



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA CIVIL INDUSTRIAL

HEURÍSTICA BASADA EN COVERING Y GENERACIÓN DE COLUMNAS
DINÁMICO PARA EL PROBLEMA DE RUTEO DINÁMICO DE VEHÍCULOS CON
VENTANAS DE TIEMPO

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN GESTIÓN DE
OPERACIONES

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL

JOSÉ ANTONIO ROJAS MEJÍAS

PROFESOR GUÍA:
CRISTIÁN CORTÉS CARILLO

MIEMBROS DE LA COMISIÓN:
ANDRÉS WEINTRAUB POHORILLE
MICHEL GENDREAU
FERNANDO ORDOÑEZ

SANTIAGO, CHILE
DICIEMBRE 2012

RESUMEN DE LA TESIS
PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL INDUSTRIAL
POR: JOSÉ ROJAS
FECHA: 06/12/2012
PROF. CRISTIÁN CORTÉS CARRILLO

HEURÍSTICA BASADA EN COVERING Y GENERACIÓN DE COLUMNAS DINÁMICO PARA EL PROBLEMA DE RUTEO DINÁMICO DE VEHÍCULOS CON VENTANAS DE TIEMPO

En la presente Tesis, se propone una metodología para enfrentar el Problema Dinámico de Ruteo de Técnicos con Ventanas de Tiempo. La naturaleza dinámica del problema está comprendida en los clientes que aparecen dinámicamente y deben ser asignados a la mejor ruta posible mediante algún criterio. La metodología propuesta puede ser resumida en tres pasos:

- Usar información histórica para crear nodos ficticios, los que sirven como puntos de espera (y de covering) para los vehículos que han terminado sus tareas en agenda.
- Resolver el problema de ruteo estático, que comprende los nodos reales y los puntos de espera, mediante Generación de Columnas.
- Añadir las llamadas entrantes usando la heurística Generación de Columnas Dinámica.

La Generación de Columnas Dinámica es una heurística que se basa en la propuesta de Chen[6], la cual genera nuevas columnas usando la información de los nuevos nodos junto con un subconjunto de las columnas generadas anteriormente en la resolución del problema estático. Mientras se generan nuevas columnas, estas pueden ser añadidas al conjunto total de columnas y resolver el nuevo problema de optimización. Esta heurística permite encontrar nuevas soluciones muy rápido, por lo cual corresponde a una buena opción para resolver problemas dinámicos donde las decisiones son tomadas en tiempo real.

En cuanto a los resultados numéricos, el método propuesto (SCTTP-DCG), obtiene mejores resultados que la implementación dinámica más intuitiva realizable: el ruteo estático de nodos más la agregación Greedy de llamados dinámicos (TTP-Greedy). Por otra parte, el ruteo estático de nodos con la agregación de nodos dinámicos por medio de Generación de Columnas Dinámico (TTP-DCG) obtiene mejores resultados en comparación SCTTP-DCG. La explicación a este fenómeno tiene que ver con dos efectos: TTP-DCG gana las bondades de la inserción por medio de DCG y, además, ahorra en tiempos de desplazamiento pues no tiene que viajar a los nodos ficticios. Si bien era lógico que este fenómeno existiera, se esperaba que el SCTTP-DCG lograra recuperar los costos extra de transporte por medio de menor desplazamiento gracias a la buena ubicación frente a los llamados venideros.

Agradecimientos

Después de tanto tiempo transcurrido y de tanta gente conocida, es muy difícil mirar hacia atrás y seleccionar a quienes agradecer sin sentir que alguien se queda en el tintero. Espero que todos entiendan que en este largo proceso muchos fueron importantes para mí en mayor o menor medida, y que no estén explícitamente en esta hoja no significa que no les agradezca el apoyo que en algún momento me dieron y que me permiten escribir hoy estas palabras.

En primer lugar, quiero agradecer a mi familia quienes me formaron como persona, fomentaron siempre mi independencia y espíritu crítico y me permitieron ser feliz aún en los momentos más difíciles. En especial, agradezco a mis padres José y Elena, a mis hermanas Pamela y Tamara, y a mi abuela Virginia.

En segundo lugar, quiero agradecer a mis amigos de vida, esos que siempre están en mis recuerdos y a los cuales agradezco cada consejo, cada risa y cada gesto que en algún momento tuvieron conmigo. En particular, agradezco a Nicolás S., Cristóbal M., Eduardo B., Manuel A. y todos los T-chelmos. Y en de forma muy especial a Fabrizio B. y Carlos P.

En tercer lugar, agradezco a mis amigos de Universidad, quienes hicieron de Beauchef, a pesar de la exigencia, un lugar ameno e infinitamente interesante. Muchas gracias para Anita G., Viviana G., Gabriel E., Jaqueline VH., Olivia A., Fernando S., María Paz J., Ricardo G., Cinthya V. y Charles T. Además, quiero mencionar de forma especial a la División de Transporte, quienes me acogieron como uno más, en especial a Jaime O. y Francisco C.

En cuarto lugar, quiero agradecer a todos aquellos que hicieron posible mi estadía en Montreal, en especial a Louis-Martin Rousseau y Michel Gendreau, por guiarme y alentarme en mi trabajo, a Claudio Contardo, Clara Almonacid y Thomas Capelle, por acogermme y acompañarme en el proceso, y a todo el equipo del MGO y del DII, en especial a Julie Lagos, Fernanda Melis y Maritza Arancibia.

Quiero agradecer en forma muy especial a los profesores Cristián Cortés y Andrés Weintraub, quienes me aconsejaron, corrigieron y ayudaron durante mi período de tesista, todo con el objetivo de formarme como un buen ingeniero y enseñarme que una palabra sabia puede soltar el más duro de los nudos.

Finalmente, quiero recordar a mi gorda querida, María Jesús, que no sólo abrió todas las puertas que me permiten el día de hoy ser ingeniero, sino que me ayudó a madurar y ser una mejor persona. Se, con toda seguridad, que nos cuidas y acompañas, y lo seguirás haciendo hasta que nos veamos algún día en el cielo.

José Rojas Mejías.
Octubre, 2012

Índice general

Resumen	I
Agradecimientos	II
Índice general	III
Índice de tablas	VI
Índice de figuras	VII
1. INTRODUCCIÓN	1
1.1. Motivación	2
1.2. Objetivos	2
1.2.1. Objetivo General	2
1.2.2. Objetivos Específicos	3
1.3. Estructura del Trabajo	3
2. REVISIÓN BIBLIOGRÁFICA	4
3. HEURÍSTICA PROPUESTA	7
3.1. Heurística Propuesta	7
3.2. Detalle de Modelos	9
3.2.1. Clasificación	9
3.2.2. Problema de Ruteo de Técnicos con Ventanas de Tiempo	13
3.2.3. Descomposición para Generación de Columnas. Formulación sin decisión en tiempos de espera	16

3.2.4. Generación de Columnas Dinámico	18
4. PROBLEMA A RESOLVER	22
4.1. Descripción del Problema	22
4.2. Zona de Estudio	23
4.3. Estudio de Demanda y Estadísticas de la Zona	26
5. RESULTADOS NUMÉRICOS	32
5.1. Implementación	32
5.2. Política de Creación de Instancias	33
5.3. Corridas Preliminares	34
5.3.1. Función Objetivo	35
5.3.2. Nodos Ficticios	37
5.3.3. Generación de Columnas Dinámico	37
5.4. Resultados Numéricos	37
5.4.1. Instancias Ejecutadas	37
5.4.2. Heurísticas de Comparación	38
5.4.3. Resultados	38
6. PROBLEMA DE RUTEO DE TÉCNICOS CON COVERING TIEMPO ESPACIAL	44
6.1. Ruteo de Técnicos con Covering Tiempo Espacial	44
6.2. Formulación	46
6.2.1. Problema de Programación Lineal	46
6.2.2. Descomposición para Generación de Columnas	49
6.3. Resultados	51
7. CONCLUSIONES	52
7.1. Conclusiones Generales	52
7.2. Trabajo Futuro	53
REFERENCIAS	55

Apéndice A. ALGORITMOS	59
A.1. Modelo de programación	59
A.2. Tablas de Datos	59
A.3. Tablas de Resultados	60
A.4. Pseudocódigos para Algoritmo de Inserción Simple	60
A.4.1. Definiciones	60
A.4.2. Algoritmos	61

Índice de tablas

4.1.	Comunas de Santiago Sur-Oriente con sus habitantes y su superficie urbana.	24
4.2.	Zonas definitivas de trabajo, Santiago Sur-Oriente.	27
4.3.	Llamados totales anuales por mes y porcentaje.	28
4.4.	Llamados totales anuales por día de la semana, porcentaje y varianza.	28
4.5.	Estadística de llamados por tipo de impresora.	29
4.6.	Numero de técnicos a utilizar por mes.	30
4.7.	Distribución de Comunas en áreas de trabajo.	31
5.1.	Promedio de Costos de Instancias Corridas.	39
A.1.	Llamados totales anuales por mes.	63
A.2.	Llamados totales anuales por hora.	64
A.3.	Llamados por día de la semana por zona.	65

Índice de figuras

3.1.	Algoritmo Propuesto.	8
4.1.	Zona Sur-Oriente de Santiago.	25
4.2.	Nodos representantes de las 49 zonas a utilizar.	26
5.1.	Ejemplo de influencia de coeficientes en Función Objetivo	36
5.2.	Brecha de costos en función del número de nodos.	40
5.3.	Comparación de costos totales TTP-DCG vs TTP-Greedy.	41
5.4.	Brecha de costos en función del número de nodos.	42
5.5.	Brecha de costos en función del número de nodos.	43
6.1.	Espacio de mejora en Ruteo de Técnicos con Covering Tiempo Espacial.	45

