de los casos requiere un ponderador de tiempo en el siguiente rango: $1 \leq \tau < 5$, siendo un porcentaje superior a los dos costos anteriores. Finalmente, alrededor del 15% necesita un tiempo superior a 100 veces el original para obtener su respuesta sin interrupciones.

Número de iteraciones de IC_{T_3} resueltas según el tiempo de ejecución (ponderado por τ)

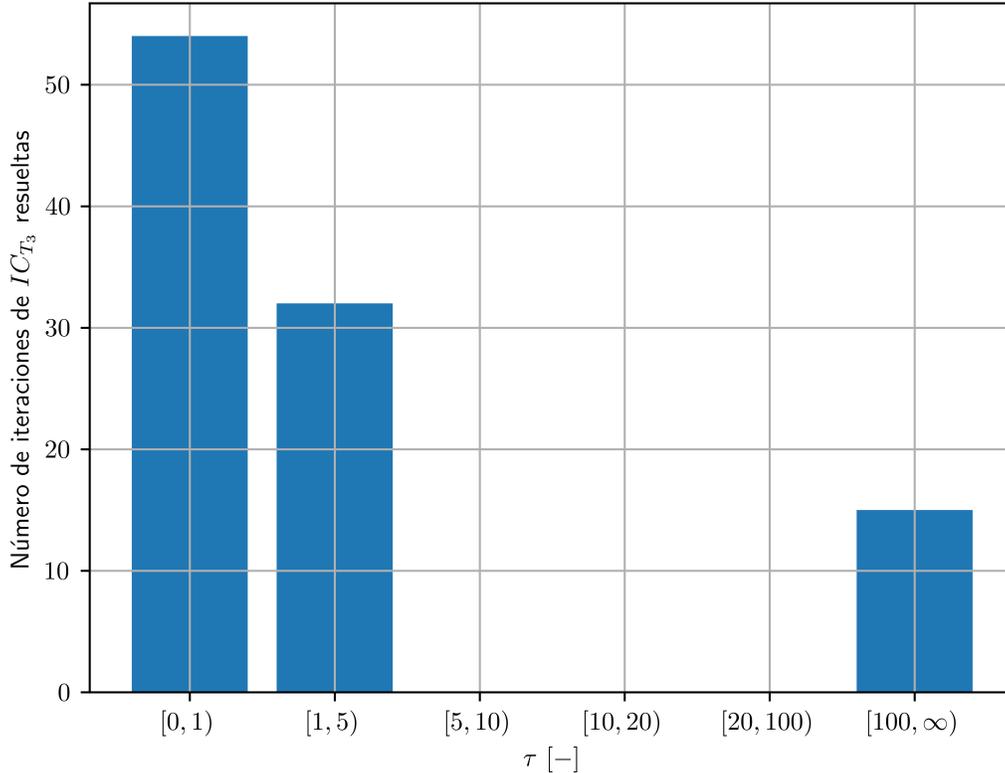


Figura E.25: Número de iteraciones de IC_{T_3} resueltas según el tiempo de ejecución.

Para comprender cómo cambian las soluciones según el tiempo de ejecución, se resolvieron las evaluaciones anteriores para diferentes valores de τ , obteniéndose los siguientes escenarios:

- $\tau = 5, 10, 20, 100$: Los valores en las variables binarias son iguales a los mencionados previamente en la Figura E.6, obteniéndose la misma clasificación para estos, según se aprecia en la Figura E.26.
- $\tau = 1$: Cuando se limita el tiempo de resolución según el requerido en el problema original, se observan cambios en los valores de las variables binarias. Particularmente, valores óptimos (*Cambia*) pasan a ser soluciones *Entera* y con un valor distinto al registrado cuando se posee mayor tiempo para la resolución de cada iteración. Esta situación se observa en la Figura E.27, notándose que los valores que se han modificado están fuera del rango de estabilidad estimado anteriormente.

Los resultados obtenidos para los casos $\tau = 1$ y $\tau \geq 5$ presentan un contraste importante, considerando que cuando el tiempo de resolución permitido es mayor o igual a 5 veces el requerido en el problema original, los valores de las variables binarias y la condición del modelo y solver son iguales a los casos en que está permitido alcanzar el tiempo por defecto de GAMS (1.000[s]). Por el contrario, limitar el tiempo de resolución a lo requerido en el caso original, genera que existan cambios en los valores de las variables binarias, lo que puede inducir a errores en la determinación del rango de estabilidad.

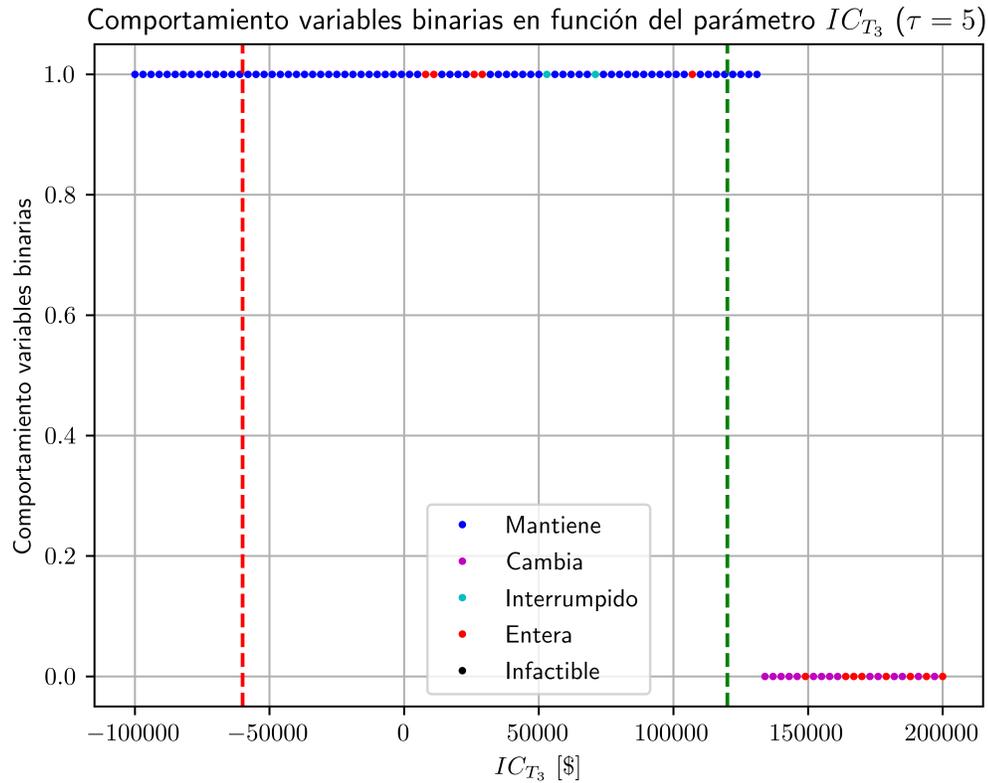


Figura E.26: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_3} con $\tau = 5$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.

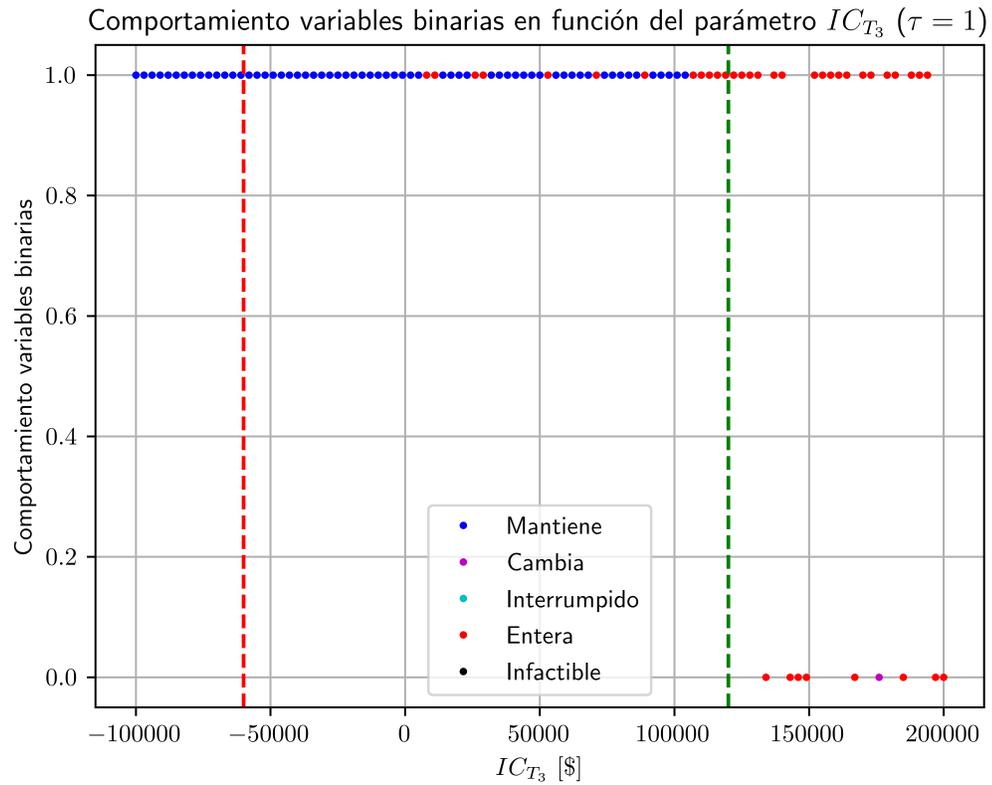


Figura E.27: Comportamiento de las variables binarias frente a cambios en el parámetro IC_{T_3} con $\tau = 1$. Línea roja: límite inferior estimado por número de iteraciones. Línea verde: límite superior estimado por número de iteraciones.

E.5. Análisis de inecuaciones

En el presente ejemplo se observa la siguiente distinción respecto a los dos anteriores: No presenta restricciones de desigualdad en que se observe directamente el efecto de los parámetros analizados, con excepción de las relacionadas con la existencia de las conexiones, formuladas como problemas del tipo *Big M* (si no existe la conexión, no hay flujo).

Los parámetros $FPU_{P_1}^{in}$ y $FPU_{P_2}^{in}$ se observan en las restricciones de balances de masa (igualdad) y en las del tipo *Big M*, por lo que no permitirían obtener información adicional respecto a cuánto es posible modificarlos sin que exista un cambio en las variables binarias. En el caso de r_{T_3} , su aparición también está relacionada con una restricción de igualdad y no aportaría información adicional directamente. Para los flujos y el porcentaje de remoción, los antecedentes que se tienen y que pueden aportar en la búsqueda de los límites del rango de estabilidad provienen de su naturaleza (no pueden ser negativos, y en el caso de r_{T_3} tampoco puede superar el 100%), por lo que las ecuaciones en que se observan directamente incluidos no generan nueva información. De una forma indirecta, se podría analizar su participación en las ecuaciones y cotas adicionales, aunque estas fueron incluidas por los autores para acelerar la búsqueda de la solución y no debiesen modificar el resultado.

En cuanto a los costos de inversión IC_{T_1} , IC_{T_2} e IC_{T_3} , estos no se encuentran en las restricciones del problema y aparecen solamente en la función objetivo, por lo que tampoco aportarían información adicional para la obtención del rango de estabilidad mediante el análisis de las restricciones activas.

La situación relatada respecto a los parámetros del ejemplo 3, permite destacar la siguiente consideración: Sería conveniente emplear el análisis de las restricciones activas en aquellos casos en que se observe una relación o inclusión directa de los parámetros analizados en restricciones de desigualdad, ya que en otros casos la búsqueda de la influencia de los parámetros podría tener el efecto contrario al deseado, es decir, generar un mayor tiempo de obtención del rango de estabilidad al desarrollar una revisión en profundidad de las restricciones existentes.

E.6. Optimización de parámetros

La resolución del problema reformulado requirió que algunos parámetros se consideren como variables, mientras que las variables binarias se incorporan como parámetros, presentando los valores obtenidos en la solución del problema original (resuelto con los valores nominales de los parámetros). Particularmente, los parámetros $FPU_{P_1}^{in}$, $FPU_{P_2}^{in}$, r_{T_3} , IC_{T_1} , IC_{T_2} e IC_{T_3} pasan a ser variables, uno a la vez.

La solución de los subproblemas para cada parámetro, el tiempo utilizado para determinarla, el empleado en la evaluación de estos valores en el caso original y su factibilidad de uso como nuevo punto inicial se presenta en la Tabla E.18. Se observa que es posible emplear una nueva estimación inicial en cuatro de los 12 límites a determinar, cuya implementación se efectúa en el algoritmo propuesto en el Anexo E.7. Existen diversas razones de la exclusión de algunas de las soluciones de los problemas reformulados para ser utilizadas como estimación inicial, que son destacadas a continuación.

Tabla E.18: Parámetros optimizados del ejemplo ilustrativo 3.

Parámetro optimizado [UN]	Valor obtenido	Tiempo ejec. reformulado [s]	Tiempo de evaluación [s]	¿Mantiene binaria?
$FPU_{P_1,op}^{in,min}$ [t/h]	NOp ^a	8,70	-	No
$FPU_{P_1,op}^{in,max}$ [t/h]	NOp ^a	8,70	-	No
$FPU_{P_2,op}^{in,min}$ [t/h]	NOp ^a	8,70	-	No
$FPU_{P_2,op}^{in,min}$ [t/h]	NOp ^a	8,76	-	No
$r_{T_3,op}^{min}$ [%]	80,8	0,30	1,08	Sí
$r_{T_3,op}^{max}$ [%]	100,0	0,16	9,39	No
$IC_{T_1,op}^{min}$ [] ^b	0,0	0,15	1,88	No
$IC_{T_1,op}^{max}$ [] ^b	10 ⁵⁰	7,73	0,28	Sí
$IC_{T_2,op}^{min}$ [] ^b	0,0	0,16	1,42	Sí
$IC_{T_2,op}^{max}$ [] ^b	10 ⁵⁰	7,76	0,19	No ^c
$IC_{T_3,op}^{min}$ [] ^b	0,0	0,13	1,03	Sí
$IC_{T_3,op}^{max}$ [] ^b	10 ⁵⁰	7,78	0,19	No ^c

^a NOp: No optimizado. La búsqueda de la solución es interrumpida por alcanzar el límite de tiempo establecido ($\tau = 5, 8,7[s]$ aprox.) y no se entrega una solución.

^b El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [€]. Ver Anexo E.1

^c La evaluación del parámetro optimizado en el problema original generó que la solución sea infactible.

Para los parámetros $FPU_{P_1}^{in}$, $FPU_{P_2}^{in}$ y r_{T_3} , el modelo original considera la inclusión de estos parámetros en cálculos previos a la resolución del problema de optimización, ya que se emplean las cotas propuestas por Ahmetovic y Grossmann [73]. Debido a esto, fue necesario eliminarlas para la resolución del problema reformulado. Particularmente, para el porcentaje de remoción (r_{T_3}) es necesario eliminar un número inferior de ecuaciones en contraste con los flujos mencionados ($FPU_{P_1}^{in}$ y $FPU_{P_2}^{in}$), por lo que el problema es resuelto en un tiempo inferior al límite impuesto ($\tau = 5$). Por el contrario, la exclusión de las ecuaciones previas en

que están involucrados los flujos ocasiona que la resolución del problema de optimización no se pueda efectuar en un tiempo inferior al límite impuesto, no obteniéndose solución en aquel período de tiempo, razón por la que no existió un valor para ser utilizado en la evaluación del problema original. Recordando que el propósito de aquellas cotas adicionales era resolver en menor tiempo el problema de optimización, su eliminación afectó al tiempo requerido en la reformulación.

En relación a los costos (IC_{T_1} , IC_{T_2} e IC_{T_3}), se obtuvieron los mismos resultados para los mínimos entre sí, ocurriendo una situación similar respecto a los máximos. Particularmente, la evaluación de $IC_{T_1,op}^{max}$ permite mantener la solución original de las variables binarias, sugiriendo que no está acotado. En contraste, la evaluación de $IC_{T_2,op}^{max}$ y $IC_{T_3,op}^{max}$ ocasiona que la solución sea infactible.

Finalmente, se destaca que en este problema nuevamente se redujo el número de decimales de la solución del problema reformulado, alcanzado solamente aquellos influyentes según la tolerancia correspondiente. Específicamente, esto se aplicó para $r_{T_3,op}^{max}$ y así evitar que la solución redondeada generara que su evaluación en el problema original estuviese fuera del rango de estabilidad.

E.7. Rango de estabilidad (algoritmo propuesto)

E.7.1. Parámetro $FPU_{P_1}^{in}$

La determinación del límite inferior del rango de estabilidad de $FPU_{P_1}^{in}$ se logró mediante las iteraciones que se encuentran en la Tabla E.19, notándose que se iniciaron desde el valor nominal tras la imposibilidad de emplear una nueva estimación inicial con los criterios definidos en la *Optimización de parámetros*. En cuanto al límite superior, por las mismas razones se inicia también desde el valor nominal, requiriéndose las iteraciones reflejadas en la Tabla E.20.

Tabla E.19: Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$ mediante el algoritmo propuesto.

i	$FPU_{P_1}^{in}$	$y' = y_0?$	$FPU_{P_1}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	40,0	Sí	—	—	1,68
1	30,0	Sí	—	—	1,33
2	20,0	Sí	—	—	1,97
3	10,0	No	20,0	$1,0 > 0,016$	9,17
Ciclo 2: $\Delta l = 1$					
0	20,0	Sí	—	—	1,98
1	19,0	Sí	—	—	1,77
2	18,0	Sí	—	—	8,89
3	17,0	Sí	—	—	9,03
4	16,0	Sí	—	—	2,65
5	15,0	Sí	—	—	2,35
6	14,0	Sí	—	—	3,41
7	13,0	Sí	—	—	3,39
8	12,0	Sí	—	—	3,41
9	11,0	No	12,0	$0,1 > 0,016$	9,53
Ciclo 3: $\Delta l = 0,1$					
0	12,0	Sí	—	—	4,42
1	11,9	Sí	—	—	9,39
2	11,8	Sí	—	—	5,35
3	11,7	Sí	—	—	9,09
4	11,6	Sí	—	—	7,69
5	11,5	Sí	—	—	6,72
6	11,4	No	11,5	$0,01 < 0,016$	9,16
Tiempo iteraciones: 112,38[s]					
Tiempo total (iteraciones + opt. de parámetros): 121,08[s]					

Tabla E.20: Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_1}^{in}$ mediante el algoritmo propuesto.

i	$FPU_{P_1}^{in}$	$y' = y_0?$	$FPU_{P_1}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	40,0	Sí	—	—	1,57
1	50,0	No	40,0	$1,0 > 0,016$	1,55
Ciclo 2: $\Delta l = 1$					
0	40,0	Sí	—	—	1,33
1	41,0	Sí	—	—	1,29
2	42,0	Sí	—	—	1,29
3	43,0	Sí	—	—	1,38
4	44,0	Sí	—	—	9,46
5	45,0	Sí	—	—	1,43
6	46,0	Sí	—	—	9,56
7	47,0	Sí	—	—	1,29
8	48,0	Sí	—	—	1,30
9	49,0	Sí	—	—	1,55
10	50,0	No	49,0	$0,1 > 0,016$	1,65
Ciclo 3: $\Delta l = 0,1$					
0	49,0	Sí	—	—	1,38
1	49,1	Sí	—	—	1,27
2	49,2	Sí	—	—	1,06
3	49,3	Sí	—	—	1,11
4	49,4	Sí	—	—	1,17
5	49,5	Sí	—	—	1,05
6	49,6	Sí	—	—	1,10
7	49,7	Sí	—	—	1,08
8	49,8	Sí	—	—	1,02
9	49,9	Sí	—	—	0,98
10	50,0	No	49,9	$0,01 < 0,016$	1,41
Tiempo iteraciones: 47,28[s]					
Tiempo total (iteraciones + opt. de parámetros): 55,98[s]					

A partir de la información de la Tabla E.21 se contrastan los resultados y tiempos empleados mediante el algoritmo referencial y el propuesto. En cuanto a los resultados, se observa que el límite inferior corresponde al mismo valor para ambas alternativas, aunque el uso del algoritmo referencial permite reducir el tiempo necesario en un 97,7 %.

En relación al límite superior, el valor obtenido difiere entre la versión referencial y la propuesta. El contraste radica en que la detención del algoritmo referencial se produce luego de alcanzar el número máximo de iteraciones para un mismo paso en 49,0, sugiriendo al

usuario actualizar los parámetros de búsqueda en caso de desear continuar el proceso de determinación del límite superior. Por el contrario, la versión propuesta se detiene tras cumplir con el criterio de tolerancia establecido, evitando la interacción con el usuario y finalizando en un tiempo inferior a la versión referencial.

Tabla E.21: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $FPU_{P_1}^n$.

Límite	Valor ref. [t/h]	Valor prop. [t/h]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	11,5	11,5	5.195,83	121,08	-97,7
Superior	49,0 ^a	49,9	2.063,19	55,98	-97,3

^a Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

La reducción de tiempo para ambos límites está mayormente influenciada por el tiempo límite por iteración incluido en el algoritmo propuesto, evitando alcanzar valores del orden de miles de segundos.

E.7.2. Parámetro $FPU_{P_2}^{in}$

Las iteraciones realizadas en el cálculo del límite inferior de $FPU_{P_2}^{in}$ se encuentran en la Tabla E.22, iniciándose desde el valor nominal al igual que ocurrió con $FPU_{P_1}^{in}$. También se utiliza el valor nominal para comenzar la búsqueda del límite superior según se observa en la Tabla E.23, ante la ausencia de una mejor adivinanza inicial.

Tabla E.22: Determinación del límite inferior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$ mediante el algoritmo propuesto.

i	$FPU_{P_2}^{in}$	$y' = y_0?$	$FPU_{P_2}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	50,0	Sí	—	—	1,41
1	40,0	No	50,0	$1,0 > 0,02$	0,63
Ciclo 2: $\Delta l = 1$					
0	50,0	Sí	—	—	1,59
1	49,0	Sí	—	—	1,10
2	48,0	Sí	—	—	1,14
3	47,0	Sí	—	—	0,98
4	46,0	Sí	—	—	0,96
5	45,0	Sí	—	—	1,10
6	44,0	Sí	—	—	1,00
7	43,0	Sí	—	—	9,11
8	42,0	Sí	—	—	9,23
9	41,0	Sí	—	—	0,81
10	40,0	No	41,0	$0,1 > 0,02$	0,67
Ciclo 3: $\Delta l = 0,1$					
0	41,0	Sí	—	—	1,08
1	40,9	Sí	—	—	0,67
2	40,8	Sí	—	—	0,85
3	40,7	Sí	—	—	0,93
4	40,6	Sí	—	—	0,71
5	40,5	Sí	—	—	0,73
6	40,4	Sí	—	—	0,82
7	40,3	Sí	—	—	0,77
8	40,2	Sí	—	—	0,78
9	40,1	Sí	—	—	0,72
10	40,0	No	40,1	$0,01 < 0,02$	0,67
Tiempo iteraciones: 38,46[s]					
Tiempo total (iteraciones + opt. de parámetros): 47,16[s]					

Tabla E.23: Determinación del límite superior del rango de estabilidad del parámetro $FPU_{P_2}^{in}$ mediante el algoritmo propuesto.

i	$FPU_{P_2}^{in}$	$y' = y_0?$	$FPU_{P_2}^{in'}$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	50,0	Sí	–	–	1,53
1	60,0	No	50,0	$1,0 > 0,02$	2,37
Ciclo 2: $\Delta l = 1$					
0	50,0	Sí	–	–	1,46
1	51,0	Sí	–	–	1,05
2	52,0	Sí	–	–	1,44
3	53,0	Sí	–	–	3,43
4	54,0	Sí	–	–	2,94
5	55,0	Sí	–	–	2,83
6	56,0	No	55,0	$0,1 > 0,02$	2,58
Ciclo 3: $\Delta l = 0,1$					
0	55,0	Sí	–	–	2,77
1	55,1	Sí	–	–	2,08
2	55,2	Sí	–	–	3,27
3	55,3	Sí	–	–	3,41
4	55,4	No	55,3	$0,01 < 0,02$	3,51
Tiempo iteraciones: 34,67[s]					
Tiempo total (iteraciones + opt. de parámetros): 43,43[s]					

Los resultados de las Tablas E.22 y E.23 se comparan con los obtenidos mediante el algoritmo referencial en la Tabla E.24, notándose que el límite superior alcanza el mismo valor y en un tiempo similar para ambos algoritmos. Por el contrario, al buscar el límite inferior se obtienen valores distintos y logrados en tiempos con diferentes órdenes de magnitud.

Tabla E.24: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de $FPU_{P_2}^{in}$.

Límite	Valor ref. [t/h]	Valor prop. [t/h]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	41,0 ^a	40,1	1.740,54	47,16	–97,3
Superior	55,3	55,3	43,40	43,43	0,1

^a Último valor evaluado, sugiriendo el algoritmo actualizar los parámetros de búsqueda si se desea continuar.

La diferencia en los resultados de ambos algoritmos para el límite inferior se debe a que la versión referencial se detiene en el valor 41,0 para sugerir actualizar los parámetros de búsqueda, reconociéndose solamente que el límite se encuentra entre 40,0 y 41,0, sin alcanzar la tolerancia definida. Por el contrario, la versión propuesta logra el nivel de precisión que permite que el criterio de tolerancia genere la finalización de las iteraciones. Pese al mayor número de iteraciones en este último caso, se logra reducir el tiempo en un 97,3% debido a la imposición del tiempo límite por iteración para obtener la solución en un tiempo razonable.

E.7.3. Parámetro r_{T_3}

Para la búsqueda del límite inferior de r_{T_3} se utiliza la estimación inicial determinada mediante la *Optimización de parámetros*, correspondiente al límite buscado. Para poder concluir aquello, se efectuaron las iteraciones de la Tabla E.25, finalizando tras alcanzar la tolerancia establecida. Por el contrario, el valor determinado en el Anexo E.6 no se consideró por generar un cambio en las variables binarias, necesitándose las iteraciones de la Tabla E.26.