Capítulo 1

Introducción

En la actualidad, existen numerosas empresas que proveen tanto productos como servicios con la característica de atender a sus clientes directamente en su ubicación, ya sea en su domicilio o en sus lugares de operación. Dentro de estas empresas podemos encontrar servicios de Courier (FedEx, UPS), servicios de entrega (Retail, Correo), servicios de reparación o mantención (servicios básicos: Agua, luz, teléfono, etc.), por nombrar algunas. Para poder llegar a cada cliente con cierta calidad de servicio, es necesario tener una red logística con capacidad suficiente, lo que implica una gran inversión en cuanto a medios de transporte y materiales y, además, una gran cantidad de dinero en términos de operación. El principal problema que presenta dicho estilo de servicio es lo complejo que se torna encontrar como utilizar los recursos óptimamente, lo que puede generar tanto sobrestimar como subestimar los recursos necesarios: Subestimar implica una reducción en la calidad de servicio, visible en clientes sin atender o atendidos tardíamente, mientras que sobrestimar implica capital en desuso y gastos operacionales innecesarios. Además, cuando el problema es dinámico (asociado a la incerteza de la operación diaria), todas las preocupaciones anteriores se vuelven aún más relevantes, pues las estimaciones pueden ser muy imprecisas si el problema no es tratado correctamente. La calidad de la información histórica dada por los registros de la operación anterior es muy importante para construir modelos de rutas consistentes, principalmente en casos donde las decisiones son dinámicas y el futuro es incierto. Es interesante, por tanto, generar técnicas que

busquen de manera eficiente y confiable decisiones óptimas en cuanto a la operación de dichos servicios, tanto en aspectos de flota como de ruteo de vehículos, las que pueden ser estáticas o dinámicas.

1.1. Motivación

Dentro de los Problemas de Ruteo de Vehículos (en adelante “VRP”), podemos mencionar el Problema de Ruteo de Técnicos (Traveling Technician Problem, en adelante “TTP”), que consiste en visitar un conjunto de puntos con un número acotado de rutas de la forma menos costosa posible, permaneciendo cierto tiempo en dicho punto. Dado que es una ligera variación del VRP, es un problema del tipo NP-Completo (Cordeau [10]), lo que implica que es un problema difícil de resolver, incluso en sus formulaciones más simples.

En las empresas, existen numerosas formas de resolver este problema, las cuales van desde complejas heurísticas computacionales hasta sistemas resueltos “a mano”, basados en el conocimiento y la experiencia de los involucrados, pero que distan de la realidad en variados puntos. De dichas diferencias, las que más llaman la atención son el hecho que no consideran la calidad de servicio dentro de la función de costos, y el no aprovechar de ninguna manera la información histórica que se tiene sobre el problema, principalmente reflejada en Bases de Datos con datos concretos sobre las necesidades de los clientes.

1.2. Objetivos

1.2.1. Objetivo General

El principal objetivo de este trabajo es desarrollar e implementar una metodología para enfrentar el Problema de Ruteo Dinámico de Vehículos desde el punto de vista de la calidad de servicio y considerando la información histórica del problema como ventaja comparativa.

1.2.2. Objetivos Específicos

- Investigar sobre actuales métodos de para enfrentar el problema de Ruteo Dinámico de Vehículos.
- Desarrollar o adaptar un modelo de resolución para el problema de Ruteo Dinámico de Vehículos que incluya el uso de información histórica.
- Estudiar si el uso de la información histórica ayuda a mejorar los resultados de la resolución.
- Resolver las instancias del Problema de Ruteo Dinámico de Técnicos de Xerox y comparar resultados con metodologías que no consideren ni dinamismo, ni uso de la información histórica.

1.3. Estructura del Trabajo

En el siguiente capítulo, Capítulo 2, se revisará la literatura relacionada a la heurística presentada, describiendo los principales avances en el área de ruteo dinámico de vehículos y abordando los tópicos relevantes para el desarrollo de dicha heurística. En el Capítulo 3 se explica el detalle de la heurística propuesta, haciendo énfasis en los modelos involucrados. Posteriormente, en el Capítulo 4 se detalla el problema que se resuelve con la heurística propuesta y se describen las instancias utilizadas, mientras que en el Capítulo 5 se ilustran los resultados obtenidos. Para finalizar, en el Capítulo 6 se muestra la formulación y los resultados preliminares de una versión complejizada de la heurística propuesta, mientras que el Capítulo 7 muestra las principales conclusiones de este trabajo.

Capítulo 2

Revisión Bibliográfica

El Problema de Ruteo de Vehículos con Ventanas de Tiempo (VRPTW) corresponde a una extensión del clásico Problema de Ruteo de Vehículos (VRP) donde el servicio para cada cliente debe comenzar dentro de una ventana de tiempo específica, generalmente dada por la disponibilidad del cliente, y donde, generalmente el vehículo debe mantenerse donde el cliente durante el servicio. Una variante de este problema, el cual se trata en este trabajo, es considerar dichas ventanas de tiempo de forma no rígida, dando la posibilidad de llegar fuera de lo establecido pero con cierto costo.

El VRPTW puede ser definido como sigue (Braysy [4]). Sea $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ un grafo conexo con un conjunto de $n+1$ nodos, cada uno de los cuales puede ser servido dentro de un intervalo específico de tiempo o “Ventana de Tiempo”, y un conjunto de \mathcal{E} arcos con pesos $d_{i,j}$ y con tiempos de viaje $t_{i,j}$. Se considera además un tiempo de atención para cada nodo, s_i , y que un vehículo puede llegar a un nodo antes de la apertura de la ventana de tiempo y esperar sin costo extra, pero no se le permite llegar después del cierre de la ventana. El nodo 0 representa el depósito de autos. Cada nodo i , que no sea el depósito, requiere una atención y el objetivo es encontrar el número mínimo de viajes, \mathcal{K}^* , para un conjunto de vehículos idénticos. Otros objetivos pueden ser minimizar la distancia recorrida o el tiempo utilizado por los vehículos. Un detalle importante es decir que todos los parámetros del

problema se dan por conocidos con certeza. Además, cada cliente debe ser servido solamente por un vehículo. Para el VRPTW clásico cada viaje corresponde a una ruta factible que comienza y termina en el depósito.

De la estructura de la formulación, es posible apreciar que el VRPTW conlleva una complejidad asociada importante: VRPTW es un problema “NP-hard” incluso en sus formulaciones más fáciles. Más aún, encontrar una solución factible para el VRPTW con una flota fija es, en sí, un problema NP-completo (Savelsbergh [18]). Por esto, existen dos importantes vertientes dentro la investigación del VRPTW: Una se centra en lograr la resolución exacta del problema utilizando la creciente capacidad computacional, el paralelismo (Gendreau [8]) y desarrollos matemáticos avanzados (Barnhart [1]); La otra, se preocupa de diseñar heurísticas capaces de resolver problemas reales con enfoques de optimización, como lo muestra Cordeau [7].

La gran mayoría de la investigación realizada en los últimos 40 años ha estado enfocada a la búsqueda de resoluciones exactas para problemas estáticos, donde todos los parámetros asociados son conocidos (Ropke [17], Laporte [13]). Sin embargo, las operaciones reales consideran numerosas variables inciertas que deben ser enfrentadas de forma inmediata, por lo cual se hace atractivo el uso de heurísticas desarrolladas a la medida. Más aún, la variabilidad de las condiciones en las cuales se realiza la resolución exacta de un problema puede generar que la solución obtenida pierda validez e incluso factibilidad, obligando a reoptimizar cada vez que dichas condiciones cambien. Esto refuerza aún más la idea de investigar y desarrollar modelos que estén preparados para enfrentar dicha variabilidad.

Este tipo de enfoque, llamado “enfoque dinámico”, ha tenido una especial atención en los últimos 15 años. En Ichoua [9] se puede ver un compilado de trabajos realizados en el área. Dentro de los estudios relativos al enfoque dinámico, podemos identificar dos grandes tipos de metodologías en la literatura, que se diferencian en la forma en la que enfrentan el dinamismo de las condiciones. El primer tipo, llamado desde “enfoque local”, genera soluciones a partir únicamente de la información que se tiene disponible y, por lo general, consiste en la resolución repetitiva de un problema estático. El segundo tipo, llamado “enfoque futuro”, trata de incorporar la información adicional que se tiene sobre el problema (distribuciones de probabilidad, información histórica, etc.) para predecir

el comportamiento futuro de las variables dinámicas. Dentro de los estudios que utilizan “enfoque futuro” para problemas de ruteo dinámico, podemos encontrar a: Bent and Van Hentenryck [3], que genera reglas mediante las cuales un vehículo que termina de servir un cliente debe esperar en el mismo lugar o bien dirigirse a un punto de espera; Larsen, Madsen, and Solomon [14] que decide si un vehículo desocupado espera en su actual ubicación o viaja a algún otro cliente (dinámico o existente) en función de la intensidad de los llamados por zona; y Chen [6], que agrega los clientes dinámicos por medio de la generación de nuevas rutas factibles por medio de “Dynamic Column Generation”, un método para agregar información (nuevos llamados) a un problema de Generación de Columnas.

Capítulo 3

Heurística Propuesta

El presente capítulo muestra una heurística para solucionar el problema del ruteo dinámico de técnicos, utilizando la información histórica del problema.

3.1. Heurística Propuesta

Básicamente, la heurística consiste en utilizar la información histórica existente, en particular los llamados pasados, para formar conjuntos de llamados, o “clusters”, que sirvan para indicar cuales son los lugares representativos de las llamadas futuras y poder utilizarlos como puntos de espera. Una vez obtenidos los lugares de espera, en adelante “idle points”, se procede a rutear determinísticamente todos los llamados que estén disponibles junto con los “idle points”, los cuales se ubican exclusivamente al final de las rutas. Dicho ruteo es realizado por medio la optimización de un problema de programación lineal, resuelto a través de Generación de Columnas. La particularidad de este método es que cada columna generada corresponde a una ruta factible, por lo cual es posible utilizarlas en procesos posteriores de la heurística.

Las rutas obtenidas de la resolución del problema, conocidas como rutas estáticas, se ponen en operación y se utilizan hasta que un nuevo llamado llega. Cuando esto ocurre, se toma un

subconjunto de las rutas generadas durante el proceso de Generación de Columnas y se inserta el nuevo llamado a dichas rutas bajo algún criterio de costo, para obtener así nuevas rutas. Finalmente, se actualiza el Problema de Programación Lineal y se resuelve nuevamente, pero considerando tanto las rutas generadas en el ruteo estático como las recientemente creadas.

La figura 3.1 muestra el algoritmo propuesto, definiendo los procesos utilizados y sus inputs y outputs.

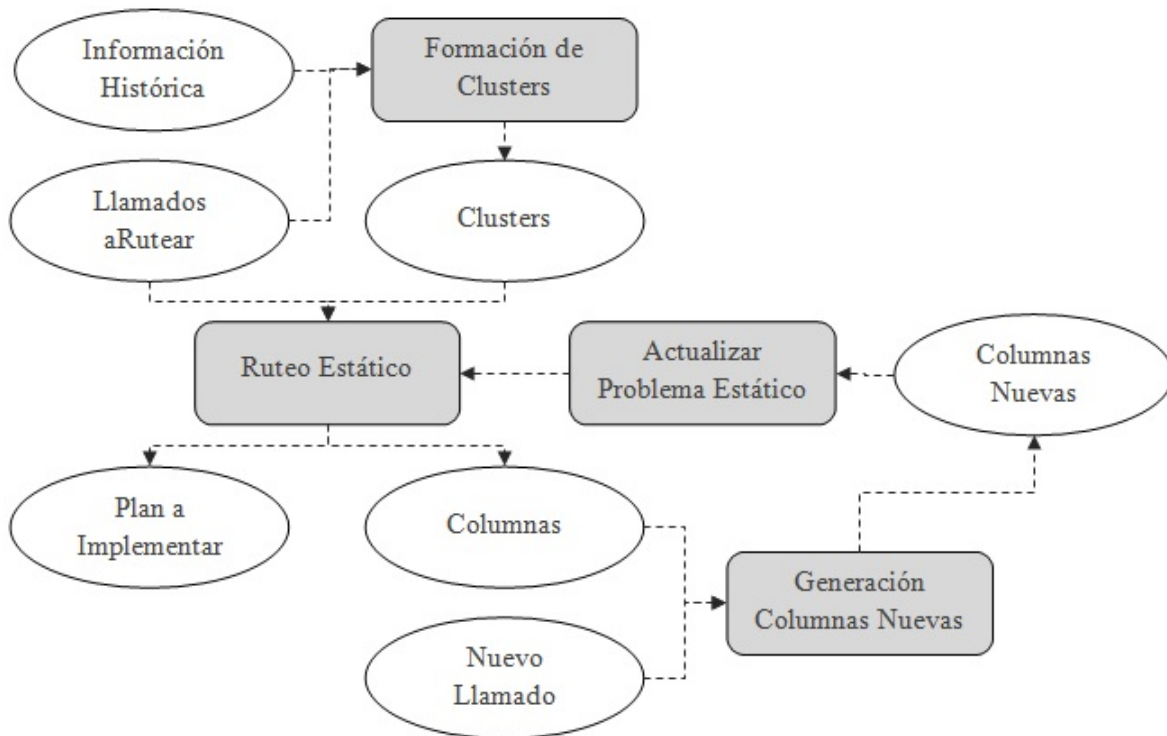


Figura 3.1: Algoritmo Propuesto.

3.2. Detalle de Modelos

3.2.1. Clasificación

Para los problemas de optimización dinámicos, en particular para los problemas de ruteo dinámico, es muy importante anticipar los eventos futuros, pues el resultado general del ejercicio no sólo depende de cuan buenas sean las rutas creadas con los datos que se tienen a mano, sino que también dependen de cuan efectiva es la respuesta que se le da a los eventos que no se conocen de antemano. Es por esto que muchos de los trabajos orientados a Ruteo de Vehículos Dinámicos en los últimos años apuntan a la utilización de la información histórica de los datos para lograr cierta ventaja al momento de optimizar: En el trabajo de Ichoua[9] se pueden ver las distintas formas de enfrentar los problemas de ruteo utilizando información histórica, haciendo énfasis en la creación de nodos para generar cobertura dentro del territorio. Por otro lado, Weintraub [20] añade costos a la función objetivo de su problema de Ruteo de Vehículos de Emergencia para no dejar zonas descubiertas por más de cierto tiempo.

En el presente trabajo, aprovechamos la información histórica agrupando los llamados de los clientes con el objetivo de crear zonas de espera, las que se pueden entender como lugares donde la probabilidad de que llegue un llamado es mayor. La idea es probar que si los vehículos se encuentran en uno de estos puntos de espera (o tienden a él), la respuesta que dichos vehículos presentan ante los nuevos llamados es mejor, a pesar que se invierta tiempo en llegar ha dichos puntos. La agrupación de llamados históricos se realiza por medio de diferentes técnicas de Análisis de Clasificación (“Clustering Análisis”), actividad que se preocupa de asignar un conjunto de elementos a deferentes grupos, en base a algún criterio de similaridad que dichos elementos presenten. Cabe destacar que el Análisis de Clasificación corresponde a un área muy importante en Data Mining y en Análisis Estadístico de Datos y es comúnmente usado en Machine Learning, Pattern Recognition, Análisis de Imágenes, etc.

Dentro del Análisis de Clasificación existen dos grandes ramas: Clasificación Supervisada y Clasificación no Supervisada. Un método de clasificación pertenece a alguna de estas dos ramas

dependiendo si es capaz de asignar un nuevo input a alguna clase supervisada o no. En la Clasificación Supervisada, el mapeo desde un conjunto de vectores ($x \in \mathbb{R}^d$. Con d la dimensión) a un conjunto discreto de “labels” ($Y = 1..C$, con C cardinalidad del conjunto), es modelada en términos de una función $Y = Y(X, w)$, donde w corresponde a parámetros ajustables. Los valores de estos parámetros son determinados por algún método de optimización que busca minimizar una función de riesgo, en un conjunto finitos de datos de entrada (X_i, Y_i) , con i en $1..N$, con N la cardinalidad del conjunto de pruebas.

En la Clasificación No Supervisada, llamada comúnmente “clustering” (del inglés “cluster”, grupo), no existe un conjunto de “labels” que permita calibrar parámetros funcionales, por lo cual el objetivo de dicha técnica es separar el conjunto de elementos en un número discreto de grupos por medio de algún patrón no visible. Debido a la naturaleza exploratoria del procedimiento, la Clasificación No Supervisada suele ser un proceso subjetivo, donde la eficacia del método depende del juicio de quien lo usa.

Dado que no se tienen “labels” de clasificación para los datos a utilizar, se utilizan métodos de Clasificación No Supervisada para el problema presentado.

Dada la versatilidad del método y su fácil implementación, se utiliza K-means para la creación de las zonas de espera a partir de los datos históricos.

K-Means (o K-Medias) corresponde a un método de clasificación no supervisado que particiona un conjunto de objetos N -dimensional en k conjuntos (MacQueen[16]), donde cada observación pertenece al cluster cuya media esté más cercana.

De acuerdo a la descripción de (Jain[12]), el algoritmo asociado a K-Means es el siguiente:

1. Se eligen al azar k puntos (medias) dentro del hipervolumen que contiene al conjunto de objetos
2. Se asigna cada objeto al punto (media) más cercano.
3. Se recalcula el centro del conjunto usando los objetos que lo componen.

4. Si el criterio de convergencia no se cumple, se vuelve al paso 2.

Los criterios de convergencia suelen corresponder a:

- No se reasignan más que un umbral de elementos dentro de una iteración.
- Reducción marginal en el error cuadrado de las muestras.

A pesar que el algoritmo siempre termina y arroja un resultado, dicho resultado no necesariamente encuentra la configuración óptima, correspondiente al mínimo global de la función objetivo. Esto ocurre porque el algoritmo es muy sensible a los puntos de inicio elegidos al azar, por lo cual distintos inicios pueden llevar a distintos resultados. Para minimizar dicho efecto, en este trabajo se utiliza la versión de Programación Lineal Entera del algoritmo recientemente mostrado.

Sea N el conjunto de elementos a agrupar, donde cada elemento posee C características. Se define f_c^i , con $i \in N$ y $c \in \{1..C\}$, como la coordenada c del elemento i . Sea $M_{i,j} = \sqrt{\sum_{c \in C} (f_c^i - f_c^j)^2}$, matriz simétrica de distancia euclidiana entre los puntos i y j .

A su vez, se definen las siguientes variables de decisión

$$z_j = \begin{cases} 1 & \text{si elemento } j \text{ es centroide} \\ 0 & \text{en otro caso} \end{cases}$$

$$w_{i,j} = \begin{cases} 1 & \text{si elemento } i \text{ está asociado a centroide } j \\ 0 & \text{en otro caso} \end{cases}$$

En base a esto, el Problema de Programación Lineal queda definido como

$$\min_{z,w} \quad \sum_{i,j} w_{i,j} M_{i,j} \quad (3.1)$$

$$\text{s.a.} \quad \sum_{j \in N} z_j = k \quad (3.2)$$

$$\sum_{i \in N} w_{i,j} \leq z_j * \text{Card}(N) \quad \forall j \in N \quad (3.3)$$

$$\sum_{j \in N} w_{i,j} = 1 \quad \forall i \in N \quad (3.4)$$

La función objetivo (3.1) minimiza la suma de las distancia euclidiana total intragrupos. La restricción (3.2) fija el número de centroides elegidos en k . La restricción (3.3) asegura la consistencia de asociación: un elemento se asocia a otro si y sólo si este corresponde a un centroide. Finalmente, La restricción (3.4) exige que todos los elementos sean asignados a un centroide.