Tabla E.25: Determinación del límite inferior del rango de estabilidad del parámetro r_{T_3} mediante el algoritmo propuesto.

i	r_{T_3}	$y' = y_0?$	r_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	80,8	Sí	–	–	1,16
1	70,8	No	80,8	$1,0 > 0,038$	1,83
Ciclo 2: $\Delta l = 1$					
0	80,8	Sí	–	–	1,15
1	79,8	No	80,8	$0,1 > 0,038$	9,23
Ciclo 3: $\Delta l = 0,1$					
0	80,8	Sí	–	–	1,43
1	80,7	No	80,8	$0,01 < 0,038$	10,64
Tiempo iteraciones: 25,44[s]					
Tiempo total (iteraciones + opt. de parámetros): 26,82[s]					

Observando las ejecuciones efectuadas para determinar ambos límites, se destaca que para el límite inferior las iteraciones fueron necesarias para reducir el paso de búsqueda hasta cumplir con el criterio de detención por alcance de la tolerancia establecida. En cuanto al límite superior, la evaluación del valor 105,0 no se efectúa por encontrarse fuera del rango factible según la información previa incorporada en el algoritmo (porcentaje de remoción debe estar entre 0 y 100%).

Al comparar los resultados anteriores con los obtenidos mediante el algoritmo referencial, se observa que ambos procedimientos determinan el mismo rango de estabilidad, según se resume en la Tabla E.27. Se destaca también que la *Optimización de parámetros* generó una diferencia de tiempo entre ambos casos, ya que para el límite inferior permitió iniciar la búsqueda en el límite solicitado, reduciendo en un 78,4% el tiempo en relación a lo requerido en la versión referencial. Por el contrario, para el límite superior se empleó tiempo en determinar un posible nuevo valor inicial y, tras descartarse, genera que en total se haya elevado en un 43,4% el tiempo necesario para determinar la cota superior del rango de estabilidad. Observando el tiempo total para determinar ambos límites, el algoritmo propuesto es más rápido que el referencial.

Tabla E.26: Determinación del límite superior del rango de estabilidad del parámetro r_{T_3} mediante el algoritmo propuesto.

i	r_{T_3}	$y' = y_0?$	r_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10$					
0	95,0	Sí	–	–	1,27
1	105,0	No	95,0	$1,0 > 0,038$	0,00
Ciclo 2: $\Delta l = 1$					
0	95,0	Sí	–	–	1,37
1	96,0	Sí	–	–	1,12
2	97,0	Sí	–	–	1,87
3	98,0	No	97,0	$0,1 > 0,038$	1,51
Ciclo 3: $\Delta l = 0,1$					
0	97,0	Sí	–	–	2,27
1	97,1	Sí	–	–	2,66
2	97,2	Sí	–	–	2,08
3	97,3	Sí	–	–	2,07
4	97,4	Sí	–	–	2,44
5	97,5	No	97,4	$0,01 < 0,038$	2,82
Tiempo iteraciones: 21,48[s]					
Tiempo total (iteraciones + opt. de parámetros): 31,03[s]					

Tabla E.27: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de r_{T_3} .

Límite	Valor ref. [%]	Valor prop. [%]	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	80,8	80,8	124,42	26,82	–78,4
Superior	97,4	97,4	21,64	31,03	43,4

E.7.4. Parámetro IC_{T_1}

Para encontrar el límite inferior de IC_{T_1} se efectuaron las iteraciones de la Tabla E.28, descartándose un nuevo valor inicial proveniente de la *Optimización de parámetros* por generar una solución diferente.

Tabla E.28: Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_1} mediante el algoritmo propuesto.

i	IC_{T_1}	$y' = y_0?$	IC_{T_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	10.000	Sí	—	—	1,37
1	0	No	10.000	$1.000 > 4,0$	2,00
Ciclo 2: $\Delta l = 1.000$					
0	10.000	Sí	—	—	1,37
1	9.000	Sí	—	—	1,31
2	8.000	Sí	—	—	1,43
3	7.000	Sí	—	—	1,29
4	6.000	Sí	—	—	4,61
5	5.000	Sí	—	—	1,85
6	4.000	No	5.000	$100 > 4,0$	9,57
Ciclo 3: $\Delta l = 100$					
0	5.000	Sí	—	—	1,83
1	4.900	Sí	—	—	1,78
2	4.800	Sí	—	—	1,93
3	4.700	Sí	—	—	2,17
4	4.600	Sí	—	—	8,99
5	4.500	Sí	—	—	2,60
6	4.400	Sí	—	—	2,58
7	4.300	Sí	—	—	2,39
8	4.200	Sí	—	—	2,57
9	4.100	No	4.200	$10 > 4,0$	2,79
Ciclo 4: $\Delta l = 10$					
0	4.200	Sí	—	—	3,73
1	4.190	Sí	—	—	2,75
2	4.180	Sí	—	—	2,77
3	4.170	Sí	—	—	2,33
4	4.160	Sí	—	—	2,94
5	4.150	Sí	—	—	2,97
6	4.140	No	4.150	$1,0 < 4,0$	4,26
Tiempo iteraciones: 76,18[s]					
Tiempo total (iteraciones + opt. de parámetros): 78,21[s]					

En contraste con el límite inferior, para el superior se utiliza como valor inicial el estimado en el Anexo E.6, siendo el máximo alcanzable, según se observa en la Tabla E.29.

Tabla E.29: Determinación del límite superior del rango de estabilidad del parámetro IC_{T_1} mediante el algoritmo propuesto.

i	IC_{T_1}	$y' = y_0?$	IC_{T_1}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	10^{50}	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 8,01[s]					

Al comparar los resultados principales de las Tablas E.28 y E.29 con los determinados mediante el algoritmo referencial, se observa que alcanzan el mismo límite inferior, según se resume en la Tabla E.30. Por el contrario, los resultados para el límite superior difieren, ya que el algoritmo referencial detiene las iteraciones y sugiere que el usuario actualice los parámetros de búsqueda, mientras que el algoritmo propuesto no requiere aquella interacción con el usuario y finaliza determinando que no existe un límite superior o este es mayor al máximo valor expresable en GAMS.

Tabla E.30: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de IC_{T_1} .

Límite	Valor ref. [] ^a	Valor prop. [] ^a	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	4.150	4.150	1.131,07	78,21	–93,1
Superior	100.000^b	NA ^c	2.042,17	8,01	–99,6

^a El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [\\$]. Ver Anexo E.1

^b Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

^c NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresado en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

Analizando los tiempos requeridos entre ambos algoritmos se observa que la versión propuesta logra una reducción por sobre el 93 % para ambos casos. Para el límite inferior la reducción se debe al límite de tiempo impuesto por iteración, mientras que para el superior aquel límite influye también, junto con el uso de una mejor estimación inicial para evitar requerir evaluaciones adicionales desde el valor obtenido a partir de la *Optimización de parámetros*.

E.7.5. Parámetro IC_{T_2}

La determinación del límite inferior del rango de estabilidad de IC_{T_2} se efectuó a partir de la *Optimización de parámetros*, determinándose el valor 0 como el punto inicial. Considerando que el costo no puede ser negativo, el valor nulo corresponde al límite inferior del rango de estabilidad, tal como se ilustra en la Tabla E.31. Por otro lado, el procedimiento implementado en el Anexo E.6 no permite obtener una nueva estimación inicial para la búsqueda del límite superior, empleándose el valor nominal y requiriéndose las iteraciones de la Tabla E.32. Tras no registrar cambios en las variables binarias al aumentar el valor del parámetro, el algoritmo finaliza las evaluaciones y sugiere actualizar los parámetros de búsqueda si se desea continuar iterando.

Tabla E.31: Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_2} mediante el algoritmo propuesto.

i	IC_{T_2}	$y' = y_0?$	IC_{T_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	0,0	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 1,58[s]					

Tabla E.32: Determinación del límite superior del rango de estabilidad del parámetro IC_{T_2} mediante el algoritmo propuesto.

i	IC_{T_2}	$y' = y_0?$	IC_{T_2}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	25.000	Sí	–	–	1,55
1	35.000	Sí	–	–	1,40
2	45.000	Sí	–	–	1,42
3	55.000	Sí	–	–	0,71
4	65.000	Sí	–	–	1,17
5	75.000	Sí	–	–	1,06
6	85.000	Sí	–	–	1,16
7	95.000	Sí	–	–	1,04
8	105.000	Sí	–	–	1,17
9	115.000	Sí	–	–	1,04
10	125.000	Sí	–	–	1,31
Tiempo iteraciones: 13,03[s]					
Tiempo total (iteraciones + opt. de parámetros): 20,98[s]					

Para analizar el rendimiento del algoritmo propuesto y los resultados anteriores, se realiza un contraste con lo obtenido mediante el algoritmo referencial, resumiéndose en la Tabla E.33. Se observa que los límites obtenidos son diferentes y que es posible reducir el tiempo necesario para uno de los límites, mientras que para el otro aumenta.

Tabla E.33: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de IC_{T_2} .

Límite	Valor ref. [] ^a	Valor prop. [] ^a	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	-65.000 ^b	0,0	1.026,65	1,58	-99,8
Superior	115.000 ^b	125.000 ^b	12,76	20,98	64,6

^a El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [\\$]. Ver Anexo E.1

^b Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

La explicación de la diferencia presentada en el procedimiento y resultados del límite inferior se fundamenta en el uso de *Información previa* sobre el parámetro en estudio. En particular, aquella consideración ocasiona que la solución sea 0 según el algoritmo propuesto, debido a que es el menor valor posible. Por el contrario, el algoritmo referencial evalúa valores inferiores a aquel límite lógico para el costo IC_{T_2} , generando un mayor tiempo de búsqueda al realizar iteraciones que no son necesarias para este caso. Junto con ello, la adivinanza inicial obtenida desde la *Optimización de parámetros* permite comenzar la búsqueda precisamente desde el límite buscado, ocasionando finalmente una disminución del 99,8% del tiempo.

Respecto al límite superior, para ambos algoritmos se alcanza un valor en el que no se han producido cambios en las variables binarias, sugiriendo al usuario actualizar los parámetros de búsqueda en caso de necesitar continuar con las iteraciones. Sin embargo, el algoritmo propuesto emplea tiempo adicional en evaluar un posible nuevo punto inicial, que es descartado. Junto con ello, efectúa una evaluación más al permitir que el número de iteraciones para un mismo paso (i) alcance el valor de 10, mientras que en la versión referencial se considera lograr un valor inferior a 10. A partir de lo anterior, para este caso el algoritmo propuesto ocasiona un aumento del 64,6% en el tiempo de implementación. De todas formas, la reducción lograda para el límite inferior compensa el aumento en la búsqueda del límite superior, por lo que la implementación completa del algoritmo propuesto se realiza en un tiempo menor al requerido por el algoritmo referencial.

E.7.6. Parámetro IC_{T_3}

Al igual que para el parámetro anterior, la búsqueda del límite inferior se puede realizar a partir de un nuevo punto inicial obtenido desde la *Optimización de parámetros*, correspondiente también al menor valor posible para el costo IC_{T_3} , ocasionando que no se requieran iteraciones adicionales, tal como se expone en la Tabla E.34. Para la búsqueda del límite superior se genera una situación similar a la de IC_{T_2} , ya que una nueva estimación inicial es descartada, requiriéndose las iteraciones de la Tabla E.35 para observar que no se genera un cambio en las variables binarias para los valores evaluados.

Tabla E.34: Determinación del límite inferior del rango de estabilidad del parámetro IC_{T_3} mediante el algoritmo propuesto.

i	IC_{T_3}	$y' = y_0?$	IC_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	0,0	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 1,16[s]					

Tabla E.35: Determinación del límite superior del rango de estabilidad del parámetro IC_{T_3} mediante el algoritmo propuesto.

i	IC_{T_3}	$y' = y_0?$	IC_{T_3}'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 10.000$					
0	30.000	Sí	–	–	1,42
1	40.000	Sí	–	–	1,04
2	50.000	Sí	–	–	1,00
3	60.000	Sí	–	–	1,24
4	70.000	Sí	–	–	1,25
5	80.000	Sí	–	–	1,30
6	90.000	Sí	–	–	1,25
7	100.000	Sí	–	–	2,07
8	110.000	Sí	–	–	3,18
9	120.000	Sí	–	–	2,08
10	130.000	Sí	–	–	3,04
Tiempo iteraciones: 18,87[s]					
Tiempo total (iteraciones + opt. de parámetros): 26,84[s]					

La comparación entre el algoritmo propuesto y el referencial se efectúa a partir de la información de la Tabla E.36, contrastándose los resultados y el tiempo requerido. Se observa que la versión referencial evalúa valores inferiores a los posibles para el costo IC_{T_3} , situación que no ocurre con la versión propuesta, lográndose una reducción del 90,8% del tiempo necesario en la determinación del límite inferior, disminución que está influenciada también por la nueva estimación inicial determinada.

Tabla E.36: Comparación del rendimiento del algoritmo referencial y del propuesto en la determinación del rango de estabilidad de IC_{T_3} .

Límite	Valor ref. [] ^a	Valor prop. [] ^a	Tiempo ref. [s]	Tiempo prop. [s]	Variación [%]
Inferior	-60.000 ^b	0,0	12,60	1,16	-90,8
Superior	120.000 ^b	130.000 ^b	19,42	26,84	38,2

^a El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [\\$]. Ver Anexo E.1

^b Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

En relación al límite superior, se observa un caso similar a lo ocurrido con IC_{T_2} , ya que el procedimiento efectuado en la *Optimización de parámetros* aumenta el tiempo de ejecución del algoritmo propuesto en contraste con el referencial. Además, se evalúa hasta un valor superior, razón por la que las evaluaciones adicionales producen un aumento del 38,2% del tiempo. Sin embargo, considerando el tiempo total requerido para determinar ambos límites, el algoritmo propuesto permite una respuesta más rápida que el referencial.

E.8. Índice de estabilidad

La determinación de los índices de estabilidad inferiores y superiores de los parámetros del ejemplo 3 se realiza a partir de la ecuación 5.4 propuesta en el Capítulo 5. Para este propósito se emplean los valores nominales y los límites del rango de estabilidad calculados en la sección E.7, obteniéndose los resultados de la Tabla E.37:

Tabla E.37: Índice de estabilidad inferior y superior de los parámetros del ejemplo 3.

Par. [UN]	Nominal	Rango de estabilidad	IE^{inf} [%]	IE^{sup} [%]
$FPU_{P_1}^{in}$ [t/h]	40	$11,5 \leq FPU_{P_1}^{in} \leq 49,9$	71,3	24,8
$FPU_{P_2}^{in}$ [t/h]	50	$40,1 \leq FPU_{P_2}^{in} \leq 55,3$	19,8	10,6
r_{T_3} [%]	95	$80,8 \leq Cost_{T_1} \leq 97,4$	14,9	2,5
IC_{T_1} [] ^a	10.000	$4.150 \leq IC_{T_1}$	58,5	NA ^b
IC_{T_2} [] ^a	25.000	$0 \leq IC_{T_2} \leq 125.000^c$	100,0	400,0
IC_{T_3} [] ^a	30.000	$0 \leq IC_{T_3} \leq 130.000^c$	100,0	333,3

^a El costo de inversión para unidades de tratamiento, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, posee unidades en [\$]. Ver Anexo E.1

^b NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresable en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

^c Último valor evaluado por el algoritmo, sugiriendo actualizar los parámetros de búsqueda si se desea continuar iterando.

Los resultados anteriores señalan que el parámetro crítico corresponde a r_{T_3} , ya que la proximidad entre el valor nominal y su límite superior es la de menor magnitud relativa en comparación con el resto de los parámetros. Específicamente, un error superior a 2,5 % en la estimación de r_{T_3} ocasionaría que la solución óptima fuese otra. En contraste, se destaca que los parámetros IC_{T_2} e IC_{T_3} son los que presentan el mayor índice de estabilidad, alcanzando el 100 %. Este límite se genera por la interpretación del costo, ya que no podría disminuir a un valor inferior a cero y además porque se utilizan conexiones con las plantas T_2 y T_3 en la red óptima, por lo que disminuir sus costos no debiese generar cambios en la elección de las instalaciones. Por el contrario, al estar escogidas las conexiones con estas plantas, un aumento en su costo podría ocasionar un cambio en las variables binarias, pero a partir de lo obtenido en el algoritmo referencial no es posible concluir respecto a valores superiores a 125.000 y 130.000 para los costos IC_{T_2} e IC_{T_3} , respectivamente.

Anexo F

Ejemplo ilustrativo 4

F.1. Formulación

La formulación detallada a continuación proviene del problema planteado por Gormaz-Cuevas et al. [75]. Esta estructura se analizó y se detectaron características que influyen en el desempeño del algoritmo propuesto. A partir de ello se concluyen simplificaciones necesarias para la correcta implementación del algoritmo, destacadas en la sección F.2.

F.1.1. Conjuntos

- W : Conjunto de fuentes de agua.
- ED : Conjunto de plantas existentes de tratamiento de agua potable (DWTP).
- EW : Conjunto de plantas existentes de tratamiento de agua residual (WWTP).
- IE : Conjunto de plantas de tratamiento de efluentes industriales (IWTP).
- $ND0$: Conjunto de plantas nuevas de tratamiento de agua potable (DWTP).
- MW : Conjunto de plantas de tratamiento de agua residual modificadas (WWTP).
- NW : Conjunto de plantas nuevas de tratamiento de agua residual (WWTP).
- RC : Conjunto de consumo residencial.
- CC : Conjunto de consumo comercial.
- PC : Conjunto de riego de parques urbanos.
- AC : Conjunto de riego agrícola.
- IC : Conjunto de consumo industrial.
- D : Conjunto de nodos de distribución.
- C : Conjunto de nodos de recolección.
- K : Conjunto de cursos de descarga natural.
- S : Conjunto del sumidero.

F.1.1.1. Subconjuntos

- W^G : Fuentes de agua subterránea.
- W^S : Fuentes de agua superficiales.
- DL : DWTPs existentes y de gran capacidad.
- DS : DWTPs existentes y de baja capacidad.
- WL : WWTPs existentes y de gran capacidad.
- WS : WWTPs existentes y de baja capacidad.
- PD : Riego de parques urbanos con consumo de agua potable.
- PF : Riego de parques urbanos con consumo de agua dulce.
- IL : Consumidores industriales de gran escala.
- IB : Consumidores industriales por distrito.
- EL : IWTP conectadas con consumidores industriales de gran escala.
- EB : IWTP conectadas con consumidores industriales por distrito.
- NL : DWTPs nuevas y de gran escala.
- NS : DWTPs nuevas y de pequeña escala.
- MA : WWTPs modificadas con calidad de salida de agua potable.
- MB : WWTPs modificadas con calidad de salida de agua dulce.
- MC : WWTPs modificadas con calidad de salida de agua de riego.
- NA : WWTPs nuevas con calidad de salida de agua potable.
- NB : WWTPs nuevas con calidad de salida de agua dulce.
- NC : WWTPs nuevas con calidad de salida de agua de riego.
- ND : WWTPs nuevas con calidad de salida de agua de descarga.
- K^G : Cursos de descarga de aguas subterráneas naturales.
- K^S : Cursos de descarga de aguas superficiales naturales.

F.1.1.1.1. Sub-subconjuntos

- DG : DWTPs existentes y de pequeña escala con consumo de agua subterránea.
- DS : DWTPs existentes y de pequeña escala con consumo de agua superficial.
- IL^d : Consumidores industriales de agua potable de gran escala.

- IL^f : Consumidores industriales de agua dulce de gran escala.
- EL^d : IWTP relacionada a IL^d .
- EL^f : IWTP relacionada a IL^f .
- MA^l : WWTPs modificadas y de gran escala con calidad de salida de agua potable.
- MB^l : WWTPs modificadas y de gran escala con calidad de salida de agua dulce.
- MC^l : WWTPs modificadas y de gran escala con calidad de salida de agua de riego.
- MA^s : WWTPs modificadas y de pequeña escala con calidad de salida de agua potable.
- MB^s : WWTPs modificadas y de pequeña escala con calidad de salida de agua dulce.
- MC^s : WWTPs modificadas y de pequeña escala con calidad de salida de agua de riego.
- NA^l : WWTPs nuevas y de gran escala con calidad de salida de agua potable.
- NB^l : WWTPs nuevas y de gran escala con calidad de salida de agua dulce.
- NC^l : WWTPs nuevas y de gran escala con calidad de salida de agua de riego.
- ND^l : WWTPs nuevas y de gran escala con calidad de salida de agua de descarga.
- NA^s : WWTPs nuevas y de pequeña escala con calidad de salida de agua potable.
- NB^s : WWTPs nuevas y de pequeña escala con calidad de salida de agua dulce.
- NC^s : WWTPs nuevas y de pequeña escala con calidad de salida de agua de riego.
- ND^s : WWTPs nuevas y de pequeña escala con calidad de salida de agua de descarga.

F.1.1.1.2. Sub-sub-subconjuntos

- IL^{ds} : Industrias de gran escala que consumen agua con calidad potable y descargan con calidad de agua residual al alcantarillado.
- IL^{dn} : Industrias de gran escala con consumo de agua con calidad potable y descarga a los cursos naturales de agua.
- IL^{fs} : Industrias de gran escala con consumo de agua dulce y descarga de agua residual al alcantarillado.
- IL^{fn} : Industrias de gran escala con consumo de agua dulce y descarga de agua a los cursos naturales de agua.
- EL^{ds} : IWTP relacionada a IL^{ds} .
- EL^{dn} : IWTP relacionada a IL^{dn} .
- EL^{fs} : IWTP relacionada a IL^{fs} .
- EL^{fn} : IWTP relacionada a IL^{fn} .

F.1.1.2. Otros tipos de conjuntos no mencionados previamente

- ON_j : Nodos de origen para cada balance en el nodo j .
- DN_j : Nodos de destino para cada balance de masa en el nodo j .
- AS_j : Todos los conjuntos excepto las fuentes de agua, cursos naturales de descarga o sumidero.
- CFW : Conjuntos de consumo de agua dulce ($\{ED \cup ND0 \cup PC \cup AC \cup IC\}$).
- WT : Conjuntos que poseen pérdidas de agua ($\{ED \cup EW \cup IE \cup ND0 \cup MW \cup NW \cup RC \cup CC \cup PC \cup AC \cup IC \cup D \cup C\}$).
- DNC : Conjuntos que poseen descarga de agua en fuentes naturales ($\{EW \cup EL^{dn} \cup EL^{fn} \cup ND\}$).
- AD_j : Conjunto que incluye los nodos de consumo ($\{RC \cup CC \cup PC \cup AC \cup IC\}$).
- CT : Conjunto que representa cada distrito.

F.1.1.3. Sub-índices generales

- w : Nodo de fuente de agua asociado al conjunto W .
- d : Nodo de distribución asociado al conjunto D .
- c : Nodo de recolección asociado al conjunto C .
- k : Nodo de descarga a cursos naturales relacionado al conjunto K .
- i_1 : Nodo de consumo residencial asociado al conjunto RC .
- i_2 : Nodo de consumo comercial relacionado con el conjunto CC .
- h_1 : Nodo de riego agrícola asociado al conjunto AC .
- u_1 : Nodo de riego de parques urbanos con consumo de agua potable, asociado al subconjunto PD .
- u_2 : Nodo de riego de parques urbanos con consumo de agua dulce, relacionado con el subconjunto PF .
- f_1 : Nodo de consumo industrial de gran escala que requiere agua con calidad potable y descarga con calidad el agua residual al alcantarillado, asociado al sub-sub-subconjunto IL^{ds} .
- f_2 : Nodo de consumo industrial de gran escala que requiere agua con calidad potable y descarga a los cursos naturales, asociado al sub-sub-subconjunto IL^{dn} .
- f_3 : Nodo de consumo industrial de gran escala que requiere agua dulce y descarga con calidad el agua residual al alcantarillado, asociado al sub-sub-subconjunto IL^{fs} .
- f_4 : Nodo de consumo industrial de gran escala que requiere agua dulce y descarga a los cursos naturales, asociado al sub-sub-subconjunto IL^{fn} .

- f_5 : Nodo de consumo industrial por distrito, relacionado con el subconjunto IB .
- k_0 : Nodo de descarga a cursos naturales de agua subterránea, relacionado con el subconjunto K^G .
- k_1 : Nodo de descarga a cursos de agua superficial, asociado al subconjunto K^S .

F.1.1.4. Sub-índices de plantas de tratamiento

- a_1 : Nodo de DWTP existente y de gran capacidad, asociado al subconjunto DL .
- a_2 : Nodo de DWTP existente y de baja capacidad, relacionado al subconjunto DS .
- b_1 : Nodo de WWTP existente y de gran capacidad, asociado al subconjunto WL .
- b_2 : Nodo de WWTP existente y de baja capacidad, relacionado al subconjunto WS .
- g_1 : Nodo de IWTP asociado al sub-sub-subconjunto EL^{ds} .
- g_2 : Nodo de IWTP relacionado al sub-sub-subconjunto EL^{dn} .
- g_3 : Nodo de IWTP asociado al sub-sub-subconjunto EL^{fs} .
- g_4 : Nodo de IWTP asociado al sub-sub-subconjunto EL^{fn} .
- g_5 : Nodo de IWTP asociado al sub-sub-subconjunto EB .
- e_1 : Nodo de DWTP nueva y de gran escala, relacionado al subconjunto NL .
- e_2 : Nodo de DWTP nueva y de pequeña escala, asociado al subconjunto NS .
- l_1 : Nodo de WWTP modificada y de gran escala con calidad de salida de agua potable, relacionado al sub-subconjunto MA^l .
- l_2 : Nodo de WWTP modificada y de pequeña escala con calidad de salida de agua potable, relacionado al sub-subconjunto MA^s .
- m_1 : Nodo de WWTP modificada y de gran escala con calidad de salida de agua dulce, asociado al sub-subconjunto MB^l .
- m_2 : Nodo de WWTP modificada y de pequeña escala con calidad de salida de agua dulce, relacionado al sub-subconjunto MB^s .
- n_1 : Nodo de WWTP modificada y de gran escala con calidad de salida de agua de riego, asociado al sub-subconjunto MC^l .
- n_2 : Nodo de WWTP modificada y de pequeña escala con calidad de salida de agua de riego, relacionado al sub-subconjunto MC^s .
- p_1 : Nodo de WWTP nueva y de gran escala con calidad de salida de agua potable, relacionado al sub-subconjunto NA^l .
- p_2 : Nodo de WWTP nueva y de pequeña escala con calidad de salida de agua potable, asociado al sub-subconjunto NA^s .

- q_1 : Nodo de WWTP nueva y de gran escala con calidad de salida de agua dulce, relacionado al sub-subconjunto NB^l .
- q_2 : Nodo de WWTP nueva y de pequeña escala con calidad de salida de agua dulce, asociado al sub-subconjunto NB^s .
- r_1 : Nodo de WWTP nueva y de gran escala con calidad de salida de agua de riego, relacionado al sub-subconjunto NC^l .
- r_2 : Nodo de WWTP nueva y de pequeña escala con calidad de salida de agua de riego, asociado al sub-subconjunto NC^s .
- t_1 : Nodo de WWTP nueva y de gran escala con calidad de salida de agua de descarga, relacionado al sub-subconjunto ND^l .
- t_2 : Nodo de WWTP nueva y de pequeña escala con calidad de salida de agua de descarga, asociado al sub-subconjunto ND^s .

F.1.2. Parámetros

- AF_i : Flujo actual de la planta existente i .
- C_i : Potencial de calentamiento global (GWP) de la planta i por unidad de volumen.
- Cfr_i : Costo fijo de la planta i .
- Cvr_i : Costo de inversión de la planta i .
- D : Diámetro de la tubería.
- DM_k : Demanda de agua del nodo k .
- $dist_{i,j}$: Distancia del nodo i al nodo j .
- E : Costo de electricidad.
- f : Coeficiente de Darcy.
- g : Aceleración de gravedad.
- h_i : Altura del nodo i .
- K_f : Factor para anualizar el costo de inversión.
- M : Parámetro incluido como cota superior en las restricciones del tipo *Big M*.
- m : Análogamente a M , m se emplea como cota inferior u holgura con un valor pequeño.
- T : Tiempo de operación en un año.
- Ter_i : Costo del terreno de la planta i .
- v : Velocidad promedio de los flujos de agua.
- w_i : Importancia o peso relativo de la función objetivo i .

Considerando las dimensiones del problema y sus múltiples parámetros, se destacarán en la sección F.2 aquellos que son analizados en el presente trabajo. Los valores del resto de parámetros pueden ser consultados en la publicación de Gormaz-Cuevas et al. [75].

F.1.3. Variables

- $F_{i,j}$: Flujo de agua del nodo i al nodo j .
- E_i : Existencia del nodo i .
- γ : Variable a ser minimizada en el método de *programación por metas*.

F.1.4. Restricciones

Las restricciones consideradas en este problema son las siguientes [75]:

Balances de masa

Se asume que los flujos de salida de las plantas de tratamiento poseen la calidad requerida por el consumidor al que se suministrará. A partir de ello, solo se requieren los balances de masa globales. Adicionalmente, se asume estado estacionario en cada nodo, por lo que la suma de flujos de la entrada es igual a la salida. Considerando también densidad constante en los flujos, es posible plantear balances de flujos volumétricos.

- Balance global:

Se formula que la suma de los flujos de fuentes de agua dulce superficial y subterránea son iguales a los de descarga destinados al ambiente más las pérdidas en cada nodo que van al sumidero:

$$\sum_{w \in W} \sum_{i \in CFW} F_{w,i} = \sum_{j \in DNC} \sum_{k \in K} F_{j,k} + \sum_{h \in WT} \sum_{s \in S} F_{h,s} \quad (F.1)$$

- Balance de masa por nodo:

Para aquellos nodos que poseen entradas y salidas, la expresión general del balance en cada uno de estos es el siguiente:

$$\sum_{i \in ON_j} F_{i,j} = \sum_{h \in DN_j} F_{j,h} + \sum_{s \in S} F_{j,s} \quad \forall j \in AS_j \quad (F.2)$$

Considerando las características de cada uno de los nodos de la red, se especificarán a continuación los balances en cada caso.

Tanto el consumo comercial como el residencial pueden ser suplidos por nuevas y existentes DWTPs, nuevas y modificadas WWTPs pequeñas con calidad de salida de agua potable y por nodos de distribución. En cuanto a los flujos de salida de estos nodos, pueden ser enviados a nodos de recolección, sumidero y a existentes, nuevas o modificadas WWTPs pequeñas.

$$\begin{aligned} & \sum_{a_2 \in DS} F_{a_2,j} + \sum_{e_2 \in NS} F_{e_2,j} + \sum_{l_2 \in MA^s} F_{l_2,j} + \sum_{p_2 \in NA^s} F_{p_2,j} + \sum_{d \in D} F_{d,j} \\ = & \sum_{c \in C} F_{j,c} + \sum_{b_2 \in WS} F_{j,b_2} + \sum_{p_2 \in NA^s} F_{j,p_2} + \sum_{q_2 \in NB^s} F_{j,q_2} + \sum_{r_2 \in NC^s} F_{j,r_2} + \sum_{t_2 \in ND^s} F_{j,t_2} \quad (F.3) \\ & + \sum_{l_2 \in MA^s} F_{j,l_2} + \sum_{m_2 \in MB^s} F_{j,m_2} + \sum_{n_2 \in MC^s} F_{j,n_2} + \sum_{s \in S} F_{j,s} \quad \forall j \in \{RC \cup CC\} \end{aligned}$$

Respecto a los nodos de consumidores industriales, estos puede recibir agua potable y dulce. Además, cada uno tiene asociado otro que representa una planta de tratamiento de efluentes (IWTP), con la que pueden descartar directamente a cursos naturales de agua o al alcantarillado. Existen dos tipos de consumidores industriales, clasificados según su tamaño, cuyos balances de masa se describen a continuación:

Los consumidores de menor escala o locales pueden ser suplidos con agua potable desde nuevas y existentes DWTPs, nuevas WWTPs grandes y pequeñas con calidad de salida de agua potable, y desde nodos de distribución. En cuanto a sus flujos de salida, estos pueden ser enviados a las IWTPs y al sumidero, tal como se observa en la ecuación F.4:

$$\sum_{a_2 \in DS} F_{a_2,j} + \sum_{e_2 \in NS} F_{e_2,j} + \sum_{l_2 \in MA^s} F_{l_2,j} + \sum_{p_2 \in NA^s} F_{p_2,j} + \sum_{d \in D} F_{d,j} = \sum_{g_5 \in EB} F_{j,g_5} + \sum_{s \in S} F_{j,s} \quad \forall j \in IB \quad (F.4)$$

Los consumidores industriales de gran escala pueden ser clasificados según la calidad de agua de entrada y salida: consumo de agua dulce y descarga a cursos naturales; consumo de agua dulce y descarga al alcantarillado; consumo de agua potable y descarga a cursos naturales; y consumo de agua potable y descarga al alcantarillado. A continuación, se muestran los casos en que el flujo se suple con agua potable y se descarga a cursos naturales (ecuación F.5) y el balance para la situación en que la entrada posee calidad de agua dulce y descarga al alcantarillado (ecuación F.6):

$$\sum_{d \in D} F_{d,j} + \sum_{a_2 \in DS} F_{a_2,j} + \sum_{e_2 \in NS} F_{e_2,j} + \sum_{p_2 \in NA^s} F_{p_2,j} + \sum_{l_2 \in MA^s} F_{l_2,j} = \sum_{g_2 \in EL^{dn}} F_{j,g_2} + \sum_{s \in S} F_{j,s} \quad \forall j \in IL^{dn} \quad (F.5)$$

$$\sum_{w \in W} F_{w,j} + \sum_{q_1 \in NB^l} F_{q_1,j} + \sum_{q_2 \in NB^s} F_{q_2,j} + \sum_{m_1 \in MB^l} F_{m_1,j} + \sum_{m_2 \in MB^s} F_{m_2,j} = \sum_{g_3 \in EL^{fs}} F_{j,g_3} + \sum_{s \in S} F_{j,s} \quad \forall j \in IL^{fs} \quad (F.6)$$

El consumo para riego agrícola puede ser cubierto desde fuentes naturales o WWTPs con flujos de salida con calidad de agua de riego, ya sean grandes o pequeñas, nuevas o modificadas. En cuanto a la descarga, esta puede ser a cursos subterráneos y al sumidero.

$$\sum_{w \in W} F_{w,j} + \sum_{r_1 \in NC^l} F_{r_1,j} + \sum_{r_2 \in NC^s} F_{r_2,j} + \sum_{n_1 \in MC^l} F_{n_1,j} + \sum_{n_2 \in MC^s} F_{n_2,j} = \sum_{k_0 \in KG} F_{j,k_0} + \sum_{s \in S} F_{j,s} \quad j \in AC \quad (F.7)$$

El riego de áreas urbanas puede clasificarse en dos subconjuntos según el origen del agua. Para el primer subconjunto se puede suministrar agua potable o dulce, mientras que el segundo solo con agua dulce. Para el primer caso, el balance de masa es el representado en la ecuación F.8, mientras que para el segundo se utiliza la expresión F.9:

$$\sum_{w \in W} F_{w,j} + \sum_{r_1 \in NC^l} F_{r_1,j} + \sum_{r_2 \in NC^s} F_{r_2,j} + \sum_{n_1 \in MC^l} F_{n_1,j} + \sum_{n_2 \in MC^s} F_{n_2,j} = \sum_{k_0 \in KG} F_{j,k_0} + \sum_{s \in S} F_{j,s} \quad \forall j \in PF \quad (F.8)$$

$$\begin{aligned}
& \sum_{a_2 \in DS} F_{a_2,j} + \sum_{d \in D} F_{d,j} + \sum_{r_1 \in NC^l} F_{r_1,j} + \sum_{r_2 \in NC^s} F_{r_2,j} + \sum_{n_1 \in MC^l} F_{n_1,j} + \sum_{n_2 \in MC^s} F_{n_2,j} \\
& = \sum_{k_0 \in K^G} F_{j,k_0} + \sum_{s \in S} F_{j,s} \quad \forall j \in PD
\end{aligned} \tag{F.9}$$

Respecto a los nodos de tratamiento, estos pueden dividirse en plantas existentes, modificadas o nuevas, presentándose WWTP o DWTP, las que pueden ser grandes o pequeñas. Se ha establecido que las plantas de gran escala deben conectarse con nodos de distribución y recolección para facilitar el suministro en la red, mientras que las plantas de pequeña escala no se conectan con estos nodos.

En cuanto a las plantas existentes, el balance de masa de las DWTPs de gran escala se observa en la ecuación F.10, mientras que en la expresión F.11 se ilustra el balance de masa de las DWTPs de menor escala:

$$\begin{aligned}
& \sum_{w \in W} F_{w,j} + \sum_{q_1 \in NB^l} F_{q_1,j} + \sum_{q_2 \in NB^s} F_{q_2,j} + \sum_{m_1 \in MB^l} F_{m_1,j} + \sum_{m_2 \in MB^s} F_{m_2,j} = \sum_{d \in D} F_{j,d} + \sum_{s \in S} F_{j,s} \\
& \quad \forall j \in DL
\end{aligned} \tag{F.10}$$

$$\begin{aligned}
& \sum_{w \in W} F_{w,j} + \sum_{q_1 \in NB^l} F_{q_1,j} + \sum_{q_2 \in NB^s} F_{q_2,j} + \sum_{m_1 \in MB^l} F_{m_1,j} + \sum_{m_2 \in MB^s} F_{m_2,j} \\
& = \sum_{i_1 \in RC} F_{j,i_1} + \sum_{i_2 \in CC} F_{j,i_2} + \sum_{u_1 \in PD} F_{j,u_1} + \sum_{f_1 \in IL^{ds}} F_{j,f_1} \\
& + \sum_{f_2 \in IL^{dn}} F_{j,f_2} + \sum_{f_3 \in IB} F_{j,f_3} + \sum_{s \in S} F_{j,s} \quad \forall j \in DS
\end{aligned} \tag{F.11}$$

Para el caso de las WWTPs existentes, estas solamente pueden generar flujos de salida con calidad de descarga a cursos naturales. Los flujos de entrada a las WWTPs existentes y de gran escala provienen de nodos de recolección y las salidas se envían a los cursos naturales o al sumidero, cuyo balance de masa se encuentra en la ecuación F.12. Por otro lado, el balance de masa para las WWTPs existentes y de menor escala es similar al caso de gran escala, excepto porque los flujos se conectan directamente con los nodos de consumo y no a través de nodos de recolección o distribución.

$$\sum_{c \in C} F_{c,j} = \sum_{k_1 \in K^s} F_{j,k_1} + \sum_{s \in S} F_{j,s} \quad \forall j \in WL \tag{F.12}$$

Respecto a las nuevas plantas de tratamiento, estas pueden ser instaladas en ubicaciones diferentes a las existentes. Esta diferencia es la principal entre las DWTPs nuevas y existentes, razón por la que el balance de masa se formula de la misma forma en ambos casos. Por otro lado, las nuevas WWTPs pueden presentar cuatro calidades de salida diferentes: agua potable, dulce, de riego y con calidad de descarga a fuente natural, cuyos balances de masa para las plantas de gran escala se representan en las ecuaciones F.13, F.14, F.15 y F.16, respectivamente. Las WWTPs nuevas y de pequeña escala presentan la misma diferencia con las de gran tamaño que la descrita para las existentes.

$$\sum_{c \in C} F_{c,j} = \sum_{d \in D} F_{j,d} + \sum_{s \in S} F_{j,s} \quad \forall j \in NA^l \quad (\text{F.13})$$

$$\begin{aligned} \sum_{c \in C} F_{c,j} = & \sum_{a_1 \in DL} F_{j,a_1} + \sum_{a_2 \in DS} F_{j,a_2} + \sum_{f_3 \in IL^{fs}} F_{j,f_3} + \\ & \sum_{f_4 \in IL^{fn}} F_{j,f_4} + \sum_{s \in S} F_{j,s} \quad \forall j \in NB^l \end{aligned} \quad (\text{F.14})$$

$$\sum_{c \in C} F_{c,j} = \sum_{h_1 \in AC} F_{j,h_1} + \sum_{u_1 \in PD} F_{j,u_1} + \sum_{u_2 \in PF} F_{j,u_2} + \sum_{s \in S} F_{j,s} \quad \forall j \in NC^l \quad (\text{F.15})$$

$$\sum_{c \in C} F_{c,j} = \sum_{k_1 \in K^s} F_{j,k_1} + \sum_{s \in S} F_{j,s} \quad \forall j \in ND^l \quad (\text{F.16})$$

En el caso de las plantas de tratamiento modificadas, estas pueden ser adaptadas para que sus flujos de salida presenten diferentes calidades: agua potable, agua dulce y de riego. Los balances de masa correspondientes son iguales a los de las plantas nuevas. Particularmente, la calidad de salida de las plantas de tratamiento de agua industrial estará condicionada por la planta de consumo industrial a la que está conectada.

Los nodos de distribución y recolección permiten representar las pérdidas de agua de las plantas de agua de gran escala. Los flujos que ingresan y salen del nodo de distribución se observan en la ecuación F.17, mientras que el balance de masa del nodo de recolección se observa en la ecuación F.18.

$$\begin{aligned} & \sum_{a_1 \in DL} F_{a_1,j} + \sum_{e_1 \in NL} F_{e_1,j} + \sum_{p_1 \in NA^l} F_{p_1,j} + \sum_{l_1 \in MA^l} F_{l_1,j} = \\ & \sum_{i_1 \in RC} F_{j,i_1} + \sum_{i_2 \in CC} F_{j,i_2} + \sum_{u_1 \in PD} F_{j,u_1} + \sum_{f_1 \in IL^{ds}} F_{j,f_1} \\ & + \sum_{f_2 \in IL^{dn}} F_{j,f_2} + \sum_{f_5 \in IB} F_{j,f_5} + \sum_{s \in S} F_{j,s} \quad \forall j \in D \end{aligned} \quad (\text{F.17})$$

$$\begin{aligned} & \sum_{i_1 \in RC} F_{i_1,j} + \sum_{i_2 \in CC} F_{i_2,j} + \sum_{g_1 \in EL^{ds}} F_{g_1,j} + \sum_{g_3 \in EL^{fs}} F_{g_3,j} + \sum_{g_5 \in EB} F_{g_5,j} = \\ & \sum_{b_1 \in WL} F_{j,b_1} + \sum_{p_1 \in NA^l} F_{j,p_1} + \sum_{q_1 \in NB^l} F_{j,q_1} + \sum_{r_1 \in NC^l} F_{j,r_1} + \sum_{t_1 \in ND^l} F_{j,t_1} \\ & + \sum_{l_1 \in MA^l} F_{j,l_1} + \sum_{m_1 \in MB^l} F_{j,m_1} + \sum_{n_1 \in MC^l} F_{j,n_1} + \sum_{s \in S} F_{j,s} \quad \forall j \in C \end{aligned} \quad (\text{F.18})$$

Suplir la demanda

Cada nodo de consumo tiene asociada una demanda de agua. Esta demanda es diferente según la ubicación y el tipo de consumo. La restricción que permite asegurar el cumplimiento de la demanda se expresa en la ecuación F.19:

$$\sum_{i \in ON_j} F_{i,j} \geq DM_{\{j,p\}} \quad \forall j \in AD_j, \quad \forall p \in CT \quad (\text{F.19})$$

Capacidad de plantas de tratamiento

Una planta de tratamiento de un determinado distrito debe ser capaz de cubrir la demanda de los usuarios en aquel distrito.

- Capacidad de plantas nuevas de tratamiento de agua potable:

En relación a las plantas nuevas de tratamiento de agua potable de gran capacidad, la restricción F.20 relaciona los flujos que salen de la planta correspondiente y la demanda del distrito.

$$\sum_{d \in D} F_{i,d} \geq \sum_{j \in AD_j} DM_{\{j,p'\}} \cdot E_i \quad \forall i \in NL, \forall p' \in CT \mid p' \text{ es el distrito donde se ubica } i \quad (\text{F.20})$$

Para el caso de las plantas de pequeña escala, se flexibiliza la relación de flujos y demandas para su instalación. Considerando que en la realidad podría suplirse la demanda de consumidores cercanos a las plantas pese a que estén en otro distrito, se incluye el parámetro m para flexibilizar la formulación de la restricción F.21.

$$\sum_j F_{i,j} \leq \left(\sum_j DM_{\{j,p'\}} + m \right) \cdot E_i \quad \forall i \in NS, \forall p' \in CT \mid p' \text{ es el distrito donde se ubica } i \quad (\text{F.21})$$

- Capacidad de plantas nuevas de tratamiento de agua residual:

Para este tipo de plantas también se realiza una distinción entre las de gran y pequeña escala. Para el caso de gran escala, la restricción que debe cumplirse es la expresada en la ecuación F.22, mientras que para las de pequeña escala se utiliza la restricción F.23.

$$\sum_{c \in C} F_{c,i} \geq \sum_j DM_{\{j,p'\}} \cdot E_i \quad \forall i \in \{NA^l \cup NB^l \cup NC^l \cup ND^l\}, \forall p' \in CT \mid p' \text{ es el distrito donde se ubica } i \quad (\text{F.22})$$

$$\sum_j F_{j,i} \leq \left(\sum_j DM_{\{j,p'\}} + m \right) \cdot E_i \quad \forall i \in \{NA^s \cup NB^s \cup NC^s \cup ND^s\}, \forall p' \in CT \mid p' \text{ es el distrito donde se ubica } i \quad (\text{F.23})$$

- Capacidad de plantas de tratamiento existentes o modificadas:

En las plantas de tratamiento de agua residual los flujos varían según la estación del año, según lo descrito por Gormaz-Cuevas et al. [74]. Particularmente, existe un mayor consumo en verano que en invierno, por lo que proponen que las plantas existentes y modificadas solo estarán instaladas si el flujo tratado es mayor que la variación en invierno y menor que la de verano. Específicamente, cada planta puede tratar 25 % menos que la fluctuación de invierno, siendo un 15 % menos que el flujo actual. Estos valores se utilizan para expresar la restricción F.24 . Por otro lado, la planta puede tratar un 23 % más que el flujo actual, correspondiente a la variación de verano. Para la creación de la cota superior no se incluye algún factor que flexibilice el planteamiento,

representándose con la ecuación F.25.

$$\sum_{i \in ON_j} F_{i,j} \geq 0,75 \cdot 0,85 \cdot AF_j \cdot E_j \quad \forall j \in EW \quad (F.24)$$

$$\sum_{i \in ON_j} F_{i,j} \leq 1,23 \cdot AF_j \cdot E_j \quad \forall j \in EW \quad (F.25)$$

Respecto a las plantas de tratamiento de agua potable, el razonamiento y planteamiento es similar a lo expresado para las plantas anteriores, aunque se basan en este caso en los flujos de salida, según lo observado en las ecuaciones F.26 y F.27.

$$\sum_{i \in DN_j} F_{j,i} \geq 0,75 \cdot 0,85 \cdot AF_j \cdot E_j \quad \forall j \in ED \quad (F.26)$$

$$\sum_{i \in DN_j} F_{j,i} \leq 1,23 \cdot AF_j \cdot E_j \quad \forall j \in ED \quad (F.27)$$

Relaciones lógicas de existencia

Para que los flujos se dirijan a una planta, esta debe estar instalada. Estas relaciones se plantean utilizando el método de *BigM* en la ecuación F.28. Además, una planta existente y una modificada no se pueden encontrar en la misma ubicación, siendo representada esta restricción en la ecuación F.29.

$$\sum_{i \in ON_j} F_{i,j} \leq M \cdot E_j \quad \forall j \in \{NW \cup ND0\} \quad (F.28)$$

$$\sum_{j \in \{EW \cup MW\}} E_j \leq 1 \quad (F.29)$$

Costos

Los costos presentes en el problema se pueden clasificar en operacionales (*OpC*) y de inversión (*CapC*). Para los costos operacionales se considera el costo de transporte de agua, según se observa en la ecuación F.30. Por otro lado, los costos de inversión se asocian a la instalación o modificación de plantas existentes, planteándose en la ecuación F.31.

$$OpC = T \cdot E \cdot \sum_{j \in DN_j} \sum_{i \in ON_i} F_{i,j} \cdot \left(g \cdot (h_j - h_i) + f \cdot \frac{dist_{i,j} \cdot v^2}{2 \cdot D} \right) \quad (F.30)$$

$$CapC = K_f \cdot \sum_i E_i \cdot \left(Cvr_i \cdot \sum_{j \in ON_j} F_{j,i} + Cfr_i + Ter_i \right), \quad i \in \{ND0 \cup MW \cup NW\} \quad (F.31)$$

Potencial de calentamiento global

El planteamiento original de Gormaz-Cuevas et al. [74] consideraba como indicador ambiental solamente la extracción de agua desde la fuente. Para complementar el trabajo efectuado, Gormaz-Cuevas et al. [75] incluyeron un indicador ambiental adicional: Potencial de calentamiento global (GWP). Para medir el GWP total de la red (TWGP), se utiliza la expresión F.32, donde C_i representa el GWP de la planta i por unidad de volumen.

$$TWGP = \sum_{j \in AD_j} \sum_i C_i \cdot F_{i,j}, \quad i \in \{NW \cup MW \cup ED \cup ND0\} \quad (F.32)$$

F.1.5. Funciones objetivo

El problema de optimización considera la minimización de tres funciones objetivo opuestas: consumo de agua desde la fuente, costo total y el potencial de calentamiento global total (TGWP), siendo presentadas en las ecuaciones F.33, F.34 y F.35, respectivamente.

$$FO1 = G0 = \sum_{w \in W} \sum_{j \in CFW} F_{w,j} \quad (F.33)$$

$$FO2 = TC = (OpC + CapC) \quad (F.34)$$

$$FO3 = TGWP = \sum_{j \in AD_j} \sum_i C_i \cdot F_{i,j}, \quad i \in \{NW \cup MW \cup ED \cup ND0\} \quad (F.35)$$

El problema de optimización multiobjetivo se resuelve con el método de *programación por metas*. La comparación entre las tres funciones objetivo incluye una normalización empleando valores ideales y anti-ideales de cada función objetivo. El valor ideal (*id*) es la función objetivo minimizada en un problema mono-objetivo, mientras que el caso anti-ideal (*non-id*) se obtiene con la maximización individual de cada función objetivo. De esta forma, se analizarán como indicadores las expresiones F.36, F.37 y F.38. Para el caso particular del método de *programación por metas*, estas funciones objetivo son incluidas dentro de las restricciones F.39, F.40 y F.41, donde w_i representa la importancia relativa de cada indicador y γ es la variable a minimizar en la función objetivo observada en la expresión F.42.

$$FO1_{MO} = \frac{G0 - G0_{id}}{G0_{non-id} - G0_{id}} \quad (F.36)$$

$$FO2_{MO} = \frac{TC - TC_{id}}{TC_{non-id} - TC_{id}} \quad (F.37)$$

$$FO3_{MO} = \frac{TGWP - TGWP_{id}}{TGWP_{non-id} - TGWP_{id}} \quad (F.38)$$

$$w_1 \cdot FO1_{MO} \leq \gamma \quad (F.39)$$

$$w_2 \cdot FO2_{MO} \leq \gamma \quad (F.40)$$

$$w_3 \cdot FO3_{MO} \leq \gamma \quad (F.41)$$

$$\min \gamma \quad (F.42)$$

En la sección F.2 se analizará la formulación presentada y las simplificaciones y supuestos necesarios para la implementación del algoritmo propuesto en este problema. En particular, se discutirá sobre la forma de incorporar los valores involucrados en las funciones objetivo, representadas como restricciones en el método de *programación por metas*.

F.2. Análisis de la formulación

La resolución del problema propuesto originalmente por Gormaz-Cuevas et al. [75] permitió detectar características que influyen en el desempeño del algoritmo, que deben ser advertidas para su correcta implementación en este caso y en otros.

F.2.1. Características del problema

El problema presenta diferentes soluciones óptimas, generando que incluso un cambio inferior al 1 % en alguno de sus parámetros produzca una solución diferente. Adicionalmente, la solución cambia al utilizar diferentes adivinanzas iniciales para sus variables. Es por ello que no se podría implementar correctamente el algoritmo propuesto. Las características específicas de la formulación que generan las múltiples soluciones son las siguientes:

Alternativas similares

La formulación original del problema permite escoger entre alternativas que ofrecen soluciones similares. Específicamente, generan los mismos valores en las funciones objetivo para diferentes configuraciones. Para que ello no ocurra, deberían considerarse nuevos indicadores que permitan discriminar claramente entre las diferentes opciones o disminuir el número de opciones con características similares a evaluar.

Importancia relativa de las funciones objetivo en programación por metas

Un aspecto que ocasiona alternativas similares es la preponderancia de las funciones objetivo. En *programación por metas*, se minimiza una variable adicional (γ) con el propósito de disminuir la diferencia entre la solución ideal y la del problema multiobjetivo. Esto ocasiona que la o las funciones objetivo que se relacionen directamente con las variables binarias deban presentar una importancia relativa que permita que el método de *programación por metas* obtenga la mejor solución en relación a las decisiones de instalación. Específicamente, en el problema analizado la función objetivo económica podía adoptar diferentes valores, porque las restricciones adicionales F.39 y F.41 se encuentran activas, mientras que la expresión F.40 no. Con ello, el problema puede escoger diferentes configuraciones óptimas en la red, ya que la función económica no afecta a la minimización de γ .