Dado el contexto en el cual se utiliza la heurística que se está proponiendo (atención de urgencia, reparaciones, etc.), se considera que debe haber algún factor dentro del algoritmo que considere la latencia de una llamada para una zona determinada. Por ejemplo, si tenemos dos clientes equivalentes, pero uno a uno de ellos se le solucionó un problema hace poco tiempo, es más probable que si se genera una llamada ésta corresponda al cliente que no ha tenido problemas recientemente. Si bien esto no está de acuerdo con que las propiedades de la distribución exponencial, si es consistente con el supuesto básico de este trabajo, donde no es posible a partir de los datos asumir una distribución para la ocurrencia de los fenómenos que nos interesan.

Para tomar en cosideración la latencia del problema, se propone una medida de distancia que considere el tiempo de ocurrencia de un evento histórico, para así acercar los centrides del algoritmo de K-medias a los lugares donde se sospecha ocurrirá un nuevo fenómeno relevante.

Sea t_0 el momento actual, t_k el momento de ocurrencia del fenómeno k y t_{min} el momento

de ocurrencia del evento más antiguo en la base utilizada, se define el ponderador ζ como

$$\zeta_k = \frac{t_k - t_{min}}{t_0 - t_{min}}$$

Donde la nueva matriz de distancia entre el elemento i y el centroide j , $\hat{M}_{i,j}$, queda definida como

$$\hat{M}_{i,j} = \zeta_j \sqrt{\sum_{c \in C} (f_c^i - f_c^j)^2}$$

Se observa que el ponderador ζ se acerca a 0 cuando el tiempo de ocurrencia del fenómeno es próximo al más antiguo, por lo cual, la distancia relativa existente desde un elemento a un centroide es menor mientras el centroide sea más antiguo y viceversa.

3.2.2. Problema de Ruteo de Técnicos con Ventanas de Tiempo

A continuación, se muestra la formulación del problema de Ruteo de Vehículos con nodos ficticios, la cual tiene por objetivo utilizar de forma inteligente la información histórica que se tiene sobre el problema para generar ruteos dinámicos más eficientes en términos de calidad de servicio. Para este estudio, la calidad de servicio se mide mediante el tiempo de espera del cliente, que corresponde al tiempo entre el llamado del cliente y la espera límite de la ventana de tiempo, y la violación de dicha ventana de tiempo. A continuación, se muestra la formulación general del problema y su representación como un problema de *Generación de Columnas* mediante la cual se solucionará el problema.

Problema de Ruteo de Vehículos con Ventanas de Tiempo y Nodos Ficticios

El modelo general para el problema, desde el cual se desprende la formulación del Problema Maestro y el subproblema relativo a Generación de Columnas, es el siguiente:

- I_0 : Conjunto de nodos ficticios generados
 I_1 : Conjunto de nodos iniciales
 I_2 : Conjunto de nodos a ser ruteados dentro del día
 Z : Conjunto de zonas.
 F : Hora en que finaliza la jornada laboral
 K : Número de técnicos disponibles
 A : Conjunto de todos los arcos existentes
 l_i : Momento de la llamada relativa a nodo i
 b_i : Ventana de tiempo superior para cada nodo
 s_i : Tiempo de reparación promedio para cada nodo
 $t_{i,j}$: Tiempo de viaje estimado entre el nodo i y nodo j
 d_z : Densidad de llamados de zona z
 $Cov_{i,z}$: Covertura que el nodo ficticio i ofrece a la zona z
 $MinCov_z$: Covertura mínima requerida para la zona z

$$A = \{(i, j) / i \in I_1 \cup I_2, j \in I_0 \cup I_2, i \neq j\}$$

- $w_{i,k}$: Variable de Decisión. Momento en que técnico k llega a nodo i
 δ_i^k : Variable de Decisión. Violación de restricción de tiempo de llegada del técnico k al nodo i
 f_z : Variable de Decisión. Cobertura ofrecida a la zona z
 $x_{i,j}^k$: Variable de Decisión

$$x_{i,j}^k = \begin{cases} 1 & \text{si técnico } k \text{ atiende a nodo } j \text{ inmediatamente después de nodo } i \\ 0 & \text{en caso contrario} \end{cases}$$

- λ_i^k : Variable Auxiliar. Demora en atender al nodo i dentro de ventana de tiempo

$$\lambda_i^k = \begin{cases} (w_i^k - l_i - \delta_i^k) & \text{si } w_i^k \text{ es distinto de } 0 \\ 0 & \text{en otro caso} \end{cases}$$

Utilizando los parámetros α , β y φ (con α , β y $\varphi \in [0, 1]$) y las variables de decisión definidas, el problema a resolver tiene la siguiente forma:

$$\min_{x,w,\delta,f} \sum_{k \in K} \sum_{i \in I_2} \delta_{i,k} + \alpha \sum_{k \in K} \sum_{i \in I_1 \cup I_2} \lambda_{i,k} - \beta \sum_{z \in Z} d_z f_z + \varphi \sum_{k \in K} \sum_{(i,j) \in A} t_{i,j} x_{i,j}^k \quad (3.5)$$

$$\text{s.a.} \quad \sum_{k \in K} \sum_{j: (i,j) \in A} x_{i,j}^k = 1 \quad \forall i \in I_1 \cup I_2 \quad (3.6)$$

$$\sum_{(i,j) \in A / j \in I_0} x_{i,j}^k = 1 \quad \forall k \in K \quad (3.7)$$

$$\sum_{j: (i,j) \in A} x_{i,j}^k - \sum_{j: (j,i) \in A} x_{j,i}^k = W_i^k \quad \forall k \in K, i \in I_1 \cup I_2 \quad (3.8)$$

$$w_{i,k} \leq F \sum_{j: (j,i) \in A} x_{j,i}^k \quad \forall i \in I_0, k \in K \quad (3.9)$$

$$w_{i,k} - \delta_{i,k} \leq b_i \quad \forall i \in I_2, k \in K \quad (3.10)$$

$$w_{i,k} + s_i + t_{i,j} - w_{j,k} \leq (1 - x_{i,j}^k)N \quad \forall (i,j) \in A, k \in K \quad (3.11)$$

$$\sum_{k \in K} \sum_{(i,j) \in A / j \in I_0} x_{i,j}^k Cov_{j,z} \geq f_z \quad \forall z \in Z \quad (3.12)$$

$$\lambda_{i,k} + \delta_{i,k} \leq D \quad \forall i \in I_1 \cup I_2, \forall k \in K \quad (3.13)$$

El parámetro W_i^k está definido como

$$W_i^k = \begin{cases} 1 & \text{si } i = k, \text{ con } k \in K \\ 0 & \text{en caso contrario} \end{cases}$$

A su vez, el parámetro $N \gg 1$ es un parámetro auxiliar para acotar la restricción (3.11).

La función objetivo (3.5) considera el costo de violación de ventanas de tiempo como costo primario, y los costos de tiempos de espera, de transporte y un premio por aumentar el covering del problema. La restricción (3.6) exige que cada nodo sea visitado 1 vez, sin importar que técnico pase por el nodo. A su vez, la restricción (3.7) restringe que de todos los nodos ficticios generados se utilicen sólo K (uno por técnico). La restricción (3.8) asegura la conservación de flujo. La restricción (3.9) restringe la llegada al nodo terminal a un momento no superior al final de jornada. La restricción (3.10) corresponde a la restricción relativa a las ventanas de tiempo, mientras de la restricción (3.11) garantiza la consistencia temporal de las rutas. Se destaca que para la restricción (3.11) los valores de la variable s_i está fija en 0 para todo nodo ficticio.

La restricción (3.12) asegura que el covering de los nodos ficticios respete las necesidades de cada zona. Finalmente, la restricción (3.13) permite que la espera total de los clientes para ser atendidos sea menor a un tiempo fijo (por ejemplo, 48 hrs.).

3.2.3. Descomposición para Generación de Columnas. Formulación sin decisión en tiempos de espera

Problema Maestro

El problema maestro se considera un problema de *Set Partitioning* donde el problema a resolver se define como se muestra a continuación:

- R^* : Conjunto de todas las rutas generadas
 θ_r : Variable de decisión binaria; se usa o no la ruta r
 C_r : Costo asociado a ruta r
 $H_{i,r}$: Matriz binaria; nodo i se encuentra en ruta r , o no

$$\min_{\theta, f} \sum_{r \in R^*} C_r \theta_r - \sum_{z \in Z} d_z f_z \quad (3.14)$$

$$\text{s.a.} \quad \sum_{r \in R^*} \theta_r \leq K \quad (3.15)$$

$$\sum_{r \in R^*} H_{i,r} \theta_r \geq 1 \quad \forall i : (i, j) \in A \quad (3.16)$$

$$\sum_{r \in R^*} \theta_r \sum_{i \in I_0} Cov_{i,z} H_{i,r} \geq f_z \quad \forall z \in Z \quad (3.17)$$

La función objetivo (3.14) minimiza costos de transporte de las rutas utilizadas y maximiza el covering. La restricción (3.15) garantiza que la cantidad de rutas utilizadas no exceda la cantidad de técnicos disponibles. La restricción (3.16) asegura que todos los nodos sean visitados al menos una vez. La restricción (3.17) conserva la relación entre la cobertura de las zonas y las rutas utilizadas. Se define además π como la variable dual asociada a la restricción (3.15), mientras que γ_i es la variable dual asociada a la restricción (3.16).

Problema de Pricing

La siguiente formulación se encarga de generar rutas de costo mínimo (por eso el subíndice k desaparece de la formulación).