Normalización de las funciones objetivo en programación por metas

Otra característica que presenta el método de *programación por metas* es la normalización de las funciones objetivo, con el propósito de comparar indicadores con diferentes unidades y órdenes de magnitud. Sin embargo, esto puede generar implícitamente una modificación en la importancia relativa de las funciones objetivo. Por ejemplo, un denominador mayor en las expresiones F.36, F.37 y F.38 genera una menor influencia en la minimización de γ . Específicamente, en el presente problema la diferencia entre los valores ideales y anti-ideales es cercana al 50 % para las funciones objetivo ambientales (extracción de agua y potencial de calentamiento global), mientras que para la función económica presentan diferentes órdenes de magnitud. Es por ello que las decisiones de instalación, relacionadas directamente en la formulación de la función económica, cambian incluso por variaciones inferiores al 1 % en los parámetros o tras resolver el problema con diferentes valores iniciales. Matemáticamente, se produce un *trade-off* entre los costos de operación e instalación, generándose soluciones óptimas diferentes que no influyen en la minimización de γ por el efecto de la normalización.

F.2.2. Simplificaciones para la implementación del algoritmo

El análisis de las situaciones anteriores muestra los casos que generan dificultades para la correcta implementación del algoritmo. Estas provienen de las decisiones realizadas en la formulación, por lo que son ajenas al alcance del algoritmo. Para implementar el algoritmo en el caso de estudio, teniendo presente las discusiones anteriores, se realizan las siguientes simplificaciones:

- Se analizan tres tipos de consumidores: residencial, industrial y agrícola. Debido a la calidad del agua que se requiere, se incorporó el consumo comercial al residencial y el de riego de parques al agrícola.
- En relación al consumo de plantas industriales grandes, la planta considerada en el problema propuesto por Gormaz-Cuevas et al. [75] se incluye dentro del consumo industrial del distrito correspondiente. Junto con ello, las plantas de tratamiento de agua residual industrial pueden recibir flujos provenientes de consumidores industriales de diferentes distritos.
- La ubicación de los consumidores agrícolas corresponde a la de los parques, ya que están presentes en todos los distritos, a diferencia de los consumidores agrícolas en el planteamiento original. Donde se encontraban previamente los consumos agrícola, se asume la mayor pérdida de agua entre ambos casos (66 %), mientras que en los casos en que originalmente solo existían parques, se mantiene la pérdida de agua (55 %).
- Las variables binarias se reducen, manteniéndose la evaluación de las plantas existentes y la posibilidad de instalar nuevas plantas de tratamiento con diferentes calidades en aquellos distritos donde no existe previamente una planta de tratamiento (*NW* y *SE*). Se elimina la opción de instalar plantas de tratamiento en los distritos que ya existen (*NE* y *SW*), privilegiando que se usen las que ya están instaladas si así se requiere. Adicionalmente, no se podrán modificar las ya instaladas y no se podrán instalar nuevas plantas con calidad de potable. Esto se debe a que las plantas nuevas pueden presentar aquella calidad si así se necesita como parte de la solución.
- Los diámetros de las tuberías se asumen iguales a 1 metro para cada conexión.
- Se utiliza $m = 0$, eliminando la holgura en las instalaciones.
- La importancia relativa de la función objetivo se divide de la siguiente manera: 0,5 para el indicador económico y 0,5 para los casos ambientales. De este último, 0,25 corresponde a la extracción de agua y 0,25 al potencial de calentamiento global.
- La normalización de la función objetivo económica considera un valor anti-ideal de 10 veces el ideal, buscando que el denominador influya en un orden de magnitud similar para todas las funciones objetivo.

A partir de los cambios anteriores, se disminuyen las alternativas similares y la función objetivo económica presenta una importancia igual a las otras funciones objetivo en conjunto, a diferencia del planteamiento original. Se propone como extensión a futuro que se analice en mayor detalle cómo se realizan las normalizaciones en el método de *programación por metas*, ya que la importancia relativa es influenciada por el denominador escogido.

F.2.3. Resumen del nuevo problema

El problema simplificado y que será utilizado como el caso de estudio principal de la tesis posee 719 variables continuas (de 1.097 originalmente), 20 variables binarias (50 en el caso original) y 410 ecuaciones (535 previamente). El número de distritos y sus demandas se mantienen, representando a la red del Gran Santiago. Las demandas de agua de los nodos agrupados se resumen en la Tabla F.1.

Tabla F.1: Consumo de agua por tipo (residencial (RC), agrícola (AC) e industrial (IC)) y distrito en el ejemplo ilustrativo 4.

Distrito	RC [m^3/s]	AC [m^3/s]	IC [m^3/s]
<i>NE</i>	1,30	2,23	0,09
<i>NW</i>	3,83	0,27	0,38
<i>SE</i>	3,08	6,17	1,18
<i>SW</i>	3,18	0,31	0,14

F.2.4. Parámetros de interés

Los parámetros que se analizarán mediante la implementación del algoritmo propuesto serán las demandas de agua, el parámetro M utilizado en la formulación de restricciones del tipo *Big M* y el costo de electricidad E . Las demandas y los costos corresponden a parámetros que son recurrentemente analizados por la incertidumbre de sus valores en el tiempo, tal como se ha destacado previamente. Adicionalmente, el problema original presentaba una sensibilidad incluso al parámetro M , por lo que la nueva versión también incorporará su análisis, y su valor nominal será de 1.000, ya que se requiere un valor suficientemente alto que no afecte las variaciones que se realizarán en los flujos. Considerando que el procedimiento sería análogo para los diferentes flujos, se estudiarán solamente las demandas de los distritos que presenten el mayor valor por tipo de consumo. De esta forma, los valores nominales de los parámetros a analizar se resumen en la Tabla F.2.

Tabla F.2: Valores nominales de los parámetros de interés del ejemplo ilustrativo 4. La demanda de agua ($DM_{k=\{j,p\}}$) se expresa incluyendo el tipo de consumo ($j = RC, AC, IC$) y el distrito ($p = NE, NW, SE, SW$).

Parámetro	Valor	Unidad
$DM_{\{RC,NW\}}$	3,83	[m^3/s]
$DM_{\{AC,SE\}}$	6,17	[m^3/s]
$DM_{\{IC,SE\}}$	1,18	[m^3/s]
E	0,158	[USD/kWh]
M	1.000	[—]

F.3. Solución óptima

La solución original del problema con las simplificaciones efectuadas se obtiene resolviéndolo con los valores nominales de los parámetros. Las variables de interés son las binarias, cuyos valores óptimos se observan en la Tabla F.3.

Tabla F.3: Instalaciones óptimas del ejemplo 4 con los valores nominales de los parámetros. La existencia del nodo ($E_{i=\{j,p\}}$) se expresa incluyendo el tipo de planta ($j = DG, DS, NA^l, NB^l, NC^l, ND^l, NA^s, NB^s, NC^s, ND^s, WL, WS$) y el distrito en que se encuentra ($p = NE, NW, SE, SW$).

Variable	Valor	Unidad	Variable	Valor	Unidad
$E_{\{NA^l, NW\}}$	-	$[-]$	$E_{\{NA^l, SE\}}$	-	$[-]$
$E_{\{NA^s, NW\}}$	1	$[-]$	$E_{\{NA^s, SE\}}$	1	$[-]$
$E_{\{NB^l, NW\}}$	-	$[-]$	$E_{\{NB^l, SE\}}$	-	$[-]$
$E_{\{NB^s, NW\}}$	-	$[-]$	$E_{\{NB^s, SE\}}$	-	$[-]$
$E_{\{NC^l, NW\}}$	-	$[-]$	$E_{\{NC^l, SE\}}$	-	$[-]$
$E_{\{NC^s, NW\}}$	-	$[-]$	$E_{\{NC^s, SE\}}$	-	$[-]$
$E_{\{ND^l, NW\}}$	-	$[-]$	$E_{\{ND^l, SE\}}$	-	$[-]$
$E_{\{ND^s, NW\}}$	-	$[-]$	$E_{\{ND^s, SE\}}$	-	$[-]$
$E_{\{DS, SE\}}$	-	$[-]$	$E_{\{DG, NW\}}$	1	$[-]$
$E_{\{WL, SW\}}$	-	$[-]$	$E_{\{WS, NE\}}$	-	$[-]$

La solución óptima incluye la instalación de tres plantas: dos nuevas y una de las existentes. Las tres plantas tienen calidad de salida de agua potable y son de pequeña escala, pudiendo conectarse directamente con los nodos de consumo.

El tiempo de ejecución de este problema fue de 71,55 [s], cuyo valor será utilizado como referencia para acotar el tiempo de resolución de las iteraciones del algoritmo propuesto. Existirán iteraciones que superen el límite establecido dentro del mismo orden de magnitud, debido a que en el software GAMS permite resolver algunos casos durante un tiempo adicional, ya que el límite es aproximado. El tiempo de cada iteración será reportado para explicitar aquella situación, que afecta al tiempo total de implementación del algoritmo.

F.4. Optimización de parámetros

Para cada parámetro en estudio se plantea un subproblema con el propósito de estimar una nueva adivinanza inicial, según el procedimiento explicado en el Capítulo 5. La solución de la minimización y maximización de cada subproblema se observa en la Tabla F.4, junto con el tiempo utilizado en la resolución y en la evaluación en el caso original, además de la factibilidad de uso como punto inicial en el algoritmo propuesto.

Tabla F.4: Parámetros optimizados del ejemplo ilustrativo 4.

Parámetro optimizado [UN]	Valor obtenido	Tiempo ejec. reformulado [s]	Tiempo de evaluación [s]	¿Mantiene binaria?
$DM_{\{RC,NW\},op}^{min} [m^3/s]$	0,0	3,77	218,41	No
$DM_{\{RC,NW\},op}^{max} [m^3/s]$	9,084	1,39	81,69	No
$DM_{\{AC,SE\},op}^{min} [m^3/s]$	2,956	1,12	5,93	Sí
$DM_{\{AC,SE\},op}^{max} [m^3/s]$	7.497.067.746	457,58	21,26	No
$DM_{\{IC,SE\},op}^{min} [m^3/s]$	0,0	1,14	373,38	No
$DM_{\{IC,SE\},op}^{max} [m^3/s]$	5,336	1,83	361,05	No
$E_{op}^{min} [USD/kWh]$	0,0	3,42	8,80	No
$E_{op}^{max} [USD/kWh]$	100.000.000	23,74	-	No ^a
$M_{op}^{min} [-]$	7,216	1,69	8,59	Sí
$M_{op}^{max} [-]$	10^{50}	91,45	129,14	Sí

^a La evaluación del parámetro optimizado en el problema original no permitió obtener una solución tras reportarse un error en GAMS.

Los resultados mostrados en la Tabla F.4 permiten concluir que una nueva estimación se puede utilizar para encontrar tres límites: inferior y superior de M , e inferior de $DM_{\{AC,SE\}}$. Particularmente, el límite superior de M no está acotado, ya que se alcanza el máximo valor posible en GAMS. Para los otros casos se descartan las estimaciones efectuadas, iniciándose el proceso iterativo desde su valor nominal.

F.5. Rango de estabilidad (algoritmo propuesto)

F.5.1. Parámetro $DM_{\{RC,NW\}}$

La búsqueda del límite inferior de $DM_{\{RC,NW\}}$ inicia desde el valor nominal y finaliza en el valor 3,11 tras alcanzar la tolerancia establecida, según se observa en la Tabla F.5. Respecto al límite superior de $DM_{\{RC,NW\}}$, se realizaron las iteraciones mostradas en la Tabla F.6, comenzando desde el valor nominal y finalizando en 4,55 por el criterio de tolerancia.

Tabla F.5: Determinación del límite inferior del rango de estabilidad del parámetro $DM_{\{RC,NW\}}$ mediante el algoritmo propuesto.

i	$DM_{\{RC,NW\}}$	$y' = y_0?$	$DM_{\{RC,NW\}}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	3,83	Sí	–	–	91,92
1	2,83	No	3,83	$0,1 > 0,0015$	366,04
Ciclo 2: $\Delta l = 0,1$					
0	3,83	Sí	–	–	56,71
1	3,73	Sí	–	–	4,50
2	3,63	Sí	–	–	6,01
3	3,53	Sí	–	–	4,89
4	3,43	Sí	–	–	4,33
5	3,33	Sí	–	–	5,01
6	3,23	Sí	–	–	4,42
7	3,13	Sí	–	–	3,72
8	3,03	No	3,13	$0,01 > 0,0015$	364,72
Ciclo 3: $\Delta l = 0,01$					
0	3,13	Sí	–	–	3,15
1	3,12	Sí	–	–	378,50
2	3,11	Sí	–	–	385,16
3	3,10	No	3,11	$0,001 < 0,0015$	374,30
Tiempo iteraciones: 2.053,38[s]					
Tiempo total (iteraciones + opt. de parámetros): 2.275,56[s]					

Tabla F.6: Determinación del límite superior del rango de estabilidad del parámetro $DM_{\{RC,NW\}}$ mediante el algoritmo propuesto.

i	$DM_{\{RC,NW\}}$	$y' = y_0?$	$DM_{\{RC,NW\}}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	3,83	Sí	—	—	66,97
1	4,83	No	3,83	$0,1 > 0,0015$	365,99
Ciclo 2: $\Delta l = 0,1$					
0	3,83	Sí	—	—	57,45
1	3,93	Sí	—	—	3,83
2	4,03	Sí	—	—	3,77
3	4,13	Sí	—	—	49,97
4	4,23	Sí	—	—	2,12
5	4,33	Sí	—	—	74,00
6	4,43	Sí	—	—	4,99
7	4,53	Sí	—	—	235,76
8	4,63	No	4,53	$0,01 > 0,0015$	361,84
Ciclo 3: $\Delta l = 0,01$					
0	4,53	Sí	—	—	229,18
1	4,54	Sí	—	—	18,12
2	4,55	Sí	—	—	107,54
3	4,56	No	4,55	$0,001 < 0,0015$	365,69
Tiempo iteraciones: 1.947,22[s]					
Tiempo total (iteraciones + opt. de parámetros): 2.030,30[s]					

F.5.2. Parámetro $DM_{\{AC,SE\}}$

La *optimización de parámetros* realizada permitió determinar un nuevo valor inicial para las iteraciones efectuadas en la búsqueda del límite inferior de $DM_{\{AC,SE\}}$. Particularmente, el valor obtenido corresponde al límite buscado, siendo necesarias las iteraciones de la Tabla F.7 para corroborar esta situación y finalizar tras disminuir el paso hasta un valor inferior a la tolerancia. En contraste, la determinación del límite superior de $DM_{\{AC,SE\}}$ comienza en su valor nominal y termina en 8,66 por cumplir con la tolerancia establecida, según se resume en la Tabla F.8.

Tabla F.7: Determinación del límite inferior del rango de estabilidad del parámetro $DM_{\{AC,SE\}}$ mediante el algoritmo propuesto.

i	$DM_{\{AC,SE\}}$	$y' = y_0?$	$DM_{\{AC,SE\}}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	2,956	Sí	–	–	5,23
1	1,956	No	2,956	$0,1 > 0,0025$	364,30
Ciclo 2: $\Delta l = 0,1$					
0	2,956	Sí	–	–	4,49
1	2,856	No	2,956	$0,01 > 0,0025$	29,47
Ciclo 3: $\Delta l = 0,01$					
0	2,956	Sí	–	–	4,65
1	2,946	No	2,956	$0,001 < 0,0025$	81,57
Tiempo iteraciones: 489,71[s]					
Tiempo total (iteraciones + opt. de parámetros): 496,76[s]					

Tabla F.8: Determinación del límite superior del rango de estabilidad del parámetro $DM_{\{AC,SE\}}$ mediante el algoritmo propuesto.

i	$DM_{\{AC,SE\}}$	$y' = y_0?$	$DM_{\{AC,SE\}}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	6,17	Sí	—	—	64,21
1	7,17	Sí	—	—	3,05
2	8,17	Sí	—	—	3,56
3	9,17	No	8,17	$0,1 > 0,0025$	34,06
Ciclo 2: $\Delta l = 0,1$					
0	8,17	Sí	—	—	3,34
1	8,27	Sí	—	—	2,88
2	8,37	Sí	—	—	3,23
3	8,47	Sí	—	—	3,09
4	8,57	Sí	—	—	5,14
5	8,67	No	8,57	$0,01 > 0,0025$	4,66
Ciclo 3: $\Delta l = 0,01$					
0	8,57	Sí	—	—	3,53
1	8,58	Sí	—	—	4,05
2	8,59	Sí	—	—	3,82
3	8,60	Sí	—	—	16,71
4	8,61	Sí	—	—	3,12
5	8,62	Sí	—	—	3,77
6	8,63	Sí	—	—	61,17
7	8,64	Sí	—	—	3,43
8	8,65	Sí	—	—	4,92
9	8,66	Sí	—	—	33,56
10	8,67	No	8,66	$0,001 < 0,0025$	3,30
Tiempo iteraciones: 268,60[s]					
Tiempo total (iteraciones + opt. de parámetros): 747,44[s]					

F.5.3. Parámetro $DM_{\{IC,SE\}}$

La determinación del límite inferior de $DM_{\{IC,SE\}}$ comienza en su valor nominal y finaliza en 0,434 tras alcanzar la tolerancia, según se observa en la Tabla F.9. En relación al límite superior de $DM_{\{IC,SE\}}$, su búsqueda comienza en el valor nominal y termina en 1,460 por no ser posible continuar disminuyendo el paso evaluado, tal como se aprecia en la Tabla F.10.

Tabla F.9: Determinación del límite inferior del rango de estabilidad del parámetro $DM_{\{IC,SE\}}$ mediante el algoritmo propuesto.

i	$DM_{\{IC,SE\}}$	$y' = y_0?$	$DM_{\{IC,SE\}}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,18	Sí	–	–	68,10
1	0,18	No	1,18	$0,1 > 0,0005$	386,60
Ciclo 2: $\Delta l = 0,1$					
0	1,18	Sí	–	–	78,19
1	1,08	Sí	–	–	58,64
2	0,98	Sí	–	–	7,94
3	0,88	Sí	–	–	81,59
4	0,78	Sí	–	–	5,02
5	0,68	Sí	–	–	8,08
6	0,58	Sí	–	–	16,48
7	0,48	Sí	–	–	8,95
8	0,38	No	0,48	$0,01 > 0,0005$	365,76
Ciclo 3: $\Delta l = 0,01$					
0	0,48	Sí	–	–	8,47
1	0,47	Sí	–	–	4,33
2	0,46	Sí	–	–	7,39
3	0,45	Sí	–	–	369,65
4	0,44	Sí	–	–	374,78
5	0,43	No	0,44	$0,001 > 0,0005$	370,47
Ciclo 4: $\Delta l = 0,001$					
0	0,440	Sí	–	–	384,95
1	0,439	Sí	–	–	377,97
2	0,438	Sí	–	–	405,33
3	0,437	Sí	–	–	370,53
4	0,436	Sí	–	–	370,78
5	0,435	Sí	–	–	368,55
6	0,434	Sí	–	–	369,57
7	0,433	No	0,434	$0,0001 < 0,0005$	371,06
Tiempo iteraciones: 5.239,18[s]					
Tiempo total (iteraciones + opt. de parámetros): 5.613,18[s]					

Tabla F.10: Determinación del límite superior del rango de estabilidad del parámetro $DM_{\{IC,SE\}}$ mediante el algoritmo propuesto.

i	$DM_{\{IC,SE\}}$	$y' = y_0?$	$DM_{\{IC,SE\}}'$	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1$					
0	1,18	Sí	–	–	79,26
1	2,18	No	1,18	$0,1 > 0,0005$	370,76
Ciclo 2: $\Delta l = 0,1$					
0	1,18	Sí	–	–	72,59
1	1,28	Sí	–	–	4,32
2	1,38	Sí	–	–	3,30
3	1,48	No	1,38	$0,01 > 0,0005$	376,06
Ciclo 3: $\Delta l = 0,01$					
0	1,38	Sí	–	–	6,93
1	1,39	Sí	–	–	3,88
2	1,40	Sí	–	–	6,01
3	1,41	Sí	–	–	7,65
4	1,42	Sí	–	–	15,21
5	1,43	Sí	–	–	77,91
6	1,44	Sí	–	–	8,88
7	1,45	Sí	–	–	14,75
8	1,46	Sí	–	–	373,27
9	1,47	No	1,46	$0,001 > 0,0005$	363,85
Ciclo 4: $\Delta l = 0,001$					
0	1,460	Sí	–	–	367,98
1	1,461	No	1,460	$0,0001 < 0,0005$	371,99
Tiempo iteraciones: 2.524,60[s]					
Tiempo total (iteraciones + opt. de parámetros): 2.887,48[s]					

F.5.4. Parámetro E

La determinación de los límites inferior y superior de E comienzan en el valor nominal del parámetro, tras descartarse el uso de una nueva adivinanza inicial. Respecto al límite inferior, las iteraciones efectuadas se resumen en la Tabla F.11, notándose que la tolerancia es alcanzada en el valor 0,1342. Por otro lado, las iteraciones evaluadas con el algoritmo propuesto para la obtención del límite superior de E se encuentran en la Tabla F.12, determinándose que el último valor que permite mantener la solución original es 0,2664, tras alcanzar la tolerancia establecida.

Tabla F.11: Determinación del límite inferior del rango de estabilidad del parámetro E mediante el algoritmo propuesto.

i	E	$y' = y_0?$	E'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 0,1$					
0	0,158	Sí	–	–	63,76
1	0,058	No	0,158	$0,01 > 0,00006$	3,60
Ciclo 2: $\Delta l = 0,01$					
0	0,158	Sí	–	–	57,46
1	0,148	Sí	–	–	3,62
2	0,138	Sí	–	–	7,33
3	0,128	No	0,138	$0,001 > 0,00006$	364,16
Ciclo 3: $\Delta l = 0,001$					
0	0,138	Sí	–	–	5,29
1	0,137	Sí	–	–	5,59
2	0,136	Sí	–	–	83,52
3	0,135	Sí	–	–	8,10
4	0,134	No	0,135	$0,0001 > 0,00006$	382,55
Ciclo 4: $\Delta l = 0,0001$					
0	0,1350	Sí	–	–	8,43
1	0,1349	Sí	–	–	8,82
2	0,1348	Sí	–	–	9,53
3	0,1347	Sí	–	–	8,50
4	0,1346	Sí	–	–	12,37
5	0,1345	Sí	–	–	137,46
6	0,1344	Sí	–	–	19,69
7	0,1343	Sí	–	–	11,34
8	0,1342	Sí	–	–	11,58
9	0,1341	No	0,1342	$0,00001 < 0,00006$	361,13
Tiempo iteraciones: 1.573,83[s]					
Tiempo total (iteraciones + opt. de parámetros): 1.586,05[s]					

Tabla F.12: Determinación del límite superior del rango de estabilidad del parámetro E mediante el algoritmo propuesto.

i	E	$y' = y_0?$	E'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 0,1$					
0	0,158	Sí	—	—	68,93
1	0,258	Sí	—	—	2,75
2	0,358	No	0,258	$0,01 > 0,00006$	370,37
Ciclo 2: $\Delta l = 0,01$					
0	0,258	Sí	—	—	3,93
1	0,268	No	0,258	$0,001 > 0,00006$	372,14
Ciclo 3: $\Delta l = 0,001$					
0	0,258	Sí	—	—	4,55
1	0,259	Sí	—	—	27,45
2	0,260	Sí	—	—	11,08
3	0,261	Sí	—	—	175,14
4	0,262	Sí	—	—	53,60
5	0,263	Sí	—	—	168,44
6	0,264	Sí	—	—	208,39
7	0,265	Sí	—	—	387,49
8	0,266	Sí	—	—	397,22
9	0,267	No	0,266	$0,0001 > 0,00006$	381,87
Ciclo 4: $\Delta l = 0,0001$					
0	0,2660	Sí	—	—	396,68
1	0,2661	Sí	—	—	392,73
2	0,2662	Sí	—	—	429,45
3	0,2663	Sí	—	—	387,94
4	0,2664	Sí	—	—	371,03
5	0,2665	No	0,2664	$0,00001 < 0,00006$	379,38
Tiempo iteraciones: 4.990,56[s]					
Tiempo total (iteraciones + opt. de parámetros): 5.014,30[s]					

F.5.5. Parámetro M

A diferencia de los casos anteriores, la *optimización de parámetros* permitió establecer un nuevo valor inicial para la búsqueda de los dos límites del rango de estabilidad de M . En relación al límite inferior de M , la nueva estimación inicial corresponde al límite buscado, lo que se concluye luego de alcanzar la tolerancia mediante las iteraciones de la Tabla F.13. Se observa que no son evaluados los casos negativos, ya que no corresponden a valores posibles para M . En cuanto a su límite superior, la nueva estimación inicial para M es el mayor valor expresable en GAMS (10^{50}), cuya evaluación mantiene la solución original, por lo que no es necesario efectuar iteraciones, ya que no es posible evaluar valores mayores. El tiempo utilizado para determinar este límite queda determinado por el empleado en la *optimización de parámetros*, tal como se refleja en la Tabla F.14.

Tabla F.13: Determinación del límite inferior del rango de estabilidad del parámetro M mediante el algoritmo propuesto.

i	M	$y' = y_0?$	M'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000$					
0	7,22	Sí	–	–	4,26
1	–992,78	No	7,22	$100 > 0,4$	0,00
Ciclo 2: $\Delta l = 100$					
0	7,22	Sí	–	–	5,59
1	–92,78	No	7,22	$10 > 0,4$	0,00
Ciclo 3: $\Delta l = 10$					
0	7,22	Sí	–	–	4,83
1	–2,78	No	7,22	$1 > 0,4$	0,00
Ciclo 4: $\Delta l = 1$					
0	7,22	Sí	–	–	6,06
1	6,22	No	7,22	$0,1 < 0,4$	3,94
Tiempo iteraciones: 24,68[s]					
Tiempo total (iteraciones + opt. de parámetros): 34,96[s]					

Tabla F.14: Determinación del límite superior del rango de estabilidad del parámetro M mediante el algoritmo propuesto.

i	M	$y' = y_0?$	M'	Δl	Tiempo [s]
Ciclo 1: $\Delta l = 1.000$					
0	10^{50}	Sí	–	–	0,00
Tiempo iteraciones: 0,00[s]					
Tiempo total (iteraciones + opt. de parámetros): 220,59[s]					

F.6. Índice de estabilidad

El rango de estabilidad de cada parámetro se utiliza en la ecuación 5.4, explicada en el Capítulo 5. Con ello, se obtienen los índices de estabilidad inferiores y superiores de cada parámetro y sus valores se encuentran en la Tabla F.15.

Tabla F.15: Índice de estabilidad inferior y superior de los parámetros del ejemplo 4.

Par. [UN]	Nominal	Rango de estabilidad	IE^{inf} [%]	IE^{sup} [%]
$DM_{\{RC,NW\}}$ [m^3/s]	3,83	$3,11 \leq DM_{\{RC,NW\}} \leq 4,55$	18,8	18,8
$DM_{\{AC,SE\}}$ [m^3/s]	6,17	$2,96 \leq DM_{\{AC,SE\}} \leq 8,66$	52,0	40,4
$DM_{\{IC,SE\}}$ [m^3/s]	1,18	$0,43 \leq DM_{\{IC,SE\}} \leq 1,46$	63,2	23,7
E [USD/kWh]	0,158	$0,1342 \leq E \leq 0,2664$	15,1	68,6
M [-]	1.000	$7,22 \leq M$	99,3	NA ^a

^a NA: No acotado. No presenta límite o es un valor superior a 10^{50} , mayor valor expresable en GAMS, concluyéndose que no está acotado el rango de estabilidad (inferior o superiormente, según corresponda).

Observando los índices de estabilidad, se reconoce que el parámetro crítico para mantener la solución original es el costo de electricidad E . Específicamente, una disminución superior al 15,1 % ocasiona que la solución óptima sea diferente.

Entre los casos particulares observados, se destaca que el parámetro $DM_{\{RC,NW\}}$ posee un índice de estabilidad inferior y superior iguales, por lo que una variación (aumento o disminución) mayor al 18,8 % genera un cambio en la solución óptima. Por otro lado, el parámetro M no presenta un límite superior en su rango de estabilidad, por lo que el índice de estabilidad está determinado solamente por el índice de estabilidad inferior. Esta situación era un resultado esperado, ya que el valor de M se escoge suficientemente grande para no afectar a los flujos del problema en las restricciones del tipo *Big M*. Es por ello que la disminución de su valor podría ocasionar que la solución cambie, ocurriendo específicamente para cambios mayores al 99,3 %.

F.7. Validación del algoritmo

La validación del algoritmo se realiza mediante la ejecución del problema original para diferentes valores aleatorios de los parámetros. La representación visual del comportamiento de la solución en cada escenario y su interpretación se encuentra a continuación:

F.7.1. Parámetro $DM_{\{RC,NW\}}$

Las 100 evaluaciones realizadas para estudiar el parámetro $DM_{\{RC,NW\}}$ presentan el comportamiento esperado, ya que todos los valores que se encuentran dentro del rango de estabilidad previamente determinado mantienen la solución, mientras que los que están fuera del rango la cambian, tal como se observa en la Figura F.1. Particularmente, se puede observar que los valores dentro del rango se obtienen en un tiempo inferior al límite impuesto, mientras que la mayoría de los que están fuera alcanzan el tiempo límite, determinándose soluciones enteras. Este efecto en el tiempo de resolución para los casos fuera del rango podría ser utilizado para profundizar estimaciones que permitan calcular con mayor velocidad el rango de estabilidad, notando que los escenarios que estén utilizando un tiempo elevado podrían desencadenar en un cambio de la solución, es decir, tratarse de valores fuera del rango de estabilidad.

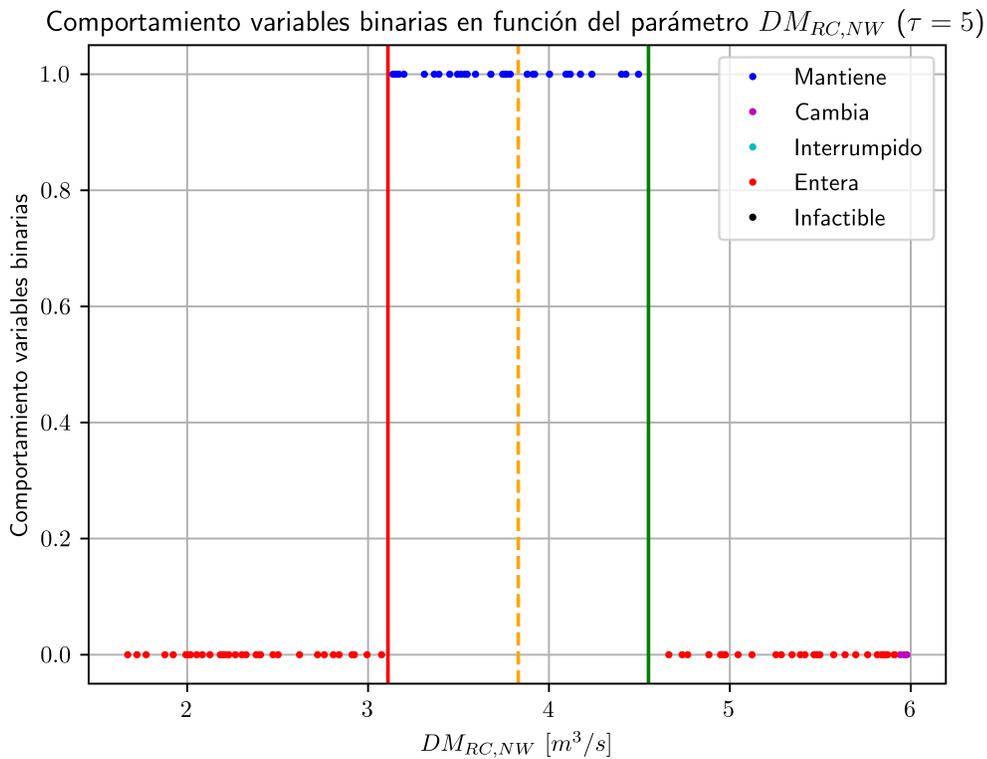


Figura F.1: Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{RC,NW\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.

F.7.2. Parámetro $DM_{\{AC,SE\}}$

Los valores evaluados para $DM_{\{AC,SE\}}$ se analizan en relación a dos zonas de interés en la Figura F.2: interior y exterior del rango de estabilidad. En relación al interior del rango, los valores evaluados en esa zona mantienen la solución, tal como se esperaba. Por el contrario, algunos valores en el exterior no concuerdan con el escenario ideal esperado, es decir, mantienen la solución pese a que se suponía que debiesen cambiarla. De todas formas, se asumía que podría ocurrir con alguno de los parámetros. Este comportamiento se puede relacionar con que existen algunos aspectos en la formulación que podrían generar que el problema presente más de una solución óptima, tal como se concluyó en la sección F.2. En particular, la función objetivo económica se compone del costo de instalación y de operación, por lo que se produce un *trade-off* entre ambos casos, generando que se permitan configuraciones diferentes pese a que no se produzcan cambios en la función objetivo correspondiente.

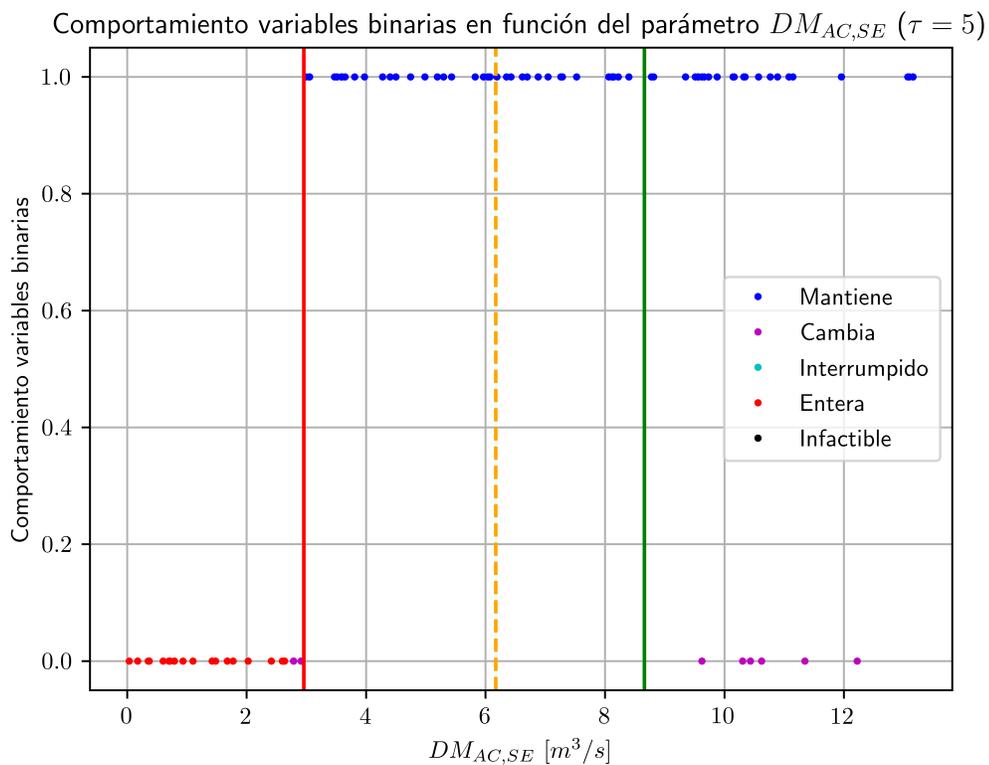


Figura F.2: Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{AC,SE\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.

La razón por la que el algoritmo se detuvo en un valor inferior a algunos de los que se observan que mantienen la solución por sobre el límite superior se debe a un cambio detectado. Precisamente, el algoritmo busca determinar un rango en que se asegure que los valores no cambiarán, es decir, las decisiones estratégicas se mantienen. Es por ello que, pese a la existencia de valores que entreguen la misma solución fuera del rango, este no puede ser expandido porque se incluirían valores que cambian la solución, dejando de ser un rango continuo y afectando la certeza de las decisiones efectuadas.

F.7.3. Parámetro $DM_{\{IC,SE\}}$

Se realizaron 100 ejecuciones con diferentes valores de $DM_{\{IC,SE\}}$, presentando el comportamiento esperado para aquellos valores al interior y exterior del rango de estabilidad calculado, tal como se representa en la Figura F.3. Se observa también que la mayoría de los casos al interior del rango se ejecutan en un tiempo inferior al límite ($\tau = 5$), ya que alcanza una solución óptima. Por el contrario, la mayoría de los casos al exterior del rango requieren un tiempo mayor, deteniéndose su ejecución en el tiempo límite y obteniéndose una solución *entera*, tal como ocurría con el parámetro $DM_{\{RC,NW\}}$.

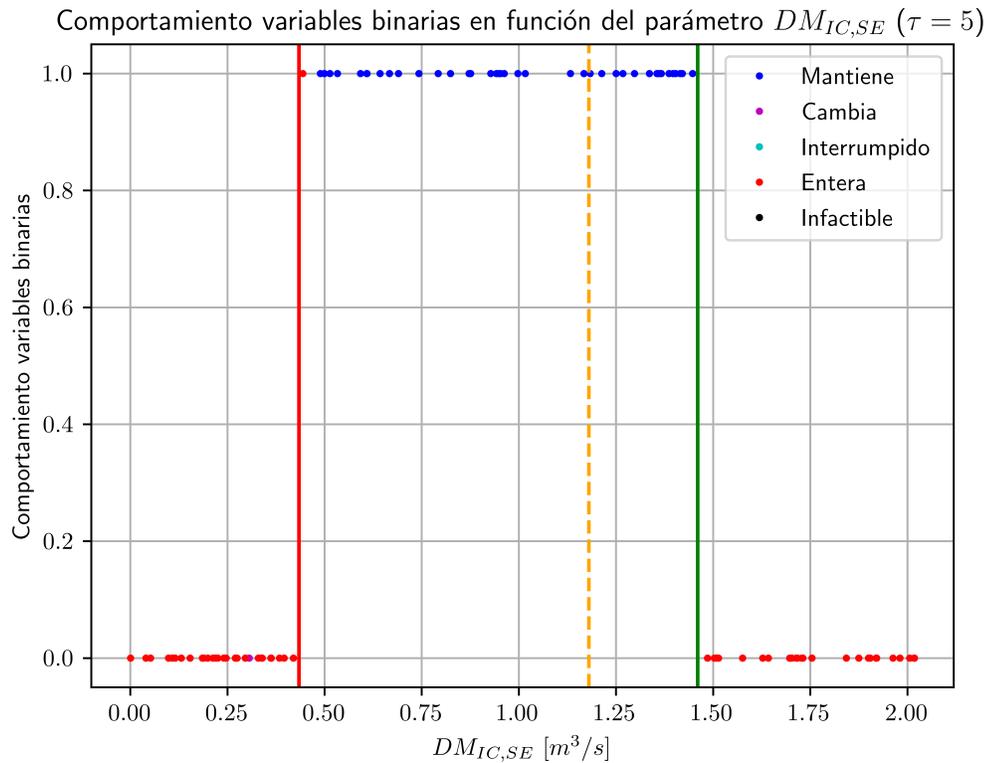


Figura F.3: Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro $DM_{\{IC,SE\}}$. Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.

F.7.4. Parámetro E

Las resoluciones efectuadas para diferentes valores de E permiten observar en la Figura F.4 que los escenarios correspondientes al interior del rango de estabilidad calculado generan la misma solución que para el valor nominal, mientras que los casos externos originan una solución diferente, tal como se esperaba. Se puede notar también que el tiempo máximo para las ejecuciones de los valores al interior del rango no es alcanzado, situación que contrasta con las soluciones *enteras* de las evaluaciones al exterior del rango. Esta situación es similar a la de dos de las demandas de agua analizadas previamente ($DM_{\{RC,NW\}}$ y $DM_{\{IC,SE\}}$).

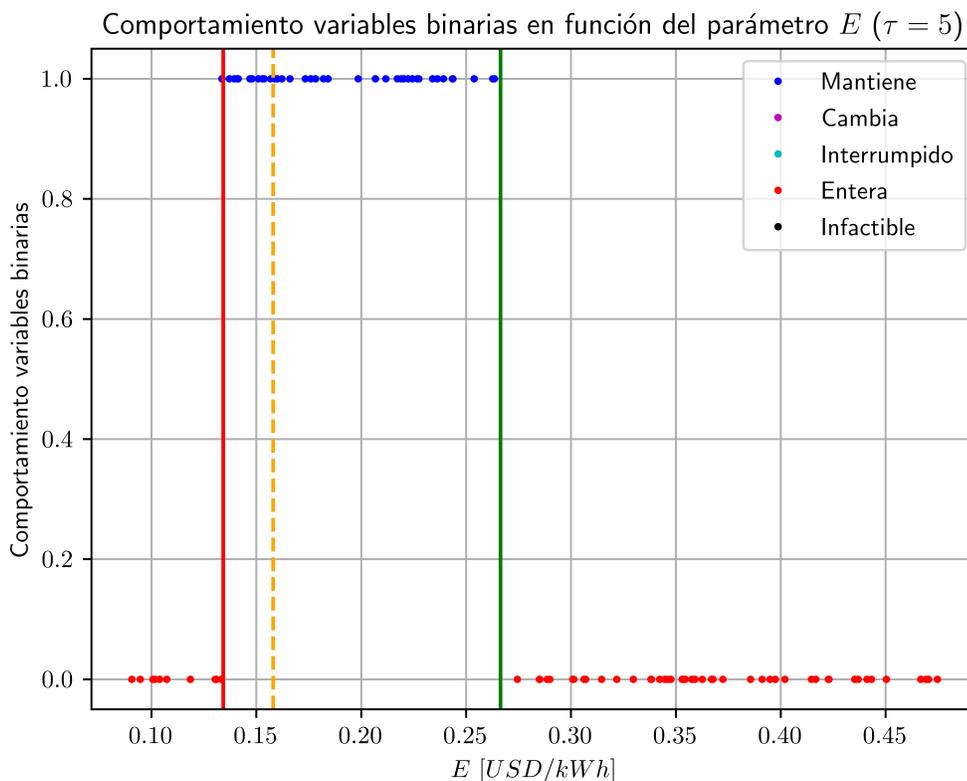


Figura F.4: Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro E . Línea roja: límite inferior calculado. Línea verde: límite superior calculado. Línea naranja discontinua: valor nominal.

F.7.5. Parámetro M

El análisis de los escenarios aleatorios de M permite reconocer que las soluciones se comportan según lo supuesto, a partir de lo observado en la Figura F.5. Específicamente, el valor $M = 3,99$ que es menor al límite inferior cambia la solución, mientras que todos los demás escenarios generan una solución igual al caso nominal, según lo esperado por estimarse que no presenta un límite superior.

El comportamiento obtenido concuerda con el propósito del parámetro M en las formulaciones del tipo *Big M*, ya que se requiere escoger un valor suficientemente grande para que no condicione a los flujos del problema, si no que solamente efectúe una relación entre la existencia de las conexiones y los nodos instalados. Por lo tanto, escoger un valor inferior al determinado como límite inferior afectaría la solución. Por el contrario, en GAMS el uso de un valor con un gran orden de magnitud puede ocasionar errores de redondeo en las resoluciones, por lo que se debe corroborar que no se produzcan valores inesperados en la solución. Por ejemplo, podrían presentarse valores en variables binarias del orden de 10^{-8} , que podrían aproximarse a cero, pero que no debiesen adoptar aquellos valores por tratarse de variables binarias (0 – 1). Es por ello que se asigna un valor suficientemente grande para acotar los posibles valores involucrados en las restricciones del tipo *Big M*, pero de órdenes de magnitud no tan elevados para no ocasionar problemas numéricos en la solución.

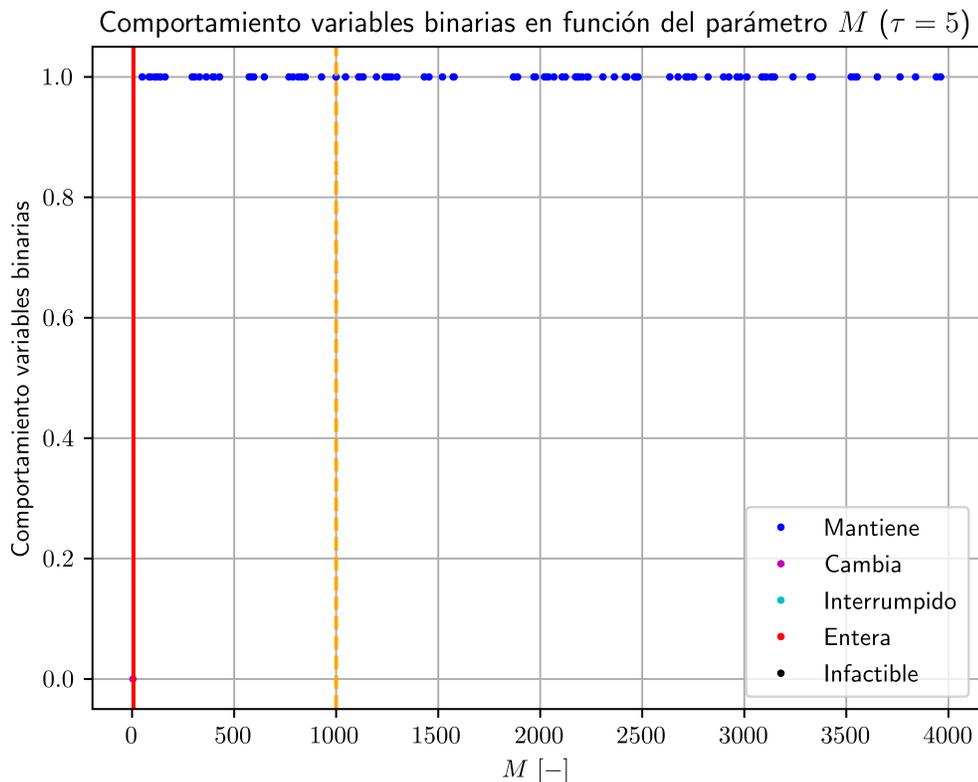


Figura F.5: Comportamiento de las variables binarias frente a cambios aleatorios en el parámetro M . Línea roja: límite inferior calculado. Línea naranja discontinua: valor nominal.

Anexo G

Publicación

A continuación, se presenta la publicación realizada en la revista *Industrial & Engineering Chemistry Research* [126] y que se basa en esta tesis.

Design water networks through MINLP formulation: An algorithm to find the stability region of parameters

Fernando Arenas-Araya, Javiera Vergara-Zambrano, and Felipe A. Díaz-Alvarado*

*Department of Chemical Engineering, Biotechnology, and Materials, Faculty of Physical
and Mathematical Sciences, Universidad de Chile, Av. Beauchef 851, Piso 6-poniente,
Center for Sustainable Design and Process Systems Engineering (ProSus), 8370456
Santiago, Chile*

E-mail: felidiaz@ing.uchile.cl

Abstract

This paper proposes an iterative algorithm to calculate the stability region of parameters in MINLP problems. It is based on a previous algorithm developed for MILP problems with modifications to accelerate the search: (i) initial point is calculated; (ii) search region is bounded by context information; (iii) time limit for iterations is estimated; (iv) the output is a stability index. The novelties of this research are the first algorithm for MINLP cases and a stability index that recognizes the critical parameters to the resilience of the solution. The proposed algorithm is applied to the water network design, and the stability region is calculated for six parameters. The modifications reduce the implementation time by 96.6 % in the case study compared to the referential version. A potential direction for this work is to include a more detailed analysis of the mathematical expressions to make the algorithm faster.

Introduction

One of the main objectives of engineering is to find the best possible solution with the available resources.¹ Optimization approaches seek such a solution through strategies that are best suited to the characteristics of the objective function, constraints, and variables.²

Chemical engineering optimization has been developed using different formulations. For example, linear programming (LP) has been used by Jezowski et al.³ to determine the area target for heat exchanger networks. Saffer and Doyle⁴ generated a predictive control model of a paper machine. Finally, Schack et al.⁵ identified an optimal solution for a Renewable-to-Chemicals production network. On the other hand, nonlinear programming (NLP) has been formulated by Poku et al.⁶ for gasoline blending, by Alvarães et al.⁷ to optimize a biodiesel plant, and by Kralj and Glavič⁸ to separate CO₂ in the methanol process.

In multiple process systems engineering applications, discrete decisions need to be modeled.⁹ Decision-making can be organized at a hierarchical level (strategic, tactical, and operational).¹⁰ Long-term decisions are at the strategic level and usually about configuration design.¹¹ The material and immaterial resources are organized at the tactical level.¹² Finally, the operational level is related to short-time decisions and can be revised and adjusted frequently.¹¹

Mixed-integer formulations have been developed to consider discrete and continuous decisions.¹³ The mentioned LP and NLP could be complemented, including integer variables, transforming to mixed-integer linear programming (MILP) and mixed-integer nonlinear programming (MINLP).⁹ MILP formulation has been used for different applications. For example, Terrazas-Moreno et al.¹⁴ scheduled and assigned tank farm operations, Díaz-Gómez et al.¹⁵ optimized the design of shale gas industrial parks, and Stebel et al.¹⁶ formulated a MILP problem to aid planning activities in complex pipeline networks.

On the other hand, MINLP formulation allows solving problems with non-linearities common in the process models.¹⁷ Although it is possible to simplify process design and synthesis problems using linear approaches, the most common formulations are NLP

and MINLP because they need to make the equations that compose them explicitly.⁹ Moreover, nonlinear behavior is intrinsic to some processes, and the implementation of MINLP formulation is required to obtain more precise solutions.¹⁸ This kind of problem has been applied in different process engineering cases. For example, Stichlmair and Frey¹⁹ formulated a problem of designing reactive distillation processes, Ciric and Floudas²⁰ optimized retrofitting heat-exchanger networks, and Szitkai et al.²¹ proposed a model for synthesizing mass exchange networks.

In process optimization, the decisions are represented by variables. For example, in process synthesis, the continuous variables usually represent flows and temperatures of the streams, and discrete variables, the equipment selection for performing a process task.²² Discrete variables included in design and process synthesis problems are usually binary (0-1 variables).¹⁷ However, if those are not 0-1 variables and take large values, they often are approximated as continuous.⁹

One of the main tasks in chemical process design is making decisions with incomplete information.²³ Process systems have uncertainty related to their process (flowrate or temperature variations) or from the model that represents the process (product demands, environmental conditions, among others).¹³ To model processes under uncertainty, it is necessary to make assumptions. For instance, there are assumptions related to assigning value to the parameters in an optimization problem affecting the output. Each assumption must be raised for specific contexts since their validity may vary in different settings.²⁴

Assumptions related to strategic decisions affect for a long time. Specifically, these decisions are made longer than operational and tactical ones¹¹ and are considered static during the implemented period.¹⁰ Therefore, it is necessary to know that the obtained solution is optimal and that an error in estimating the value assigned to parameters does not generate wrong strategic decisions. Different studies about the effect of change in parameters in the solution have been carried out, and the main findings are presented in the next section.

Literature review

Input data highly affect the output of decision aid. Therefore, the sensitivity of the conclusions must be checked to focus on the most critical parameters and revise those more carefully.²⁵ In recent years, the number of scientific publications in which sensitivity analysis is used has increased.²⁶ Most of them perform One-At-a-Time (OAT) factor analysis, modifying only one input parameter at a time and recording the changes they produce in the output variables.²⁷ An evolution of these cases corresponds to the study of changes in more than one parameter simultaneously, known as a global sensitivity analysis (GSA).²⁸

Sensitivity analysis have been studied and formalized to a different extent depending on the type of optimization problem. One of the main topics of its application in mathematical programming is to determine the stability region of parameters, in which the optimal solution does not change as a result of the change in parameter values around a nominal value.^{29,30}

For linear programming (LP) problems, sensitivity analysis has been based on the simplex method.³¹ This method allows two types of information: dual solution and interval or range of values of the coefficients on the right-hand side (RHS) or the cost function, and both metrics must be used together to avoid wrong conclusions.³¹ In the case of nonlinear programming (NLP) problems, the sensitivity analysis has been studied according to the necessary, and sufficient optimality conditions.³² Second-order sufficiency conditions are essential for optimal solutions to be differentiable functions of the analyzed parameter.³³

In contrast to the LP and NLP cases, problems with integer or mixed variables are more challenging due to the discontinuous domain that characterizes them, which has rarely been used in practical applications.³⁰ In the case of mixer-integer linear programming (MILP) problems, Yi and Lu³⁰ elaborated one of the most recent proposals for studying sensitivity analysis in MILP problems. As it is an optimal post analysis, their proposal requires calculating the solution for the nominal values of the parameters. With this, multiple resolutions of the problem are carried out using different values of the parameter, defined from a step that adapts as it approaches the limit of the stability region. On the other

hand, the literature review determined a lack of strategies to obtain the stability region of parameters in MINLP problems. The present paper addresses this issue by extending the work carried out by Yi and Lu³⁰ for MILP cases toward MINLP problems.