La formulación queda de la forma:

$$\min_{x,w,\delta} \quad \pi + \sum_{i \in I_1 \cup I_2} \delta_i + \alpha \sum_{i \in I_1 \cup I_2} \lambda_i + \varphi \sum_{(i,j) \in A} t_{i,j} x_{i,j} - \sum_{i:(i,j) \in A} x_{i,j} \gamma_i \quad (3.17)$$

$$\text{s.a.} \quad \sum_{i \in I_1} \sum_{j \in I_2} x_{i,j} = 1 \quad (3.18)$$

$$\sum_{j:(i,j) \in A} x_{i,j} - \sum_{j:(j,i) \in A} x_{j,i} = W_i^* \quad \forall i : (i,j) \in A \quad (3.19)$$

$$w_i \leq F \sum_{j:(j,i) \in A} x_{j,i} \quad \forall i \in I_0 \quad (3.20)$$

$$w_i - \delta_i \leq b_i \quad \forall i \in I_2 \quad (3.21)$$

$$w_i + s_i + t_{i,j} - w_j \leq (1 - x_{i,j})N \quad \forall (i,j) \in A \quad (3.22)$$

$$\sum_{i \in I_0} \sum_{j \in I_0 \cup I_1 \cup I_2} x_{j,i} \leq M \quad (3.23)$$

$$\lambda_i + \delta_i \leq D \quad \forall i \in I_1 \cup I_2 \quad (3.24)$$

El parámetro W_i^* está definido como

$$W_i^* = \begin{cases} 1 & \text{si } i \in I_1 \\ -1 & \text{si } i \in I_0 \\ 0 & \text{en caso contrario} \end{cases}$$

La única diferencia considerable con el modelo completo es que la restricción (3.18), junto con las restricciones de flujo, sólo permiten la construcción de una ruta de mínimo costo.

3.2.4. Generación de Columnas Dinámico

Generación de Columnas Dinámico (Chen [6]) es una extensión de la metodología de Generación de Columnas para problemas de naturaleza dinámica, en la cual se generan nuevas columnas a medida que nueva información es añadida al problema.

Dado que Generación de Columnas es una metodología que funciona bien con problemas degenerados donde el número de variables es considerablemente mayor que el número de restricciones, se comporta bien también con los Problemas de Ruteo de Vehículos, pues cada variable de decisión corresponde a una ruta factible, las cuales pueden ascender a cientos de miles, dependiendo del número de nodos a visitar. Es por esto que se considera pertinente la utilización de una versión dinámica de esta metodología para enfrentar la versión dinámica del Problema de Ruteo de Vehículos.

Básicamente, el funcionamiento de Generación de Columnas Dinámico se basa en la creación de nuevas columnas a partir de la nueva información, por medio del uso de heurísticas de modificación de las columnas que se tienen con anterioridad. Dado que la metodología se basa en la reutilización de columnas, es capaz de generar una gran cantidad de nuevas variables de decisión en muy poco tiempo, lo cual favorece enormemente su uso en situaciones donde se dispone de poco tiempo para generar un nuevo resultado.

A continuación, se explica detalladamente la metodología utilizada.

Algoritmo

1. Obtener solución del Problema Estático

Se expresa un Problema de Programación Lineal por medio de Generación de Columnas y se resuelve a optimalidad. Cada una de las variables de decisión obtenidas corresponden a una columna de la matriz asociada al problema expresado en forma estándar. A su vez, y por construcción, cada columna corresponde a una ruta factible para un vehículo. Dichas columnas se almacenan y se identifican para poder ser utilizadas posteriormente. Cabe destacar que la solución obtenida es la que se implementa operacionalmente hasta que se reciba un nuevo llamado.

2. Selección de Columnas Antiguas

En el momento en que un nuevo llamado es recibido, se selecciona un subconjunto de las columnas que se tienen almacenadas por medio de algún criterio. Para el caso particular de

esta implementación, el criterio de selección corresponde a *Menor Valor del Costo Reducido*. Se sabe que el Costo Reducido corresponde a cuanto empeora la función objetivo en el caso de aumentar en una unidad la variable de decisión asociada a éste, por lo tanto se seleccionan aquellas columnas que generan el menor cambio relativo en la función objetivo. Este razonamiento se basa en el hecho que se intenta minimizar el aumento de costos en la función objetivo al momento de agregar el nuevo requerimiento. En base a esto, se procede a seleccionar un conjunto fijo de columnas de acuerdo al criterio de *Menor Valor del Costo Reducido*.

Una vez fijado el criterio de selección, otra decisión importante es el número de columnas que se quieren utilizar. Como no existe criterio claro para seleccionar dicho número, se decide que los recursos computacionales sean los que definan un porcentaje óptimo para dicha selección. De acuerdo a esto, se utiliza dicho porcentaje como un 20%, con el objetivo de ingresar al problema de optimización un máximo de 15 nuevos nodos ficticios sin tener problemas de ejecución por falta de memoria.

3. Generación de Nuevas Columnas

Si nombramos como L el conjunto de columnas a modificar, la generación de las nuevas rutas corresponde a insertar el nuevo llamado en la posición factible menos costosa para cada una de las columnas $l \in L$. Dicha posición se obtiene mediante el algoritmo detallado en el (ANEXO). Donde *Función de Costos* corresponde a la función de costos utilizada. Por criterios de consistencia, la función de costos utilizada corresponde a la misma función de costos utilizada en el proceso de optimización.

4. Actualización del problema de optimización

Una vez obtenidas las nuevas rutas factibles, se procede a actualizar el problema de optimización. Dicho procedimiento consta de dos pasos:

- Agregar el nuevo nodo a las restricciones del problema de pricing y a las restricciones (3,16) y (3,17) del problema maestro para lograr consistencia en la resolución.
- Eliminar los nodos ya visitados y fijar los inicios de las rutas de acuerdo al estado actual de los vehículos.

Con dichos pasos cubiertos, es posible re-optimizar el problema considerando la nueva información

5. Vuelvo a paso 1.-

Una de las principales diferencias de la Generación de Columnas Dinámica utilizada y la presentada en Chen [6], corresponde a la utilización de bloques de tiempo. Mientras la versión original divide el día operacional en una serie de bloques en los cuales se acumulan llamados para insertar dinámicamente, la versión presentada inserta y re-optimiza en el momento en que se genera un nuevo llamado. Si bien el método propuesto está mucho más cerca de las nociones de *On-line Optimization* (Jaillet [11]), genera más esfuerzo operacional. Por otro lado, el modelo original utiliza un número variable de rutas posibles, bajo el argumento que es posible contratar a bajo costo y dinámicamente otro vehículo, lo cual no es efectivo en este caso.

Capítulo 4

Problema a Resolver

4.1. Descripción del Problema

El modelo desarrollado y explicado anteriormente se utiliza para resolver un problema real de despacho dinámico de técnicos. Dicho problema corresponde al Servicio de Reparación de Fallas que enfrenta Xerox Chile en la ciudad de Santiago.

Xerox es una compañía multinacional de tecnología y servicios, con ingresos anuales totales mayores a US\$15.700 billones, que ayuda a sus clientes a hacer más productivo su trabajo por medio de herramientas y servicios de administración de documentos. Su objetivo es liderar el mercado de tecnologías innovadoras con productos y con servicios de los cuales los clientes puedan depender para mejorar los resultados de sus negocios. En Chile, Xerox lleva más de 35 años y cuenta con un equipo humano de más de 600 personas con cobertura a lo largo de todo el país.

La principal preocupación del Servicio de Reparación de Fallas de Xerox (también llamado Servicio Técnico) consiste en responder a las necesidades del cliente (ya sea reparación o mantenimiento) en el menor tiempo posible, considerando que para cada cliente existe un tiempo de respuesta prometido, el cual depende de su prioridad comercial. Para responder a dichas necesidades existe

un número limitado de técnicos (más de 100) y de vehículos, los cuales deben ser distribuidos tomando en cuenta los intereses de la empresa, los que pueden ser tanto netamente económicos como de calidad de servicio.

Sólo en Santiago se reciben más de 400 llamadas diarias, las cuales deben ser atendidas lo antes posible. Cabe destacar que los planificadores, encargados de hacer el ruteo de cada vehículo, han dividido Santiago en áreas de atención y cada área tiene un número fijo de técnicos asignados para servir los requerimientos que en estas se generen. Esta estrategia responde de forma intuitiva a la lógica de ruteo establecida en trabajos como el de Carlsson[5], donde se dividen los llamados antes de efectuar el ruteo en sí.

De acuerdo a la operación diaria establecida, cada vez que llega un nuevo llamado éste pasa a la lista de espera correspondiente al área donde se originó el requerimiento hasta el momento en que algún técnico disponible es asignado. Dicha asignación es llevada a cabo por un equipo de despacho, quien se encarga de mantener las listas de espera de cada área ordenada según el tiempo de espera del cliente y su prioridad, tomando en cuenta además la posición de cada técnico al momento de la asignación.

4.2. Zona de Estudio

La ciudad de Santiago posee un total de 37 comunas y alrededor de 17 millones de habitantes, las cuales generan más de 150 mil llamados en un año, o un equivalente a 420 llamados para cada día hábil. De dichos llamados, un 35 % por ciento se concentran en las comunas de Santiago Centro y de Providencia, las cuales concentran la mayoría de las actividades económicas de Santiago en 36 km^2 , el equivalente a un 6 % del área total del Gran Santiago. Debido a la alta concentración de llamados de esta zona, se considera poco relevante realizar el ruteo de técnicos en dicha zona, principalmente debido a el desplazamiento casi nulo de los vehículos entre llamados. Por otro lado, la zona Sur-Oriente de Santiago, con un área aproximada de 370 km^2 (57 % del área total), alberga aproximadamente 33 mil llamados anuales, equivalente a 105 llamados en promedio para cada día

de operación. Dado que la densidad de llamados en dicha zona es menor, se selecciona para resolver el problema antes descrito. De esta forma, se espera que se pueda observar mejor la relación entre distancia recorrida por cada técnico y el tiempo de espera de cada cliente.