One of the motivations to cover this lack is that multiple investigations have been developed for MINLP problems. Besides, it is expected that more work will be created and deepened in the following decade on this topic.³⁴ Also, studying mixed-integer cases is relevant because, in the mathematical formulation, strategic decisions can be represented by binary variables. For example, Latifi et al.³⁵ associated those variables with the choice of locations, Faria and Bagajewicz³⁶ used those to represent connections between units, and Lienqueo et al.³⁷ for selecting techniques to be used in the processes, among other cases. Given the evolution of algorithms to solve MINLP problems, the first approach proposed to determine the stability region of parameters in these problems is carried out through a practical approach. It can be complemented in the future with new research developed on this topic. The context in which it is applied is briefly described as follows.

Among the engineering challenges, one of the most significant challenges is solving climate change effects.³⁸ In particular, the scarcity of water stands out due to the increase in temperature and variations in the level of rainfall in different regions.^{39,40} Efforts to use the available resources sustainably have increased in recent years in research, industry, politics, and society.⁵ In this context, engineering seeks to rethink how processes and products are designed, built, operated, and evaluated.⁴¹ Consumption and production must be designed with a sustainable approach to deal with the scarcity of water, food, and goods.⁴² In particular, the security of water supply is a crucial aspect due to the vulnerability of water availability to continuous changes in weather patterns,⁴⁰ and the need for this resource in daily life and different production processes.⁴³

Water distribution networks (WDN) can represent how consumers and water sources are related. WDN optimization searches for the best way to transfer water from sources to the users, satisfying their requirements.⁴⁴ Different reviews have been developed for this

kind of network, and its components.^{45–48} Besides, numerous researchers have formulated water network optimization problems as MINLP. For example, Altherr et al.⁴⁹ proposed an algorithm to design resilient water networks for high-rise buildings. Kralj⁵⁰ stated an MINLP model to collect industrial wastewater using a separation technique. Jiménez-Gutiérrez et al.⁵¹ formulated a model for integrating energy, mass, and properties in water networks simultaneously. Lastly, Pungthong and Siemanond⁵² optimized the design of water/wastewater networks with multiple contaminants, among others.

Like in other engineering projects, the design of WDN requires making responsible assumptions. Moreover, considering that the use of assumptions that are not valid is likely to cause results that are not desired.²⁴ In water networks design, the parameters of the proposed models are a source of uncertainty, as they require the assumption of a value. It is proposed that the first analysis of the stability region of MINLP problems would be carried out in water networks problems due to the critical world water situation. Finally, because of the permanence in time of the strategic decisions, the case study focuses on examining the behavior of the binary variables, which can be associated with layout modifications.

This paper proposes an algorithm to determine the stability region of the parameters of MINLP problems in water networks. Specifically, the algorithm quantifies the intervals for parameters where the obtained water network is still optimal. On the other hand, another novelty of this work is a development of an index to compare stability regions of parameters. The output allows establishing a prioritization of the parameters that must be analyzed in greater depth, thus achieving greater certainty about the resilience of the solution obtained.

The outline of the article is as follows. First, the methodology used in this paper is presented. Then, the algorithm section explains the main steps of the process to determine the stability region of parameters. Subsequently, the case study is presented, and the algorithm is applied to that optimization problem. Afterward, the results are discussed. Finally, in the last section, the main conclusions are highlighted.

Methodology

In the methodology proposed, the stability region is calculated for those parameters whose value is supported by assumptions. Initially, a literature search was conducted to acknowledge that no tools have been developed for MINLP problems applied to practical problems. The best option found to calculate the stability region is the one proposed by Yi and Lu³⁰, which determines the stability region of parameters in MILP problems. As no work of this type has been developed for MINLP cases, the reference MILP algorithm has been chosen as an initial comparison and source of inspiration.

New steps are proposed to the work of Yi and Lu³⁰ to make the algorithm efficient for MINLP cases. The objective of these new steps is to guess a starting point closer to the limit of the stability region to accelerate the search and bound the resolution time of each iteration and the range of search with information on the context. In addition, the output allows comparing stability regions of different parameters. The methodology and resources used are described as follows.

Determining stability region

The stability region of parameters is defined over the behavior of binary variables, and changes in other variables are allowed. The prioritization is based on the relation between binary variables and strategic decisions, as these decisions are hardest to change if the solution is not the best. Therefore, it is essential to confirm that decisions are correct even if the parameters change their values. Then, while the binary variables keep their original value, it is considered that the parameter value is inside of its stability region.

The effect of the parameters in the solution is recognized with iterative evaluations of one parameter at a time. The iterations correspond to the resolution of the optimization problem for different parameter values. These iterations continue until a change in binary variables is detected. This procedure is inspired by Yi and Lu³⁰ algorithm. Its initial search

step and the way to calculate the tolerance are the same, but the initial guess changes in this new proposal. It consists of solving a subproblem, fixing the binary variables solution, and considering the parameter of interest as a variable. The formulation of this subproblem is presented in the *Formulation of the parameter optimization* section.

The original resolution allows knowing a referential time to the iterations. It is also considered in the proposed algorithm among the original value of the binary variables. It allows limiting the time of each iteration to the scale and time of the analyzed case. In addition, its consideration allows for reducing the algorithm implementation time.

The range of search is bounded by previous information. Some physical and logical bounds are incorporated to analyze only the possible parameter values in the context of each problem and avoid iterations in infeasible values, allowing a reduction of the time spent by the algorithm.

Calculating stability index of parameters

A comparison of the stability region of parameters is necessary to decide which parameter is riskier to generate a different optimal solution. However, the comparison cannot be made directly with the value of the stability region because of different orders of magnitude. Therefore, it is suggested to normalize those values to avoid wrong interpretations.

A proposed index compares the stability of parameters quantifying the risk of a change in the solution. The main idea is to calculate the margin of error using the nominal value and the extremes of the stability region. The index is calculated as the normalized difference between the nominal value and the closest limit. Specifically, it allows knowing how near or far the estimation is to the possibility of the optimal solution being different. Therefore, a higher stability index represents a higher margin of error in the solution and resilience for the optimal solution.

Computational resources and implementation

Each iteration requires the resolution of an optimization problem, which is carried out using the GAMS⁵³ software and the BARON⁵⁴ solver, specialized for MINLP problems. In addition, the Python⁵⁵ software is also used because the different executions are carried out with different values of each parameter, updated according to the behavior of the binary variables. Both software were implemented in an Intel Core I7-7700 HQ computer with 16 GB of memory. The interaction of the two previous software is carried out as follows:

Python is used to declare nominal values of parameters and the model to execute GAMS with this information through GAMS Python API (Application Programming Interfaces). Once the results are obtained, the parameter value is updated in Python, and a new solution is obtained. Then, the model is executed again with a different parameter value depending on the change or permanence of binary variables values. One of the factors considered about GAMS is a maximum execution time, whose default value is 1000 seconds (the user can modify it). This time is part of the analysis to be carried out based on the results of the referential algorithm and the new one.

It is important to highlight the solver can change in other case studies. The use of BARON in this article is not directly linked with the algorithm, but the case study. In other cases, the algorithm to define the stability region of parameters should use the same solver chosen for the original implementation because the methodology requires solving the same problem iteratively (changing parameters). The use of other solvers is critical in large-scale problems because of the execution time.

The general procedure of the proposed algorithm for MINLP cases is presented in the next section. Subsequently, a case study is used in the *Illustrative example* section to contextualize the elements of the algorithm and evaluate its performance.

Algorithm

An iterative algorithm to determine the stability region of parameters in MINLP problems is developed based on the MILP version of Yi and Lu³⁰. The main goal of the new version is to accelerate the search of limits of stability region because nonlinear problems are more challenging than linear problems.

The main characteristics of the proposed version in contrast to the referential algorithm³⁰ are: (i) the range of analysis is bounded by previous information of parameters; (ii) the initial point of iterations is estimated, and it is closer to the searched limit; (iii) a time limit is incorporated based on the resolution time of the original problem; (iv) the output of the algorithm is an index to compare stability region of different parameters. The justification of each one is presented in Table 1.

The referential algorithm developed by Yi and Lu³⁰ has been represented in Figure 1 with green boxes. Red boxes have been included to show the main proposed changes to adapt the methodology to MINLP models. Orange boxes are also included to highlight those steps in the original algorithm with a minor modification.

The mathematical expressions of the elements present in the steps of Figure 1 will be presented in the *Algorithm application* section adapted to the case study (Figure 8).

The proposed procedure is summarized: The algorithm starts selecting the interest parameter and variables (all the binary variables by default). The search parameters (initial step and tolerance) are calculated with the nominal value of the parameter. The step corresponds to the difference between two consecutive values evaluated by the algorithm, and the tolerance is the minimum value that can adopt the step. Solving the original problem allows obtaining the value of variables the user expects to maintain. Those results are implemented in a subproblem to determine a new possible initial guess (named optimized parameter). If the candidate for the initial guess maintains the original solution, it is accepted to start the iterations. In another case, the iterations start from the nominal value. Those iterations continue until the detention criteria related to the tolerance or the maximum

Table 1: Distinguishing features of the new algorithm

Features of the new algorithm	Justification
(i) Range of search bounded by previous information.	Mathematical results about the stability region must be contrasted with the reality. For example, the algorithm should not evaluate negative values for process flows because of infeasibility about taking those values in reality. For which parameters the user has previous information about their nature or infeasibility, it is possible to incorporate that information at the beginning of the algorithm. Hence, the search is bounded to the interest range and spends less time.
(ii) New initial point for the algorithm.	Starting the search closer to the limit of the stability region can reduce the total iterations required to find it. The referential algorithm starts from the nominal value, but it is proposed to use a better initial point to accelerate the search. Because the search of stability region involves two limits (upper and lower), it is necessary to estimate two initial guesses, independent between those. Each value is obtained, minimizing and maximizing a subproblem with fixed binary variables detailed in the named <i>parameter optimization</i> procedure.
(iii) Time limit per iteration.	The time of each iteration must have a similar order of magnitude to the time spent in solving the original problem. If an individual iteration takes longer, the process requiring multiple iterations could take considerable time. To do the algorithm faster, in the optimization software is possible to edit the default time for iteration. For example, the case study presents some iterations with a longer time.
(iv) Stability index to compare the stability regions.	To compare the extension of the stability region of different parameters directly can generate wrong conclusions because of a different order of magnitude. Therefore, the comparison requires normalization and is included in a proposed index. Also, the risk of a change of binary variables is conditioned by nearness to limits of the stability region. Then, the stability of parameters must be contrasted by the index, considering the relative distance of the nominal value and those limits.

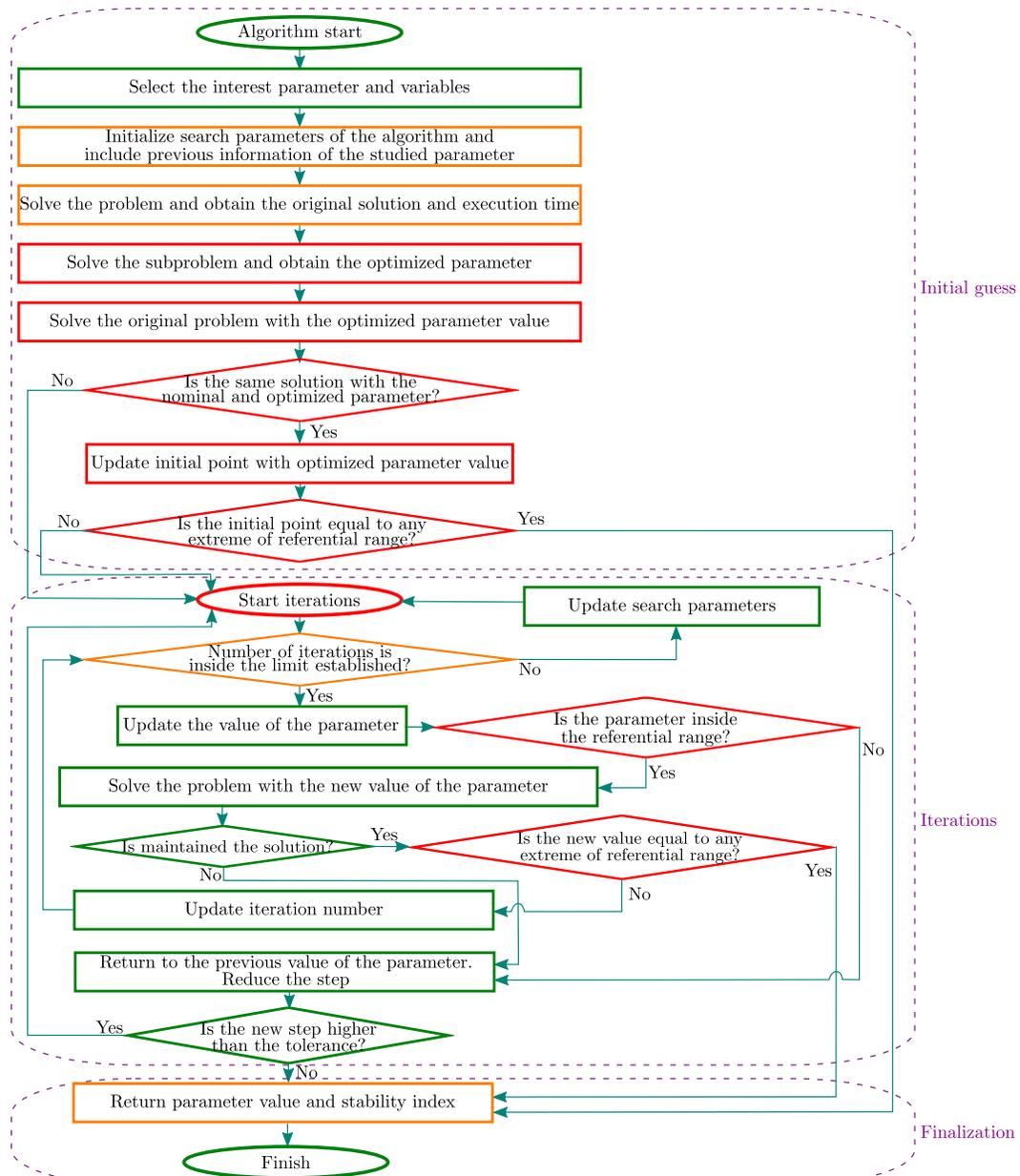


Figure 1: The proposed algorithm to obtain the stability region of parameters in MINLP problems. Green boxes: algorithm steps maintained from Yi and Lu³⁰ algorithm. Orange boxes: Modified from Yi and Lu³⁰ algorithm. Red boxes: New steps proposed for the MINLP version. Own elaboration.

number of iterations allowed is accomplished. The whole procedure considers evaluating only values inside the referential range previously defined to avoid assessing values that are infeasible in reality.

The explanation of the subproblem is shown in the *Initial guess analysis* and the *Formulation of the parameter optimization* sections. Subsequently, a more detailed description of the whole algorithm is developed in the *Illustrative example* section using a case study to exemplify the algorithm elements.

Initial guess analysis

An introductory optimization problem is included to generate graphical interpretations for the initial guess analysis. This example is a small-scale MINLP problem represented by the expression (1). The model was proposed by Kocis and Grossmann¹⁷ with some changes: the original problem does not include the parameters a and b . Specifically, the original value of a is 1.25, and that of b is 1.6. To study how much these parameters can be modified, general expressions for a and b are introduced.

$$\left\{ \begin{array}{ll} \min & z = 2x + y \\ & x, y \\ \text{s.a.} & \\ & \mathbf{a} - x^2 - y \leq 0 \\ & x + y \leq \mathbf{b} \\ & x \geq 0 \\ & y \in \{0, 1\} \end{array} \right. \quad (1)$$

The solution of the binary variable for this problem is 1. In this case, it is sought to know which values of a , and b allow the optimal solution to consider $y = 1$, namely the stability region of a and b . The proposed algorithm was applied in this example, and the stability region for a and b are expressed in equations (2) and (3), respectively.

$$0.25 \leq a \leq 1.6 \quad (2)$$

$$1.5 \leq b \quad (3)$$

Analyzing the introductory problem (1) allows recognizing a relationship between the stability region and active constraints. Points where active constraints change to inactive and vice-versa are of interest to examine if they correspond to a limit of the stability region of the parameter. For example, in Figure 2 it is observed four scenarios with different values for parameter b of the introductory example. Those graphics plotted the constraints (4) and (5) of the problem relating to variables x and y . Also, four points are highlighted because they are essential for the analysis: point C is the intersection of equation (4) and the straight line $y = 1$, which is the original problem solution; the point D is the intersection of constraint (5) and the straight line $y = 1$, so the original solution ($y = 1$) is unchanged only in the line segment \overline{CD} ; the point E corresponds to the intersection of the constraint (4) and the straight line $y = 0$; the point F is the intersection of the expression (5) and the straight line $y = 0$. Because y is a binary variable, feasible solution is only in the line segment \overline{CD} or \overline{EF} . The four scenarios of parameter b are illustrated in Figure 2 and described as follows:

$$a - x^2 - y \leq 0 \quad (4)$$

$$x + y \leq b \quad (5)$$

- (a) $b = 1.6$: Is the nominal value of parameter b , and the optimal solution is represented by point C . Constraint (4) is active, and constraint (5) is not.
- (b) $b = 1.5$: A reduction of b to the value 1.5, points C and D are equals. In this case, constraint (5) became active, and the optimal solution was maintained.
- (c) $b = 1.3$: If $b < 1.5$, the optimal value includes a change in the binary variable. Thus,

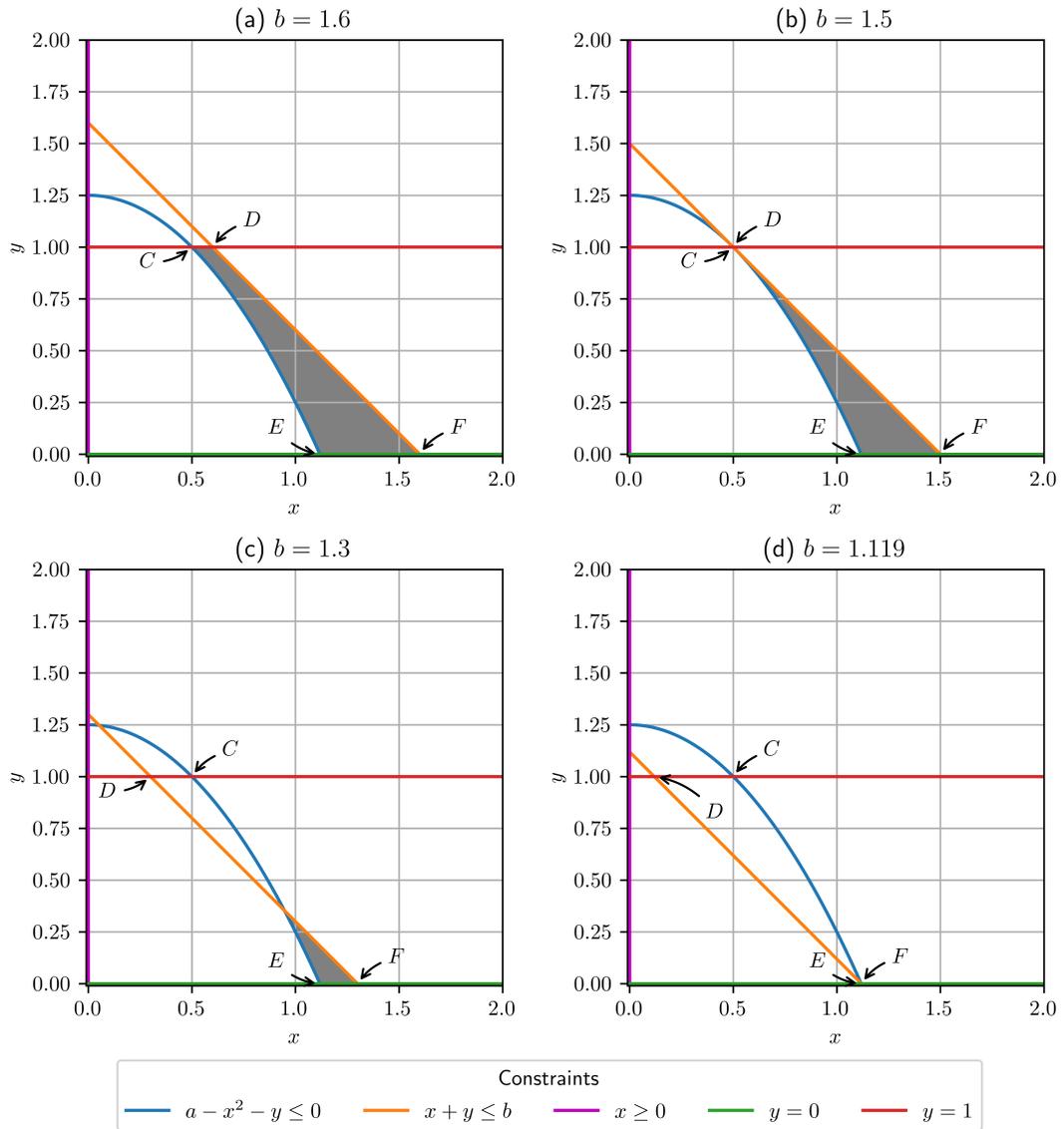


Figure 2: Constraints of the introductory example for changes in parameter b .

the optimal solution corresponds to point E .

- (d) $b = 1.119$: The minimal value of b that allows obtaining a feasible solution. In this case, points E and F are the same point. Both constraints are active, although the solution $y = 0$ is different from the original.

The effect of changes in parameter b , shown in Figure 2, expresses a relation between one of the limits of the stability region ($b = 1.5$) and the change of status of the constraints (a non-active constraint became active). A similar situation happens to plot the changes of the values of parameter a . However, the stability range of a differs from that of b because it is upper and lower bounded. How the constraints behave for different values of a is reported below, based on the scenarios in Figure 3.

- (a) $a = 1.25$: It corresponds to the nominal value of the analyzed parameter. For this case, the solution is the one represented in point C , with constraint (4) active and expression (5) not active.
- (b) $a = 1.36$: By increasing the parameter a to 1.36, C and D correspond to the same point. Thus, constraint (5) becomes active, and the value of the binary variable y in the optimum is maintained.
- (c) $a = 2.56$: Suppose the parameter a continues to increase up to 2.56. In that case, it is no longer possible for the solution to maintain the value of the binary variable (it becomes $y = 0$). Particularly, the maximum value allowing a feasible solution is 2.56, with E and F at the same point, and the two constraints of interest become active. Higher values of a would make the solution infeasible.
- (d) $a = 0.25$: Decreasing the value of a determines that its smallest value is 0.25 for the solution to consider the value of the binary variable obtained in the original case ($y = 1$). Due to the formulation of the objective function $z = 2x + y$, lower values of a would make a solution with $y = 0$ preferable.

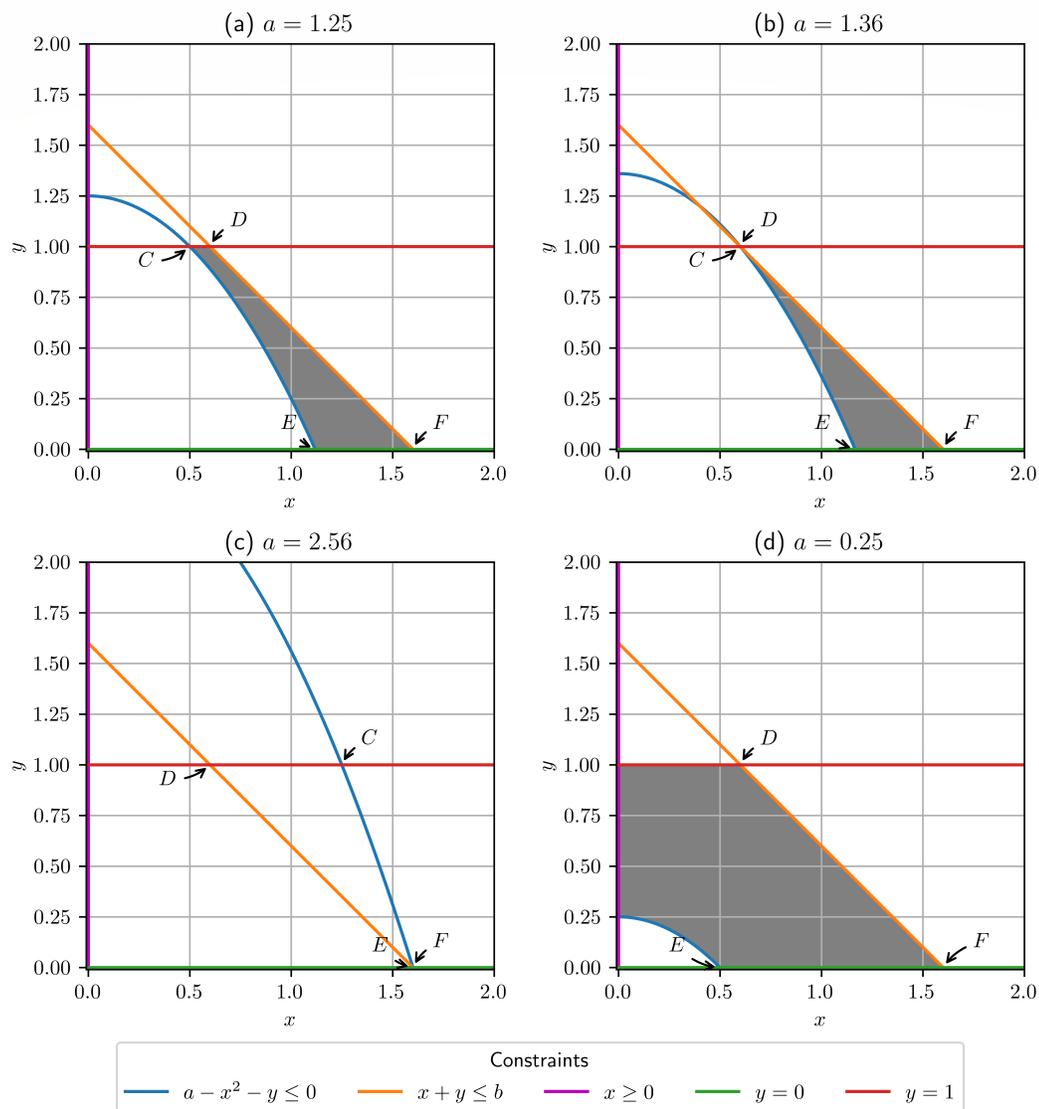


Figure 3: Constraints of the introductory example for changes in parameter a .

Small-scale problems can be analyzed graphically and equation-by-equation. However, this detailed study is not feasible in large-scale models. Considering that behavior, a reformulation of the problem (subproblem) is proposed, changing the nature of the study parameter to a variable, and binary variables are fixed with the original solution. This idea is explained as follows:

A problem reformulation is proposed and implemented, fixing binary variables values and considering like variable the parameter of interest, to determine the maximum and minimum value feasible for a parameter without changing the original binary variables solution. This subproblem only expresses a feasible region, but it is not secure that the total values kept the values of the binary variables in the original problem. Thus, this region does not necessarily correspond to a stability region, but it is evaluated to define a new starting point of the algorithm.

Graphically, the effect of the parameters b and a (variables in this new case) in the constraints are observed in Figure 4(a) and Figure 4(b), respectively. In each case is highlighted the point C , that is the solution of the optimization problem when b is minimized, and a is maximized. The opposite cases, where b is maximized, and a is minimized, are not reported because the solution is not bounded.

The graphics show that the minimum feasible value of b is 1.5. Because the nominal value of b is 1.6, it is suggested to search for the lower limit between 1.5 and 1.6. On the other hand, there are no conclusions about the maximization of b , because that solution is not bounded. Concerning a , the maximum value is 1.36, while a lower bound is not determined. Specifically, point C in each graphic of Figure 4 is a limit of the stability region ($b = 1.5$ and $a = 1.36$) though it is not guaranteed that behavior for all the problems.

The problem size shows the graphical behavior, but the cases with more variables and equations are challenging to represent. Thus, in that situation is necessary to solve the subproblem. Furthermore, considering that those solutions from the subproblem could not be part of the stability region of the parameter, how is it possible to use these values to

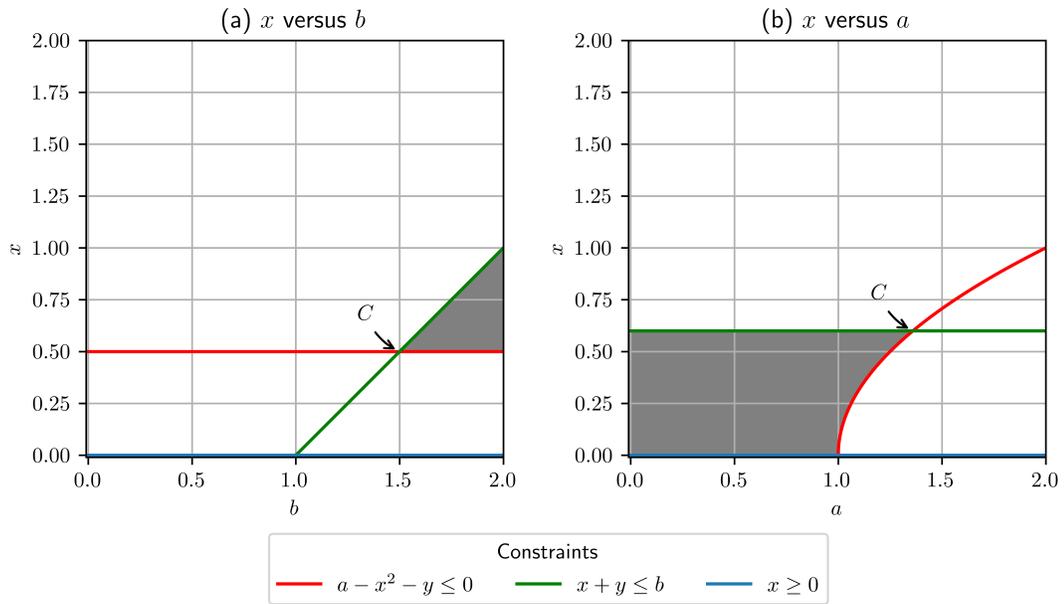


Figure 4: Constraints of the subproblem of introductory example versus parameters b and a .

search the region limits? That is analyzed as follows:

Formulation of the parameter optimization

Mathematically, the proposed subproblem to study parameter b in the example of expression (1) is raised in expression (6). Minimization is used to determine a new starting point to search for the lower limit of the stability region of the parameter. At the same time, maximization allows estimating a new point for the upper limit of the range.

$$\left\{ \begin{array}{ll} \min(\max) & \mathbf{b} \\ & x, \mathbf{b} \\ \text{s.a.} & \\ & \mathbf{a} - x^2 - y \leq 0 \\ & x + y \leq \mathbf{b} \\ & x \geq 0 \\ & \mathbf{a} = 1.25, y = 1.0 \end{array} \right. \quad (6)$$

To use the subproblem solution will be necessary to evaluate some scenarios that are explained to the minimization situation (maximization is analog):

1. The evaluation of the subproblem solution (b_{op}^{min}) in the original problem generates that the binary variable value is conserved. Thus, b_{op}^{min} is inside the stability region and is upper or equal to the lower limit.
2. The evaluation of the subproblem solution (b_{op}^{min}) in the original problem makes the binary variable change. Thus, b_{op}^{min} is outside the stability region and is lower than the lower bound.

For the introductory problem, in the first case, it is recommended to use b_{op}^{min} as a new initial point because it is the closest to the lower limit. On the contrary, the second case has a b_{op}^{min} value that changes the binary variables, which is a reason for not suggesting using it. There is a possibility that in the second case b_{op}^{min} is near the limit, but it is not insurable a priori. The situations described show that the new initial point must be implemented only if the search of the limits of the stability region is favored.

The introductory problem was presented to explain the reformulated problem or subproblem but does not show a specific context. Other algorithm steps require context information and characteristics of the problem in reality. The case study is presented below to detail the elements involved in the proposed algorithm.

Illustrative example

This section presents the case study, their formulation, and the application of the referential algorithm of Yi and Lu³⁰ and the proposed one. Finally, the performance of both algorithms is compared.

Case study

The case study corresponds to the water distribution and treatment networks. These networks are present in different industrial processes and on the scale of urban networks, with the primary objective of transferring water to the consumption nodes, which have diverse specifications.⁵⁶

The essential elements of a water network are processes or stages that use water (different kinds of consumers, as industrial,⁵⁷ domestic,⁵⁸ among others), treatment processes, water sources, water disposal areas, mixers, and flow splitters, and in some cases, it may include tanks or water storage areas.⁴⁷ Figure 5 shows the representation of a water network according to the elements indicated above. Given the growing importance of water and material resources, optimization has emerged as a set of techniques allowing more efficient use of resources.⁵⁹ The case study corresponds to one of those efforts.

Problem statement

The case study is a water network formulated as an MINLP with treatment and recirculation proposed by Ahmetovic and Grossmann⁶⁰. The nodes involved in the superstructure are the water source (N), process units (P_1 and P_2), treatment units (T_1 , T_2 and T_3), flow mixers of the process units (MPU_1 and MPU_2), treatment unit flow mixers (MTU_1 , MTU_2 and MTU_3), final flow mixer for discharge to the sink (MF), process unit flow splitters (SPU_1 and SPU_2), unit flow splitters treatment (STU_1 , STU_2 and STU_3), flow splitter of the initial water source (SI) and sink (S). The goal of the problem is to determine the best

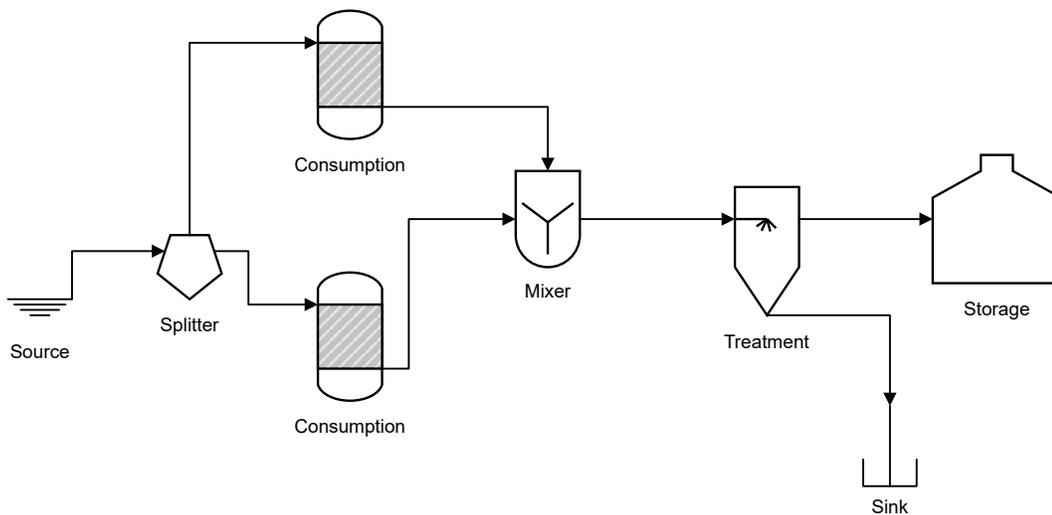


Figure 5: Representation of a water distribution and treatment network. Own elaboration with information from Jezowski⁴⁷.

configuration of the flows that minimize the total costs. The whole formulation is available in Ahmetovic and Grossmann⁶⁰ publication.

The authors presented various options to connect the network units. One of them was considered as the case study: it is possible to share flows between processes or send them to any of the treatment stages (T_2 and T_3), discarding that after the processes they are connected with the pre-treatment T_1 . The local recycle option for each node has also been removed. With this, the superstructure is shown in Figure 6. In this image, black arrows correspond to installations made by default, considering the relationship between mixers and splitters with each processing unit, treatment unit, source, and sink. Other connections are possible, and binary variables represent their installation decision (1 for installation, 0 if not). That cases are represented by arrows with different colors related to the source flow splitter to facilitate their visualization.

Six parameters are analyzed for this problem. Those are the incoming flow to the process unit P_1 , $FPU_{P_1}^{in} = 40 \text{ t/h}$; the incoming flow to the process unit P_2 , $FPU_{P_2}^{in} = 50 \text{ t/h}$; the removal ratio of contaminants in the treatment unit T_3 , $r_{T_3} = 95\%$; the investment

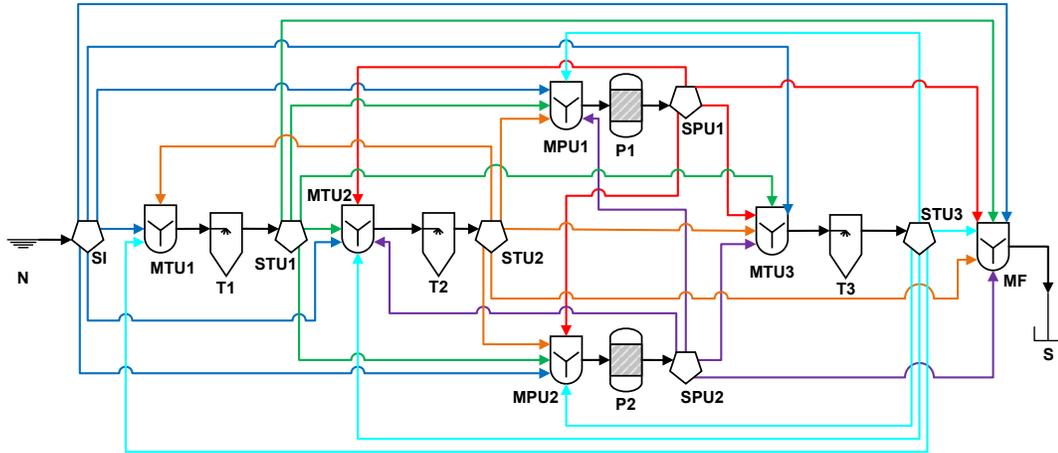


Figure 6: Superstructure of the case study. Black arrows: connections made by default. Other colored arrows: Possible connections. Own elaboration with information from Ahmetovic and Grossmann⁶⁰.

cost coefficient for treatment unit T_1 , $IC_{T_1} = 10,000$; the investment cost coefficient for treatment unit T_2 , $IC_{T_2} = 25,000$; and the investment cost coefficient for treatment unit T_3 , $IC_{T_3} = 30,000$.

With the nominal values of the parameters, the optimal solution has most binary variables with values equal to zero, meaning that non exist connections related to these variables. The only connections installed in the optimal layout are shown in Table 2 because binary variables are equal to one in the optimal solution. The representation of this solution in the network is observed in Figure 7. The goal of the analysis is to determine the region of the values of the mentioned parameters that allows obtaining the identical optimal solution for binary variables. In other words, no connections are included or removed from the optimal layout of Figure 7 by modifying the parameters within the region to be calculated.

Referential information preceding the algorithm execution allows the search to be bounded: Flows, costs, and removal ratio of contaminants cannot be negative. In particular, the removal ratio is between 0 and 100%. This information is an input to the algorithm and conditions the results below. Additionally, the original problem resolution with nominal parameters values took 1.71 seconds. This time is used as a referential to generate a time

Table 2: Connections installed in the optimal solution of the case study.

Variable	Description	Value
$y_{FP_{P_1,P_2}}$	Existence of connection between P_1 and P_2	1
$y_{FPT_{P_2,T_2}}$	Existence of connection between P_2 and T_2	1
$y_{FPT_{P_2,T_3}}$	Existence of connection between P_2 and T_3	1
$y_{FT_{T_3,T_2}}$	Existence of connection between T_3 and T_2	1
$y_{FPT_{T_2,P_1}}$	Existence of connection between T_2 and P_1	1
$y_{FPT_{T_3,P_2}}$	Existence of connection between T_3 and P_2	1

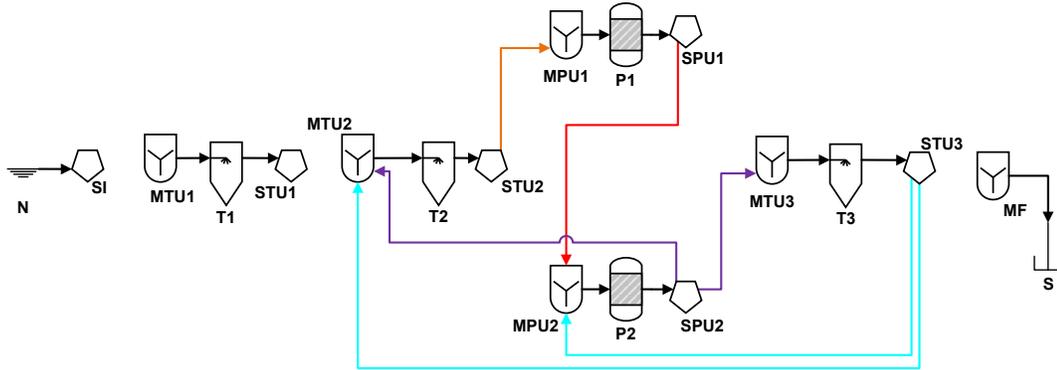


Figure 7: Original solution of the case study. Black arrows: connections made by default. Other colored arrows: connections decided by solving the optimization problem. Own elaboration with information from Ahmetovic and Grossmann⁶⁰.

limit for each new resolution (subproblems and iterations).

Algorithm application

The general procedure of the algorithm in Figure 1 is contextualized to the case study in Figure 8. The algorithm begins with selecting a parameter to study (r_{T_3}) and the variables of interest (Y , which represents the set of binary variables). Then, previous information about the bounds of parameters ($r_{T_3,ref}^{min}$ and $r_{T_3,ref}^{max}$) is incorporated; also, the search parameters are included: initial step (Δl), tolerance (t) and a counter of iterations (i). The search parameters come from the Yi and Lu³⁰ algorithm and are calculated as follows: the initial step (Δl) is the smallest number that has the same order of magnitude as the parameter under analysis; the tolerance (t) is 4/10000 times the nominal value of the parameter under

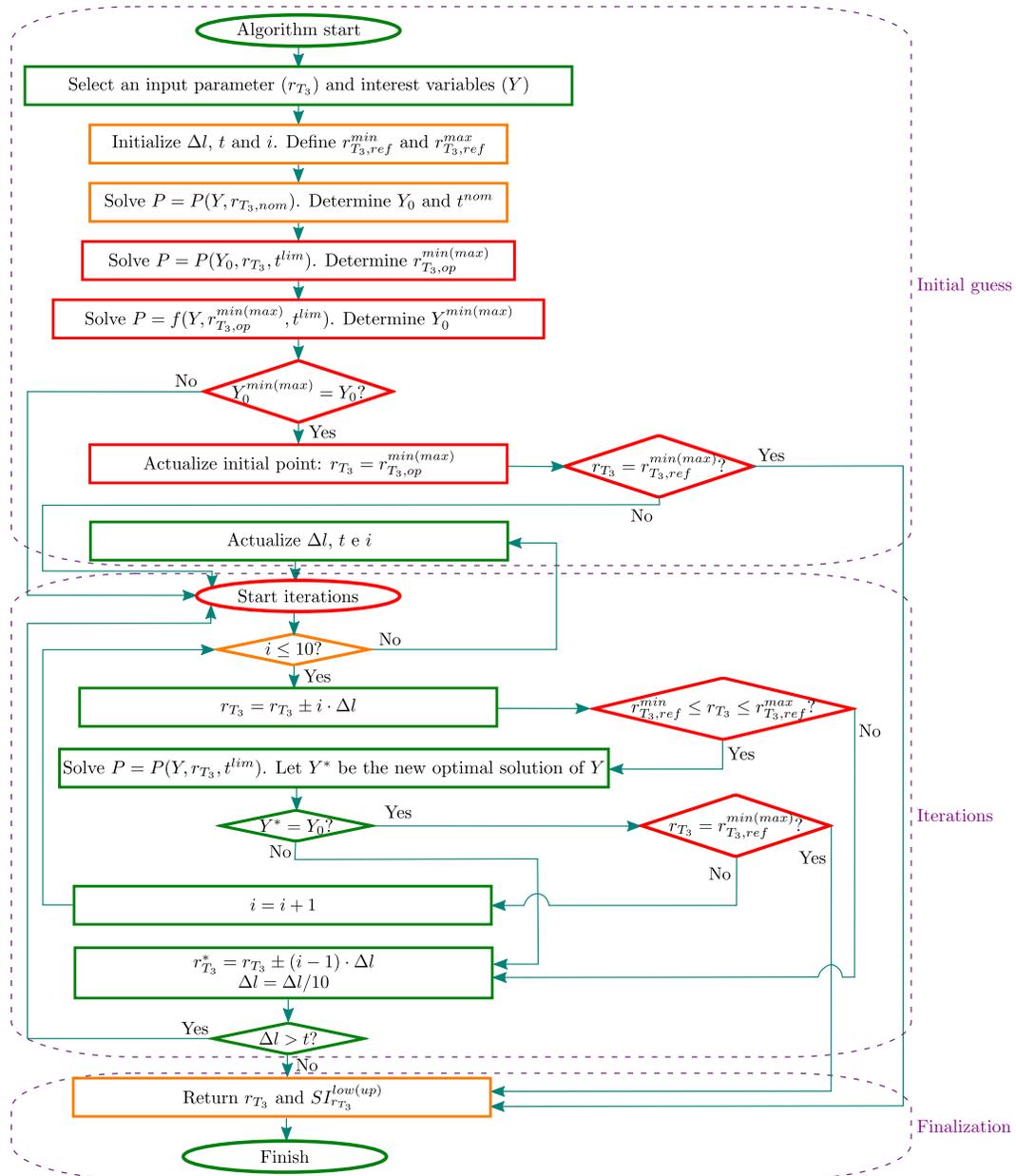


Figure 8: The proposed algorithm to obtain the stability region of parameters in MINLP problems applied for parameter r_{T_3} . Green boxes: algorithm steps maintained from Yi and Lu³⁰ algorithm. Orange boxes: Modified from Yi and Lu³⁰ algorithm. Red boxes: New steps proposed for MINLP version. Own elaboration.

study, and the authors obtained it from trial and error tests; the iteration counter(i) is initialized to 0. Subsequently, the problem is solved, and the original solution (Y_0) and the nominal time (t^{nom}) are obtained. This time allows for generating a time limit (t^{lim}) for each solution. The mentioned information and parameters are used to solve a subproblem of the original, and it is determined by the optimized value of the parameter ($r_{T_3,op}^{min(max)}$). This value is used to examine if the original solution of binary variables (Y_0) is maintained or changed.

If the optimized parameter value keeps the original value of binary variables ($Y^{min(max)} = Y_0$) and the optimized parameter is equal to some of the referential limits ($r_{T_3,ref}^{min}$ or $r_{T_3,ref}^{max}$), the evaluated value corresponds to the searched limit because the algorithm does not evaluate values out of the referential range. On the other hand, if the solution is maintained ($Y^{min(max)} = Y_0$) and the evaluated value is inside the referential range, the iterative process is initiated from the new initial guess until finding the limit for the tolerance (t) or reaching a referential limit. While not occurring in the mentioned situations, step (Δl) is updated as it happens in the Yi and Lu³⁰ algorithm, evaluating the conditions in each iteration. If no binary variables solutions change after ten iterations ($i = 10$), then the algorithm suggests actualizing the search parameters if it is necessary to continue looking for the limit. The iterative process is the same if the optimized parameter generates a different solution ($Y^{min(max)} \neq Y_0$), although it starts from the nominal value. The limit (lower or upper) of the stability region and its corresponding index ($SI_{rT_3}^{low(up)}$) is obtained when the search is finalized.

The above steps are detailed using the introductory and the illustrative examples as follows:

Initial guess

Previously, it was explained how it is proposed to obtain a new initial guess for both the lower and upper limit searched through a subproblem of the original problem. Maximization

and minimization of the subproblem allow knowing two values of interest: maximum ($r_{T_3,op}^{max}$) and minimum ($r_{T_3,op}^{min}$) values that can adopt the parameter with the original values of binary variables. In other words, overpassing those values in the original problem (without fixing binary variables) generates solutions with binary variables with different values in contrast to the original. Therefore, those values are candidates as limits of the stability region and options as initial guesses for the algorithm. These new starting points of the algorithm allow the search closer to each limit of the stability region and decrease required iterations.

The results of the subproblem are not always inside the stability region. The subproblem solution is related to the feasible region for the parameter, but it is not direct to ensure that they are inside the stability region. Therefore, a new resolution of the original problem with the parameter new value is required. The new starting point is accepted if the optimal solution maintains the binary variable values; otherwise, it is discarded.

The new algorithm includes a proposed time limit for each subproblem resolution and iteration, determined by equation (7). The time limit is defined by the time that took the original problem. This formulation allows considering the particularities of each model instead of defining a specific time for all the cases. The weight τ must be assigned by the user, according to the time and experience that the user has.

$$t^{lim} = \tau \cdot t^{nom} \quad (7)$$

Where:

- t^{lim} : Resolution time limit for each iteration of the algorithm.
- t^{nom} : Resolution time of the original problem with nominal values of parameters.
- τ : Weight to adjust the maximum resolution time for each iteration.

The subproblem of the case study was solved to determine a new initial guess. Then, those new values for the parameters were evaluated in the original problem to realize if the

binary variables had the same values. For each parameter, there are two evaluations because of the minimization and maximization subproblems that generate possible initial guesses for searching lower and upper limit, respectively.

A new initial guess is implemented for four parameters in the case study, as shown in Table 3. Discarded values exist because the proposed model⁶⁰ considers previous calculus containing parameter values. Thus, modifying the parameter involved more previous calculus, and some rounding errors affected the subproblem resolution. In particular, for the flows, that situation difficult the resolution and no solution was obtained in the time limit assigned (weight $\tau = 5$). Multiple resolutions of the original problem with different values of parameters show that solutions do not change for higher weights. For that reason, $\tau = 5$ is implemented for estimating a new initial guess and subsequently for each iteration.

Table 3: Feasibility of using the subproblem solution as an initial value of the new algorithm in search of the lower (Min. case) or upper (Max. case) limit based on a comparison of the solution of the binary variables obtained using the optimized value, and the nominal value in the original problem.

Parameter	Min: binary is maintained?	Max: binary is maintained?
$FPU_{P_1}^{in}$	No ^a	No ^a
$FPU_{P_2}^{in}$	No ^a	No ^a
r_{T_3}	Yes	No
IC_{T_1}	No	Yes
IC_{T_2}	Yes	No ^b
IC_{T_3}	Yes	No ^b

^a The maximum time for execution elapsed ($\tau = 5$, 8.6[s] approx.), and the search was interrupted without obtaining a solution.

^b Evaluating the optimized parameter in the original problem generates an infeasible solution.

Iterations

Iterative process steps of Figure 8 are applied in the illustrative example to determine the stability region of the six parameters. It is explained the procedure for parameter r_{T_3} to illustrate each step. The search for the lower limit of r_{T_3} starts in the new initial guess ($r_{T_3,op}^{min} = 80.8$) because it maintains the original solution of binary variables. The last value

evaluated that fulfills the condition of maintaining the value of the binary variables ($Y^* = Y_0$) in each loop is highlighted in the column $r_{T_3}^*$ of Table 4. The algorithm finishes in 80.8 because of achieving the tolerance (if the step is updated to 0.01, it would be minor than the tolerance of 0.038). In particular, this value is the lower limit, but the iterations are necessary to achieve the tolerance and conclude the searched limit. On the other hand, the algorithm discarded the initial guess obtained in the parameter optimization for searching the upper limit ($r_{T_3,op}^{max} = 100.0$) because of a change in binary variables solution. The search starts in the nominal value (95.0) and finishes in the value 97.4, as is shown in Table 5. The upper limit is concluded because of achieving the tolerance.

Table 4: Calculating the lower limit of the stability region of parameter r_{T_3} .

i	r_{T_3}	$Y^* = Y_0?$	$r_{T_3}^*$	Δl
Loop 1: $\Delta l = 10$				
0	80.8	Yes	–	–
1	70.8	No	80.8	$1.0 > 0.038$
Loop 2: $\Delta l = 1$				
0	80.8	Yes	–	–
1	79.8	No	80.8	$0.1 > 0.038$
Loop 3: $\Delta l = 0.1$				
0	80.8	Yes	–	–
1	80.7	No	80.8	$0.01 < 0.038$

The performance of the proposed algorithm is significantly better than the referential version. As shown in Table 6, most of the limits are obtained in less time with the proposed algorithm (A2) concerning Yi and Lu³⁰ version (A1). Specifically, the overall time has been reduced by 96.6%. The decreasing time is because of the best initial estimation from the parameter optimization and restricting the search only in a region bounded for previous information and for a limited time.