La zona Sur-Oriente de Santiago está conformada por 22 comunas, las cuales se muestran a continuación:

Comuna	Habitantes	Superficie Urb <i>km²</i>
Cerrillos	82262	19,1
El Bosque	211246	17,2
La Cisterna	92499	10,9
La Florida	468536	50,1
La Granja	167777	12,6
La Pintana	287659	46,4
La Reina	109048	26,7
Las Condes	239234	94,9
Lo Barnechea	84179	63,8
Lo Espejo	112896	8,4
Macul	130732	14,9
Maipu	432616	51,2
Ñuñoa	176780	18,2
P.A. Cerda	106108	8,2
Padre Hurtado	34257	7,4
Peñalolen	236749	59,6
Puente Alto	510534	66
San Bernardo	297774	64,6
San Joaquin	100680	10,2
San Miguel	75001	9,2
San Ramon	102104	6,8
Vitacura	89363	31,3

Tabla 4.1: Comunas de Santiago Sur-Oriente con sus habitantes y su superficie urbana.

Para facilitar la resolución del modelo, se construye sobre la zona indicada una red bi-direccional de 49 nodos, donde cada nodo representa una sub-zona (una fracción de una comuna) y se considera que cada uno de los llamados asociados a la sub-zona se ubica en el centro geográfico de dicha zona. La división realizada corresponde a zonas que, dentro de lo posible, mantienen un número de llamados diarios parecido, pero no necesariamente un área similar. Además, se considera

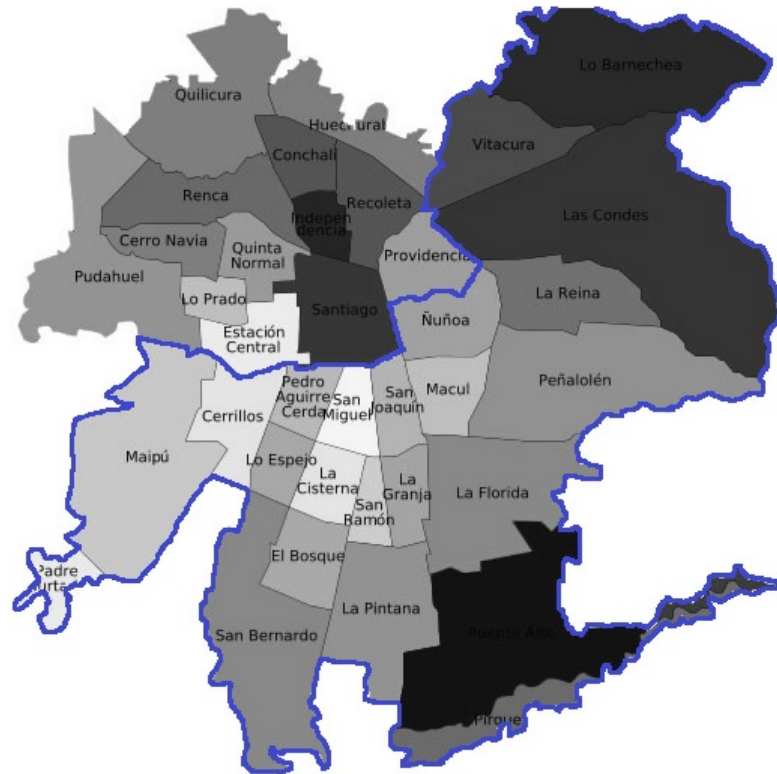


Figura 4.1: Zona Sur-Oriente de Santiago.

que la distancia existente entre 2 llamados de una misma sub-zona corresponde al radio aproximado de la misma.

La distancia entre nodos para dicha red es calculada a través de una macro ejecutada sobre Google Maps. Se prefiere esta opción pues la existencia de carreteras interurbanas y accidentes geográficos (levantamientos de tierra, ríos, sembradíos rurales, etc) generan que la distancia euclidiana pierda validez y sub o sobre estime los tiempos de viaje. Cabe destacar que, bajo el mismo argumento, la distancia entre los nodos A y B puede perfectamente ser diferente a la distancia entre los nodos B y A.

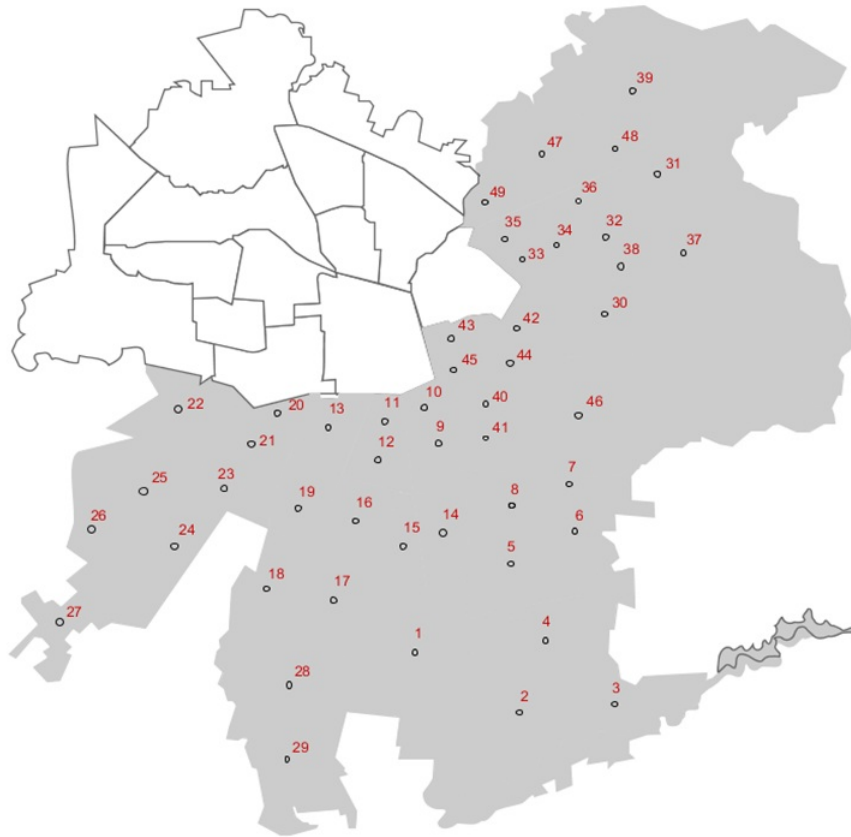


Figura 4.2: Nodos representantes de las 49 zonas a utilizar.

4.3. Estudio de Demanda y Estadísticas de la Zona

En la zona seleccionada para el estudio, correspondiente a 22 de las 37 comunas del Gran Santiago, se producen un total anual¹ de 33,359 llamados, correspondientes a un 40 % de los llamados anuales totales de la ciudad. En la tabla 4.2 se puede ver la distribución de dichos llamados en las 49 zonas de estudio.

Se puede observar que, manteniendo las concentraciones de población del gran Santiago, las zonas ubicadas en las comunas de Las Condes, Ñuñoa y Vitacura comprenden la mayor concentración de llamados, bordeando el 5 %, mientras que las zonas más periféricas (ubicadas en las

¹Considerando que el año de operación transcurre entre Mayo de 2002 y Abril de 2003

	Comuna	Llamados	%		Comuna	Llamados	%
1	La Pintana	157	0,47	26	Maipu-Rinconada	112	0,34
2	Puente Alto - Centro	611	1,83	27	Padre Hurtado	292	0,88
3	Puente Alto - Este	120	0,36	28	San Bernardo - Sur	204	0,61
4	Puente Alto - Norte	272	0,82	29	San Bernardo - Centro	815	2,44
5	La Florida - Sur	150	0,45	30	La Reina	870	2,61
6	La Florida - SurPoniente	111	0,33	31	Las Condes - Alto	1295	3,88
7	La Florida - Poniente	221	0,66	32	Las Condes - Apumanque	926	2,78
8	La Florida - Vespucio	1096	3,29	33	Las Condes - El Golf	2301	6,9
9	San Joaquín - Sur	1023	3,07	34	Las Condes - Escuela	2516	7,54
10	San Joaquín - Norte	831	2,49	35	Las Condes - Isidora	3190	9,56
11	San Miguel - Norte	663	1,99	36	Las Condes - Kennedy	680	2,04
12	San Miguel - Sur	475	1,42	37	Las Condes - Poniente	441	1,32
13	P.A. Cerda	140	0,42	38	Las Condes - Sur	334	1
14	La Granja	108	0,32	39	Lo Barnechea	591	1,77
15	San Ramon	119	0,36	40	Macul - Norte	623	1,87
16	La Cisterna	297	0,89	41	Macul - Sur	857	2,57
17	El Bosque	384	1,15	42	Ñuñoa - NorEste	597	1,79
18	San Bernardo - Norte	533	1,6	43	Ñuñoa - NorOeste	1069	3,2
19	Lo Espejo	173	0,52	44	Ñuñoa - SurEste	362	1,09
20	Cerrillos - Norte	532	1,59	45	Ñuñoa - SurOeste	1787	5,36
21	Cerrillos - Centro	342	1,03	46	Peñalolen	391	1,17
22	Maipu-Pajaritos	278	0,83	47	Vitacura - Centro	975	2,92
23	Cerrillos - Sur	552	1,65	48	Vitacura - Este	353	1,06
24	Maipu-Industrial	406	1,22	49	Vitacura - Oeste	1583	4,75
25	Maipu-Centro	601	1,8				

Tabla 4.2: Zonas definitivas de trabajo, Santiago Sur-Oriente.

comunas de Maipú, Puente Alto, Padre Hurtado, La Pintana, San Bernardo, etc.) poseen concentraciones muy bajas, las cuales no alcanzan al 0,5 %. Este fenómeno es fácilmente explicable teniendo en consideración la actividad comercial de las zonas mencionadas; mientras en las zonas más cercanas al centro (Las Condes, Vitacura) existe más actividad comercial y, por tanto, más llamados, en las zonas más periféricas (San Bernardo, La Pintana, Puente Alto) los llamados escasean. Debido a lo anterior, es importante procurar que las rutas generadas no se concentren masivamente en las zonas de mayor frecuencia, pues una llamada en una zona periférica puede resultar muy perjudicada, a pesar que se trate de un cliente con prioridad alta.