The determined regions are more precise with the proposed algorithm than the referential version. Some cases have the same limit, but others are different, highlighted in Table 6. The reasons for the differences are multiple: adding logical and physical bounds in the new algorithm allow to obtain more realistic results (for example, the costs must be

Table 5: Calculating the upper limit of the stability region of parameter r_{T_3} .

i	r_{T_3}	$Y^* = Y_0?$	$r_{T_3}^*$	Δl
Loop 1: $\Delta l = 10$				
0	95.0	Yes	—	—
1	105.0	No	95.0	$1.0 > 0.038$
Loop 2: $\Delta l = 1$				
0	95.0	Yes	—	—
1	96.0	Yes	—	—
2	97.0	Yes	—	—
3	98.0	No	97.0	$0.1 > 0.038$
Loop 3: $\Delta l = 0.1$				
0	97.0	Yes	—	—
1	97.1	Yes	—	—
2	97.2	Yes	—	—
3	97.3	Yes	—	—
4	97.4	Yes	—	—
5	97.5	No	97.4	$0.01 < 0.038$

Table 6: Comparison of the referential (A1) and proposed (A2) algorithm results.

Parameter	Limit	Results A1	Results A2	Time A1 [s]	Time A2 [s]	Variation of time [%]
$FPU_{P_1}^m [t/h]$	Lower	11.5	11.5	5,196 ^a	121	-97.7
$FPU_{P_1}^m [t/h]$	Upper	49.0^b	49.9	2,063 ^a	56	-97.3
$FPU_{P_2}^m [t/h]$	Lower	41.0^b	40.1	1,741 ^a	47	-97.3
$FPU_{P_2}^m [t/h]$	Upper	55.3	55.3	43	43	0.1
$r_{T_3} [\%]$	Lower	80.8	80.8	124	27	-78.4
$r_{T_3} [\%]$	Upper	97.4	97.4	22	31	43.4
$IC_{T_1} []^c$	Lower	4,150	4,150	1,131 ^a	78	-93.1
$IC_{T_1} []^c$	Upper	100,000^b	NB^d	2,042 ^a	8	-99.6
$IC_{T_2} []^c$	Lower	-65,000^b	0.0	1,027 ^a	2	-99.8
$IC_{T_2} []^c$	Upper	115,000^b	125,000^b	13	21	64.6
$IC_{T_3} []^c$	Lower	-60,000^b	0.0	13	1	-90.8
$IC_{T_3} []^c$	Upper	120,000^b	130,000^b	19	27	38.2

^a The maximum execution time (1000[s] by default) was reached for some iterations, and the search was interrupted, obtaining the best possible solution in that time.

^b Last value evaluated by the algorithm. It is suggested to update the search parameters to continue searching for the range limit if necessary.

^c Unit treatment investment cost, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, has units in [\\$].⁶⁰

^d NB: Non-bounded. There is not a bound or that value is over than 10^{50} .

positive values); the initial guess obtained from the subproblem recognize that IC_{T_1} is not bounded superiorly; finally, the inclusion of an additional iteration ($i \leq 10$, in contrast to the referential algorithm ($i < 10$)) allows automatizing the search and it finishes by tolerance for $FPU_{P_1}^{in}$ and $FPU_{P_2}^{in}$ obtaining more precise results. A disadvantage of the extra iteration could appear when no changes of binary variables are detected, spending more time like in the upper limit of IC_{T_2} and IC_{T_3} . However, the contrast of the overall results and time show that the proposed algorithm is more precise and efficient for this problem. It is relevant to highlight that those results are expectable because the referential algorithm was created for MILP problems, but it is the unique point of comparison.

The values calculated previously are used to determine the stability index in the next section.

Finalization

The algorithm finalizes calculating the proposed stability index. This index allows quantifying the risk of the binary variables solution change. Therefore, it is possible to prioritize parameters and assumptions critical to maintaining the optimal solution of binary variables. The index calculus is based on the distance of the nominal value concerning the limits of the stability region. Additionally, it is suggested to include a relative normalized difference to compare parameters with different orders of magnitude. It is an advantage, in contrast, to use the stability region directly because it does not incorporate the nominal value. That propose is represented with equation (8), and it must be implemented for each limit of the stability region of parameters.

$$SI_{r_{T_3}}^{low(up)} = \left| \frac{r_{T_3}^{nom} - r_{T_3}^{low(up)}}{r_{T_3}^{nom}} \right| \cdot 100[\%] \quad (8)$$

Where:

- $r_{T_3}^{nom}$: Nominal value of the parameter r_{T_3} .

- $r_{T_3}^{low(up)}$: Lower (upper) limit of the stability region of the parameter r_{T_3} .
- $SI_{r_{T_3}}^{low(up)}$: Lower (upper) stability index, which represents the normalized relative difference between the lower (upper) limit of the stability region of r_{T_3} and its nominal value.

To keep the original value of the binary variables is riskier when the chosen value (nominal) is closer than the limit of the stability region. In that case, an error in estimating the nominal value could change the optimal solution. The proximity to any limits is a risk, so the stability index (SI) must be established on the closer limit. Therefore, it is suggested to calculate the stability index from the expression (9) related to the minimal difference concerning one of the limits. A parameter with a higher value of SI has more tolerance against estimation error, so the critics are those with a lower SI .

$$SI_{r_{T_3}} = \min\left(SI_{r_{T_3}}^{low}, SI_{r_{T_3}}^{up}\right) \quad (9)$$

Values of stability region determined by the proposed algorithm from Table 6 are used in equations (8) and (9) to calculate the stability index of each parameter, and the results are shown in Table 7. In the case study, the parameter that needs to be studied in detail is the removal ratio of contaminants r_{T_3} , because of the nearness of the nominal value to the upper limit. Other particularity is that IC_{T_1} is not bounded superiorly. When a parameter is not bounded (lower or upper), it is unnecessary to determine the stability index related to this specific limit because it can change its value without modifying the solution.

In two cases (IC_{T_2} and IC_{T_3}), ten iterations were implemented without changing binary variables. For these cases, the algorithm suggests updating the search parameters if it is necessary to continue looking for the upper limit of the region. That iteration limit exists to avoid iterating indefinitely. Despite not recognizing a limit for those ten iterations, it is not feasible to state no changes in the binary variables for higher values. Therefore, it cannot be assured that the range is not bounded. For example, if the search of the upper limit of IC_{T_3}

Table 7: Stability index of parameters of the case study.

Parameter	Nominal value	Stability region	$SI^{low}[\%]$	$SI^{up}[\%]$	$SI[\%]$
$FPU_{P_1}^{in}[t/h]$	40	$11.5 \leq FPU_{P_1}^{in} \leq 49.9$	71.3	24.8	24.8
$FPU_{P_2}^{in}[t/h]$	50	$40.1 \leq FPU_{P_2}^{in} \leq 55.3$	19.8	10.6	10.6
$r_{T_3} [\%]$	95	$80.8 \leq r_{T_3} \leq 97.4$	14.9	2.5	2.5
$IC_{T_1} []^a$	10,000	$4,150 \leq IC_{T_1}$	58.5	ND ^b	58.5
$IC_{T_2} []^a$	25,000	$0 \leq IC_{T_2} \leq 125,000^c$	100.0	400.0	100.0
$IC_{T_3} []^a$	30,000	$0 \leq IC_{T_3} \leq 130,000^c$	100.0	333.0	100.0

^a Unit treatment investment cost, $AR \cdot IC_t \cdot (FTU_t^{out})^\alpha$, has units in [\\$].⁶⁰

^b ND: Non determined because it is non-bounded.

^c That is the last value evaluated by the algorithm, and it is suggested to update the search parameters.

had continued with the same step, it would have been recognized that the binary variables change when $IC_{T_3} = 140,000$ is evaluated.

Discussion

The proposed algorithm corresponds to a first approach for finding the stability region of parameters in MINLP problems. It was compared with the closest work corresponding to MILP cases. Furthermore, since the proposed algorithm exists, it would be used to reference further advances in the post-optimal analysis of MINLP problems.

The time spent by the proposed algorithm is 96.6% less than the referential one in the case study. This decrease was expected because the referential algorithm was created for MILP cases. Despite this, the proposed additional steps reduce the iterations and time required to determine the stability region of parameters. Some of the new steps and ideas are discussed as follows:

The choice of time limits affects the solution in two ways: if the time is too short, the best solution found could not be the optimum solution, and if the time is too long, it takes longer to obtain a solution. Therefore, it is recommended not to use a τ equal to 1 because it is not insurable that each iteration takes the same time as the original problem, being necessary to include a margin of error. On the contrary, it would be prudent to consider

times with similar orders of magnitude, possible with a τ value less than 10. In particular, for the examples studied, the use of a τ equal to 5 allowed reducing the total time of the algorithm and did not affect the solution. Furthermore, limiting the time allows spending an inferior order of magnitude of the time in the proposed algorithm in contrast to the referential version that required over 1,000 seconds for half of the calculated limits.

The case study showed some possible challenging situations for the proposed algorithm. For example, if there are problems with previous calculus with the interest parameter involved, subproblems could have difficulties because of rounding errors. Also, implementing subproblem takes time that affects majorly when no change in parameters is detected until achieving the limit of iterations. In those cases, the proposed algorithm required a longer time than the referential version. However, the extra time spent was compensated by reducing the time in most parameters with proposed improvements implemented in the new algorithm.

For those cases where the maximum value allowed in GAMS ($\pm 10^{50}$) was reached, it was considered that they were not bounded (lower or higher) because it is not possible to continue iterating for larger values with the same procedure. Nevertheless, a possibility to continue is to change the units of the problem expressions to reach larger values. However, reaching numbers with those significant figures is not often necessary. The proposed version allows recognizing that the upper limit of the parameter IC_{T_1} has that behavior in 8 seconds, in contrast to the referential version that spent over 2,000 seconds without realizing that no changes exist for higher values.

The stability range search is performed only within values of interest. This limitation allows the algorithm to avoid evaluating cases where its results do not agree with reality. For example, in the case study, negative costs could have been evaluated, which were eliminated by the improvements of the proposed algorithm, reducing the iterations required in its implementation.

Referential limits are not only related to physical and logical information. Also can be incorporated a value as a limit if the user is interested in the behavior of binary variables

in a specific region of values of parameters. Therefore, the algorithm requires no iterations when the parameters surpass those values if the user considers that those are never achieved. It is highlighted that the referential limits generate tighter regions in contrast with the possible mathematical region, although they are necessary because of the practical focus of the proposed algorithm. Specifically, implementing those limits in the new algorithm reduces the total iterations by avoiding evaluating values that are not in the interest or are not feasible. That contextualization also could be implemented in the referential algorithm for studying MILP cases linked with reality.

Starting iterations at an efficient starting point reduces iterations and implementation time. Comparing the results of the two algorithms (referential and proposed) shows that an initial guess obtained by the subproblem reduces the total implementation time. Another methodology could be evaluated to find initial values of better performance, although the analysis performed with the proposed one fulfills the objective of reducing the implementation time.

Other types of variables could be analyzed to maintain their original values. However, this could generate tight stability regions due to an eventual higher facility to change their value, contrasting with discrete variables. The context and the relationship with installation decisions generated the most interest in binary variables, so the context could also indicate whether it is required to keep any continuous variable.

The work is focused on making a first approximation for changes of one parameter at a time. Possible evolution of the research is to analyze a global sensitivity case. Initially, the effect of more than one simultaneously could be performed iteratively. For example, in Figures 2 and 3, the upper limit of a and the lower limit of b are determined by the intersection of constraints ($x + y \leq b$ and $a^2 - x - y \leq 0$). Therefore, increasing b will allow a to increase until the constraints intersect. Conversely, a decrease in a will also allow the lower bound of b to decrease. However, it is essential to consider the objective function because it conditions the lower limit. For example, it was shown in Figure 3.d that values for

a less than 0.25 generate that it is more convenient to have a different solution ($y = 0$). How to efficiently determine the stability range for simultaneous changes in parameters is beyond the scope of the work, although it is highlighted as a possible evolution of the research. The challenge could be to recognize a possible rule that generates a range of stability of one parameter dependent on the other.

Conclusion

This paper proposes an algorithm to find the stability region of parameters in MINLP problems. No other works have been developed for this type of problem. The algorithm was created with an iterative procedure based on a referential one for MILP developed by Yi and Lu³⁰, the closest work to find the stability region of parameters.

The new version has some improvements in contrast to the referential version to do efficient the search of stability region for MINLP cases: (i) the initial point is obtaining solving a subproblem of the original; (ii) the search is bounded to possible values in reality, considering the context of each problem; (iii) a limit time for each iteration is included, and it is calculated using the original resolution time of the problem; (iv) finally, a stability index is proposed to compare stability regions of parameters.

Both algorithms were implemented in a case study, and the new proposed steps allow reducing the algorithm time by 96.6%. The reasons for that reduction are: (i) the search started in an initial guess closest to the searched limit because a previous calculus was made before the iterations; (ii) the iterations were made only in possible values because of the implementation of referential information; (iii) some values generates higher time of resolution, and the proposed algorithm limited that time, obtaining the stability region in less time.

The proposed stability index (SI) allows recognizing parameters with a higher risk of changing the optimal solution if their estimation has errors. This index was formulated using

the nominal value and the relative distance to the limits of the stability region. Furthermore, the parameter that has a lower SI needs to be studied in-depth to ensure that the solution obtained is optimal. For example, strategic decisions are made for the long-term and are hard to modify, being necessary to establish that the solution could not change because of wrong estimations.

The case study was water network design, and the critical parameter was recognized. Also, the algorithm could be implemented in other MINLP problems, especially for process engineering problems, because parameters have uncertainty in their estimation. Each context is relevant to deciding what variables must be maintained, but the binary variables are usually more attractive because of their relation with installation decisions.

A possible extension of the algorithm could consider that the current version has difficulties estimating new initial guesses in cases with previous calculus requiring the parameter value. Other kind subproblems could be formulated to improve the initial guess or include previous information to establish a better initial value for the search.

Acknowledgement

Fernando Arenas-Araya: This work was funded by the National Agency for Research and Development (ANID) / Scholarship Program / Doctorado Nacional /2018 - 21181490

Supporting Information Available

The nomenclature of the algorithm and the examples can be revised in the Supporting Information file. This information is available free of charge via the Internet at <http://pubs.acs.org/>.

References

- (1) Shaw, M. *Engineering Problem Solving: A Classical Perspective*, 1st ed.; 2001; p 450.
- (2) Floudas, C. A. *Nonlinear and Mixed-Integer Optimization*; Oxford University Press: New York, 1995; p 462.
- (3) Jezowski, J.; Shethna, H.; Castillo, F. Area target for heat exchanger networks using linear programming. *Industrial and Engineering Chemistry Research* **2003**, *42*, 1723–1730.
- (4) Saffer, D.; Doyle, F. Efficient linear programming formulations of model predictive control for cross-directional control of a paper machine. *Industrial and Engineering Chemistry Research* **2005**, *44*, 2134–2144.
- (5) Schack, D.; Rihko-Struckmann, L.; Sundmacher, K. Linear Programming Approach for Structure Optimization of Renewable-to-Chemicals (R2Chem) Production Networks. *Industrial & Engineering Chemistry Research* **2018**, *57*, 9889–9902.
- (6) Poku, M.; Biegler, L.; Kelly, J. Nonlinear optimization with many degrees of freedom in process engineering. *Industrial and Engineering Chemistry Research* **2004**, *43*, 6803–6812.
- (7) Alvarães, A. d. O.; Prata, D. M.; Santos, L. d. S. Simulation and optimization of a continuous biodiesel plant using nonlinear programming. *Energy* **2019**, *189*, 116305.
- (8) Kralj, A. K.; Glavič, P. CO₂ separation from purge gas and flue gas in the methanol process, using NLP model optimization. *Industrial and Engineering Chemistry Research* **2007**, *46*, 6953–6962.
- (9) Biegler, L. T.; Grossmann, I. E. Retrospective on optimization. *Computers & Chemical Engineering* **2004**, *28*, 1169–1192.

- (10) Brunaud, B.; Grossmann, I. E. Perspectives in multilevel decision-making in the process industry. *Frontiers of Engineering Management* **2017**, *4*, 256.
- (11) Balaman, S. *Decision-Making for Biomass-Based Production Chains*; 2019; pp 143–183.
- (12) Bertoni, M. Introducing sustainability in value models to support design decision making: A systematic review. *Sustainability* **2017**, *9*.
- (13) Acevedo, J.; Pistikopoulos, E. A Multiparametric Programming Approach for Linear Process Engineering Problems under Uncertainty. *Industrial and Engineering Chemistry Research* **1997**, *36*, 717–728.
- (14) Terrazas-Moreno, S.; Grossmann, I. E.; Wassick, J. M. A mixed-integer linear programming model for optimizing the scheduling and assignment of tank farm operations. *Industrial and Engineering Chemistry Research* **2012**, *51*, 6441–6454.
- (15) Díaz-Gómez, J. C.; Caballero, J. A.; Jiménez-Gutiérrez, A. A Mixed-Integer Linear Programming Model for the Design of Shale Gas Industrial Parks. *ACS Sustainable Chemistry and Engineering* **2021**, *9*, 8783–8796.
- (16) Stebel, S. L.; Magatão, S. N.; Arruda, L. V.; Neves-Jr., F.; Pova, A. P.; Relvas, S. Mixed integer linear programming formulation for aiding planning activities in a complex pipeline network. *Industrial and Engineering Chemistry Research* **2012**, *51*, 11417–11433.
- (17) Kocis, G.; Grossmann, I. Global Optimization of Nonconvex Mixed-Integer Nonlinear Programming (Minlp) Problems in Process Synthesis. *Industrial and Engineering Chemistry Research* **1988**, *27*, 1407–1421.
- (18) Castillo, P. A. C.; Castro, P. M.; Mahalec, V. Global Optimization of Nonlinear Blend-Scheduling Problems. *Engineering* **2017**, *3*, 188–201.

- (19) Stichlmair, J.; Frey, T. Mixed-integer nonlinear programming optimization of reactive distillation processes. *Industrial and Engineering Chemistry Research* **2001**, *40*, 5978–5982.
- (20) Ciric, A. R.; Floudas, C. A. A Mixed Integer Nonlinear Programming Model for Retrofitting Heat-Exchanger Networks. *Industrial and Engineering Chemistry Research* **1990**, *29*, 239–251.
- (21) Szitkai, Z.; Farkas, T.; Lelkes, Z.; Rev, E.; Fonyo, Z.; Kravanja, Z. Fairly linear mixed integer nonlinear programming model for the synthesis of mass exchange networks. *Industrial and Engineering Chemistry Research* **2006**, *45*, 236–244.
- (22) Grossmann, I. E.; Trespalacios, F. Systematic modeling of discrete-continuous optimization models through generalized disjunctive programming. *AIChE Journal* **2013**, *59*, 3276–3295.
- (23) Wechsung, A.; Oldenburg, J.; Yu, J.; Polt, A. Supporting chemical process design under uncertainty. *Brazilian Journal of Chemical Engineering* **2010**, *27*, 451–460.
- (24) Pennock, M.; Wade, J. The top 10 illusions of systems engineering: A research agenda. *Procedia Computer Science* **2015**, *44*, 147–154.
- (25) Rios Insua, D.; French, S. A framework for sensitivity analysis in discrete multi-objective decision-making. *European Journal of Operational Research* **1991**, *54*, 176–190.
- (26) Ferretti, F.; Saltelli, A.; Tarantola, S. Trends in sensitivity analysis practice in the last decade. *Science of the Total Environment* **2016**, *568*, 666–670.
- (27) Iooss, B.; Lemaitre, P. Uncertainty Management in Simulation-Optimization of Complex Systems. *Operations Research/ Computer Science Interfaces Series* **2015**, *59*, 101–122.

- (28) Wagener, T.; Pianosi, F. What has Global Sensitivity Analysis ever done for us? A systematic review to support scientific advancement and to inform policy-making in earth system modelling. *Earth-Science Reviews* **2019**, *194*, 1–18.
- (29) Gransberg, D. Optimizing Haul Unit Size and Number Based on Loading Facility Characteristics. *Journal of Construction Engineering and Management* **1996**, *122*, 248–253.
- (30) Yi, C.; Lu, M. Mixed-Integer Linear Programming-Based Sensitivity Analysis in Optimization of Temporary Haul Road Layout Design for Earthmoving Operations. *Journal of Computing in Civil Engineering* **2019**, *33*, 1–14.
- (31) Jansen, B.; de Jong, J.; Roos, C.; Terlaky, T. Sensitivity analysis in linear programming: just be careful! *European Journal of Operational Research* **1997**, *101*, 15–28.
- (32) Fiacco, A.; Ishizuka, Y. Sensitivity and stability analysis for nonlinear programming. *Annals of Operations Research* **1990**, *27*, 215–236.
- (33) Büskens, C.; Maurer, H. *Online Optimization of Large Scale Systems*; Springer Berlin Heidelberg: Berlin, Heidelberg, 2001; pp 3–16.
- (34) Sahinidis, N. V. Mixed-integer nonlinear programming 2018. *Optimization and Engineering* **2019**, *20*, 301–306.
- (35) Latifi, S.; Mohammadi, E.; Khakzad, N. Process plant layout optimization with uncertainty and considering risk. *Computers and Chemical Engineering* **2017**, *106*, 224–242.
- (36) Faria, C.; Bagajewicz, M. Pyrolysis of Heavy Oil in the Presence of Supercritical Water: The Reaction Kinetics in Different Phases. *AIChE Journal* **2015**, *61*, 857–866.
- (37) Lienqueo, M. E.; Salgado, J. C.; Giaverini, O.; Asenjo, J. A. Computer-aided design

- to select optimal polypeptide tags to assist the purification of recombinant proteins. *Separation and Purification Technology* **2009**, *65*, 86–94.
- (38) Bakshi, B. The Quest for Sustainability : Challenges for Process Systems Engineering. *AIChE Journal* **2003**, *49*, 1350 – 1358.
- (39) Solomon, S.; Qin, D.; Manning, M.; Chen, Z.; Marquis, M.; Averyt, K. *Climate Change 2007 : The Physical*; Cambridge University Press: Cambridge, 2007.
- (40) Misra, A. Climate change and challenges of water and food security. *International Journal of Sustainable Built Environment* **2014**, *3*, 153–165.
- (41) Note, M. *Project Management for Information Professionals*; Chandos Publishing, 2016; Chapter 2, pp 19–39.
- (42) Ursache, M. Sustainable development between recent experiences and future challenges. *Procedia Economics and Finance* **2014**, *15*, 1316–1323.
- (43) Zhang, X.; Liu, J.; Zhao, X.; Yang, H.; Deng, X.; Jiang, X.; Li, Y. Linking physical water consumption with virtual water consumption: Methodology, application and implications. *Journal of Cleaner Production* **2019**, *228*, 1206–1217.
- (44) Nikjoofar, A.; Zarghami, M. Water Distribution Networks Designing by the Multiobjective Genetic Algorithm and Game Theory. *Metaheuristics in Water, Geotechnical and Transport Engineering* **2013**, 99–119.
- (45) Bagajewicz, M. A review of recent design procedures for water networks in refineries and process plants. *Computers and Chemical Engineering* **2000**, *24*, 2093–2113.
- (46) Foo, D. State-of-the-art review of pinch analysis techniques for Water network synthesis. *Industrial and Engineering Chemistry Research* **2009**, *48*, 5125–5159.
- (47) Jezowski, J. Review of Water Network Design Methods with Literature Annotations. *Ind. Eng. Chem. Res.* **2010**, *49*, 4475–4516.

- (48) Khor, C.; Chachuat, B.; Shah, N. Optimization of water network synthesis for single-site and continuous processes: Milestones, challenges, and future directions. *Industrial and Engineering Chemistry Research* **2014**, *53*, 10257–10275.
- (49) Altherr, L. C.; Leise, P.; Pfetsch, M. E.; Schmitt, A. Resilient layout, design and operation of energy-efficient water distribution networks for high-rise buildings using MINLP. *Optimization and Engineering* **2019**, *20*, 605–645.
- (50) Kralj, A. K. Industrial wastewater collection using a separation technique. *Journal of Industrial and Engineering Chemistry* **2012**, *18*, 1320–1325.
- (51) Jiménez-Gutiérrez, A.; Lona-Ramírez, J.; Ponce-Ortega, J. M.; El-Halwagi, M. An MINLP model for the simultaneous integration of energy, mass and properties in water networks. *Computers and Chemical Engineering* **2014**, *71*, 52–66.
- (52) Pungthong, K.; Siemanond, K. *Computer Aided Chemical Engineering*; Elsevier, 2015; Vol. 37; pp 1319–1324.
- (53) GAMS Development Corporation, General Algebraic Modeling System (GAMS) Release 28.2.0. 2019.
- (54) Sahinidis, N. BARON: A general purpose global optimization software package. *Journal of Global Optimization* **1996**, *8*, 201–205.
- (55) Python Software Foundation, Python Language Reference, Release 3.6.8. 2018.
- (56) Caballero, J.; Ravagnani, M. Water distribution networks optimization considering unknown flow directions and pipe diameters. *Computers and Chemical Engineering* **2019**, *127*, 41–48.
- (57) Prieto, D.; Swinnen, N.; Blanco, L.; Hermosilla, D.; Cauwenberg, P.; Blanco, Á.; Negro, C. Drivers and economic aspects for the implementation of advanced wastewater

- treatment and water reuse in a PVC plant. *Water Resources and Industry* **2016**, *14*, 26–30.
- (58) Böhmelt, T.; Bernauer, T.; Buhaug, H.; Gleditsch, N.; Tribaldos, T.; Wischnath, G. Demand, supply, and restraint: Determinants of domestic water conflict and cooperation. *Global Environmental Change* **2014**, *29*, 337–348.
- (59) Eck, B.; Mevissen, M. Valve Placement in Water Networks: Mixed-Integer Non-Linear Optimization with Quadratic Pipe Friction Valve Placement in Water Networks: Mixed-Integer Non-Linear Optimization with Quadratic Pipe Friction. *IBM Research Report* **2012**, *25307*.
- (60) Ahmetovic, E.; Grossmann, I. Global Superstructure Optimization for the Design of Integrated Process Water Networks. *AIChE Journal* **2010**, *57*, 434–457.