En la tabla 4.3 se observa la frecuencia de llamados totales anuales por mes de operación. Aquí se puede verificar el fenómeno observado por Souyris[19], donde la mayor concentración de llamados se produce en los meses de invierno, donde además se produce la mayor cantidad de tra-

bajadores en servicio activo, mientras en la menor concentración se presenta en los meses de verano (Diciembre, Enero, Febrero), correspondiente a época de vacaciones. En cuanto a la distribución mensual de llamados por zona, tabla A.1 que se puede observar en Anexo, es posible notar que cada zona respeta el patrón de llamados.

Mes	Llamadas	%
Mayo - 2002	3078	9,23
Junio - 2002	2775	8,32
Julio - 2002	3322	9,96
Agosto - 2002	2937	8,8
Septiembre - 2002	2521	7,56
Octubre - 2002	3129	9,38
Noviembre - 2002	2679	8,03
Diciembre - 2002	2357	7,07
Enero - 2003	2479	7,43
Febrero - 2003	2278	6,83
Marzo - 2003	2926	8,77
Abril - 2003	2878	8,63

Tabla 4.3: Llamados totales anuales por mes y porcentaje.

Respecto al promedio de llamados totales por día de la semana, los cuales se encuentran en la tabla 4.4, se puede observar que la varianza para un día cualquiera promedia los 32 llamados, lo cual implica una gran variabilidad para al momento de realizar las rutas, pues se dan las condiciones para que el número de vehículos disponibles sea insuficiente para enfrentar la demanda.

Día	Llamadas Anuales	%	Promedio Diario	σ	σ en %
Lunes	7673	23	147,56	34,97	23,7
Martes	6645	19,92	127,79	37,34	29,22
Miercoles	6706	20,1	131,49	33,71	25,64
Jueves	6686	20,04	131,1	26,34	20,09
Viernes	5649	16,93	110,76	24,52	22,14

Tabla 4.4: Llamados totales anuales por día de la semana, porcentaje y varianza.

Por otro lado, de la tabla A.2, correspondiente a los llamados promedios por hora del día (ver ANEXO), se puede observar que en las zonas de mayor frecuencia se observan dos peaks marcados: Durante las primeras horas del horario de oficina (entre 9:00 y 10:00 de la mañana), se reciben la

mayor cantidad de llamados, seguida de una reducción considerable debido a la hora de almuerzo (entre 13:00 y 15:00 horas); Posteriormente, un nuevo peak se presenta entre 15:00 y 17:00 horas. Esta configuración determina la Política de Creación de Instancias, la cual se detalla en el Capítulo 5. Cabe destacar que en las zonas de menor frecuencia, los llamados son tan escasos que no logran formar el patrón observado.

Para cada llamada, hay asociado un cliente que posee un tipo particular de máquina a reparar. De la base de datos, podemos rescatar que existen 18 familias diferentes de máquinas, cada una de las cuales tiene un tiempo de atención promedio asociado, los cuales se encuentran en la tabla 4.5. Se observa que 6 de estos grupos concentran más del 80 % de los llamados (Souyris[19]), y que la varianza asociada a los tiempo de reparación logra superar en algunos casos el 70 %. Esto implica una gran variabilidad en los tiempos de atención, lo que lleva a pensar que, para no aumentar de forma innecesaria la dificultad del problema, los tiempos de atención para el caso de estudio serán determinísticos e iguales al tiempo promedio de atención de la familia a la cual cada máquina pertenece.

Tipo Producto	Tiempo en min	$\sigma(t)$	$\sigma(t)$ en %	Llamados Anuales	%Llamados
ODP-CNV L-L	84,86	47,21	55,6	12454	37,33
ODP-CNV DIG	81,3	46,55	57,3	3678	11,03
CH-LASER	54,04	35,38	65,5	3455	10,36
ODP-WG	106,35	54,28	51	3233	9,69
ODP-DCS WG	88,11	45,81	52	2895	8,68
CH-MULTIF.	46,01	29,99	65,2	1134	3,4
ODP-DTL	122,25	63,83	52,2	1018	3,05
ODP-COLOR	128,47	69,53	54,1	931	2,79
PS	149,41	66,94	44,8	912	2,73
XES	125,92	67,56	53,7	724	2,17
CH-FAX	41,79	24,18	57,9	591	1,77
CH-INKJET	38,43	22,44	58,4	509	1,53
CH-COP. PERSONAL	67,67	48,97	72,4	466	1,4
ODP-PRO	53,15	32,76	61,6	391	1,17
ODP-DCS DPTL	110,88	55,6	50,1	377	1,13
DPS	126,79	73,97	58,3	327	0,98
CH-LASER COLOR	68,75	44,9	65,3	255	0,76
CGS	69,44	54,57	78,6	9	0,03

Tabla 4.5: Estadística de llamados por tipo de impresora.

Del trabajo realizado por Leng[15], se conoce cuál es la flota óptima para cada mes de operación, lo cual se calculó por medio de un modelo basado en simulación y en *Óptimos de Pareto*. De acuerdo a una política de nivel de servicio del 15%², la cantidad de técnicos para cada mes del año corresponde a una combinación lineal de los técnicos sugeridos por este trabajo. Dicha combinación se justifica por el hecho que en el presente trabajo se utiliza una zona ligeramente distinta a la demarcada en el trabajo de Leng[15], para poder observar de mejor manera como se comportan las rutas a nivel espacial y para evitar la concentración excesiva de técnicos en las comunas de Santiago Centro y Providencia, identificadas anteriormente como las comunas de mayor actividad económica y, por tanto, con mayor cantidad de llamados.

En la tabla 4.6, se muestra el resultado de la regresión lineal utilizada, la cual tiene la forma

$$\text{N}^\circ \text{ Técnicos Totales} = \text{Dens}(\text{Area101}) * \text{Llamados}(\text{Area 101}^*) + \text{Dens}(\text{Area110}) * \text{Llamados}(\text{Area110}^*)$$

Donde $\text{Dens}(\star)$ corresponde a la densidad de técnicos por llamados del Area \star .

Mes	Zona 101	Zona 110	Zona Utilizada
Enero	10	10	17
Febrero	10	9	17
Marzo	10	9	17
Abril	10	10	19
Mayo	9	10	19
Junio	10	9	17
Julio	10	10	19
Agosto	10	9	17
Septiembre	10	9	17
Octubre	10	10	19
Noviembre	10	9	17
Diciembre	9	9	17

Tabla 4.6: Numero de técnicos a utilizar por mes.

²Lo que implica que, en promedio, un 15% de los clientes quedan sin servicio el mismo día de la llamada

Para mayor claridad, a continuación en la tabla 4.7 se detalla la configuración mostrada por cada una de las áreas utilizadas en los cálculos anteriores.

Area	Comunas
Area 101	Santiago Centro, Providencia, Vitacura, Las Condes
Area 101*	Vitacura, Las Condes
Area 110	Cerrillos, El Bosque, La Cisterna, La Florida, La Granja, La Pintana, Lo Espejo, Maipu, P.A. Cerda, Padre Hurtado, Puente Alto, San Bernardo, San Joaquin, San Miguel, San Ramon
Area 110*	Cerrillos, El Bosque, La Cisterna, La Florida, La Granja, La Pintana, Lo Espejo, Maipu, P.A. Cerda, Padre Hurtado, Puente Alto, San Bernardo, San Joaquin, San Miguel, San Ramon, La Reina, Macul, Ñuñoa, Peñalolen

Tabla 4.7: Distribución de Comunas en áreas de trabajo.

Capítulo 5

Resultados Numéricos

El Capítulo a continuación muestra la implementación del modelo presentado en el Capítulo 3, haciendo énfasis en el software utilizado y las diferentes técnicas implementadas para mejorar la performance del código. Posteriormente, se muestra la Política de Construcción de instancias para luego detallar las Corridas Preliminares realizadas con el objetivo de decidir la mejor forma de enfrentar el problema propuesto por el Capítulo 4. Finalmente, se muestran los resultados de las corridas realizadas y se comparan dichos resultados con otras técnicas para verificar si se logran o no mejores resultados.

5.1. Implementación

La disyuntiva que presenta el modelo presentado es la necesidad de implementar Generación de Columnas. Si bien muchos softwares de optimización poseen las interfaces para implementar Generación de Columnas sin mayor problema (COIN BCP, ABACUS, ILOG CPLEX, entre otros), se decide utilizar Dynadec COMET (Bent [2]), un potente sistema de optimización híbrida, que combina constraint programming, local search y programación lineal y entera. Se utiliza este sistema porque su lenguaje se basa en C++, lenguaje de bajo nivel que permite más libertad de progra-

mación, y porque la posibilidad de utilizar constraint programming permite implementar modelos no lineales, lo cual tiene vital importancia para el modelo presentado en el Capítulo 6.

5.2. Política de Creación de Instancias

Como se menciona en el Capítulo 4, se necesita resolver el problema de despacho dinámico técnicos que enfrenta el Servicio de Reparación de Fallas de Xerox Chile en la ciudad de Santiago. Para esto, se dispone de una base de datos que contiene todos los llamados recibidos por el Servicio de Reparación de Fallas durante el período comprendido entre Mayo de 2002 y Abril de 2003. La base de datos contiene información completa sobre la llamada recibida, incluyendo el momento de recepción de la llamada, dirección del cliente, comuna, tiempo de atención, etc. Un detalle de los atributos de la base de datos se detalla en el ANEXO XXXX.

Los llamados de los clientes llegan entre las 7 y las 19, y uno de los problemas importantes del presente trabajo es decidir cómo se construyen las instancias a ser utilizadas por la heurística propuesta.

De acuerdo a la distribución de los llamados diarios mostrados en el Capítulo 4, y considerando que se busca la implementabilidad real del método de solución propuesto, se decide dividir un día de operación en 4 partes:

1. **Llamados pendientes** Existe un conjunto de llamados que no alcanzaron a ser ruteados el día en el que se recibió la llamada, ya sea por insuficiencia de la operación o por lo tarde de los llamados (en general, llamados recibidos entre 15 y 19 hrs). Estos llamados serán ruteados de forma estática, siendo atendidos a primera hora de la mañana.
2. **Primeros llamados del día**

Mientras se ejecuta ruteo base para los llamados pendientes del día anterior, digamos entre 7 y 10 hrs, una serie de nuevos llamados van llegando, los cuales se insertan dinámicamente

en la rutas ya existentes. Con estas dos primeras etapas se tienen llamados suficientes para cubrir la operación al menos hasta el mediodía.

3. Llamados mediodía

De forma equivalente a la primera parte, entre las 10 y 12 hrs se obtiene una serie de llamados que serán parte de un nuevo ruteo estático, el cuál será ejecutado al finalizar la ejecución de los llamados anteriores.

4. Llamados tarde

Finalmente, los llamados que llegan entre las 12 y las 15 hrs, durante la ejecución del ruteo estático de los llamados de mediodía, son insertados dinámicamente.

De acuerdo a esta estructura, un día de operación es tratado como 2 conjuntos de llamados independientes, lo cual se ejecuta por facilidad del estudio en curso en cuanto a tener un mix equilibrado de llamados estáticos y dinámicos que aproveche la estructura de la heurística propuesta.

5.3. Corridas Preliminares

De acuerdo a la heurística presentada en el Capítulo 3, para resolver el problema anteriormente enunciado, es necesario decidir sobre las diferentes opciones que esta metodología presenta. En particular, los tópicos más relevantes son:

- Función objetivo a utilizar dentro de la formulación del Ruteo de Técnicos con Ventanas de tiempo.
- Generar nodos ficticios por medio de K-means o totalmente al azar dentro de la formulación del Ruteo de Técnicos con Ventanas de tiempo.
- Utilizar Generación de Columnas Dinámico o un metodo Greedy para el ruteo de llamados entrantes.

Para resolver esta disyuntiva, se decide hacer una serie de corridas preliminares para tener un criterio objetivo sobre cada una de estas decisiones: se utilizará aquella opción que presente un mejor resultado en cuanto a costos totales, asegurando así que se busca obtener la mejor solución posible para el problema, desde el punto de vista de la heurística propuesta.

5.3.1. Función Objetivo

A continuación se muestra la función objetivo utilizada en el problema de programación lineal presentado en el Capítulo 3.

$$\min_{x,w,\delta,f} \sum_{k \in K} \sum_{i \in I_2} \delta_{i,k} + \alpha \sum_{k \in K} \sum_{i \in I_1 \cup I_2} \lambda_{i,k} - \beta \sum_{z \in Z} d_z f_z + \varphi \sum_{k \in K} \sum_{(i,j) \in A} t_{i,j} x_{i,j}^k$$

Se puede observar que la función comprende 4 componentes:

- **Violación de la Ventana de Tiempo:**

$$\sum_{k \in K} \sum_{i \in I_2} \delta_{i,k}$$

Dado el enfoque de calidad de servicio, se considera la componente más importante de la función objetivo. Debido a esto, se considera 1 minuto de violación de ventana de tiempo como el mayor costo en el que puede incurrir una ruta.

- **Espera de Clientes:**

$$\sum_{k \in K} \sum_{i \in I_1 \cup I_2} \lambda_{i,k}$$

Corresponde a la suma de los minutos que espera cada cliente antes que lleguen a atenderlo. Su coeficiente, α , corresponde a un número entre 0 y 1, pero que, dada la importancia de dicho costo en la calidad de servicio, se suele fijar en números mayores a 0,5.

- **Covering Espacial:**

$$\sum_{z \in Z} d_z f_z$$

Corresponde al beneficio que se obtiene al poner nodos ficticios en las zonas de mayor densidad de clientes. Su coeficiente, β , es fijo e igual a 0,25, para que dicho beneficio no supere el 10% de los costos, de forma que sea influyente en la FO, pero no determinante.

▪ **Costo de Transporte:**

$$\sum_{k \in K} \sum_{(i,j) \in A} t_{i,j} x_{i,j}^k$$

Comprende el tiempo total de desplazamiento de cada uno de los vehículos ruteados. Dependiendo de la forma en la que se enfrenta el problema, su coeficiente, φ , oscila entre 0,1 y 0,5.

Dado lo anterior, lo que hay que decidir es cual es el valor de los coeficientes α , β y φ . A partir de corridas preliminares, es posible verificar que mientras más grande es φ respecto a α , las rutas tienden a ser menos equilibradas en cuanto a número de clientes visitados, produciéndose rutas con muchos nodos pero en zonas muy compactas. Esto se explica porque prevalece el hecho de contar con viajes cortos por sobre la espera del cliente, tal como lo muestra la Figura 5.1.

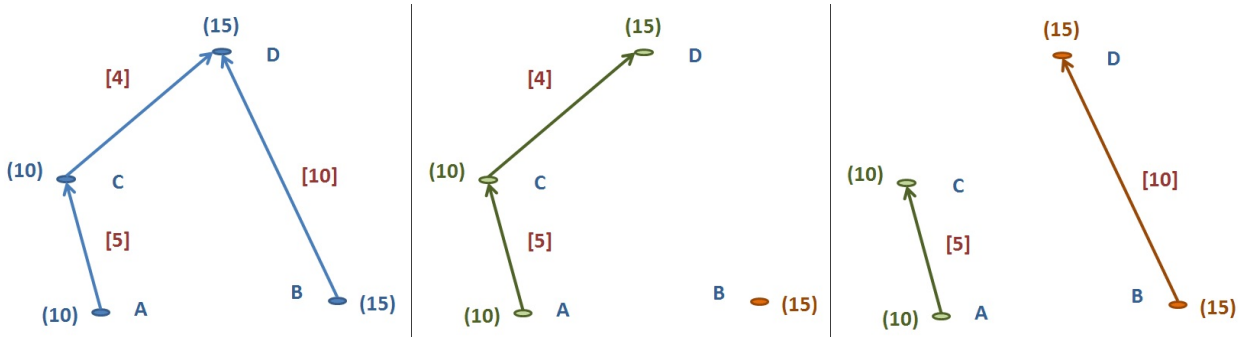


Figura 5.1: Ejemplo de influencia de coeficientes en Función Objetivo

Para la red mostrada en la primera parte de la Figura 5.1, se pide hacer 2 rutas de costo mínimo, considerando los costos de transporte (entre corchetes) y los costos de espera: lo que demora un vehículo en llegar a cada nodo. Los costos entre paréntesis corresponden a tiempos de servicio.

Si los coeficientes de los costos de espera y los de transporte son iguales a 1, el costo de la solución mostrada en la segunda parte de la figura corresponde a 53 (9 en transporte total, 15 por la

espera del nodo “C” y 29 por la espera del nodo “D”), mientras que el costo de la segunda solución mostrada corresponde a 55 (15 en transporte, 15 del nodo “C” y 25 del nodo “D”). En cambio, si los costos de espera son 2 veces más importantes que los de transporte, digamos coeficientes 1 y 0,5 respectivamente, el costo de la primera solución corresponde a 48,5, mientras que la segunda tiene un costo total de 47,5. De esto se corrobora la intuición respecto a que a mayor importancia de los costos de espera de los clientes se logra un mayor equilibrio en el número de nodos por ruta.

Dado el enfoque de calidad de servicio al que apunta este trabajo, se decide utilizar rutas equilibradas, por lo cuál se establece que la relación entre φ y α será de 2 es a 1, o sea, $\varphi = 0,5$ y $\alpha = 0,25$.

5.3.2. Nodos Ficticios

5.3.3. Generación de Columnas Dinámico

5.4. Resultados Numéricos

A continuación, se detallan los detalles de las corridas definitivas. En particular, se detallan las instancias ejecutadas, las heurísticas utilizadas como comparación, los resultados obtenidos y algunas conclusiones respecto a ellos.

5.4.1. Instancias Ejecutadas

De la base de datos disponible, se seleccionan 13 semanas, equivalentes a 65 días de operación (sólo se trabaja durante días hábiles). Dada la estructura de las instancias (detallada anteriormente), estos 65 días de operación se convierten en 130 instancias estáticas, cada una con su respectivo set de llamados dinámicos. Las instancias se construyen a intervalos de horas fijos, pero el número de nodos a rutear es variable. Esto se considera beneficioso para el estudio, pues es posible observar el comportamiento de la heurística para diferentes números de nodos, tanto estáticos como dinámicos.

5.4.2. Heurísticas de Comparación

Considerando la particularidad del problema a resolver, es necesario implementar otros métodos de resolución para comparar los resultados obtenidos por la Heurística propuesta. Dada la naturaleza del problema, se implementan los siguientes métodos:

- **TTP - Greedy:** Corresponde a la versión dinámica más intuitiva, considerando la capacidad de realizar un ruteo mediante un MIP. Esta heurística rutea los llamados pendientes de forma estática, y los llamados que llegan dinámicamente los agrega al final de la ruta con menor costo, de acuerdo a la función objetivo utilizada. En esta versión no se utilizan nodos ficticios.
- **TTP - DCG:** Corresponde a una versión simplificada de la heurística propuesta, la diferencia radica en el hecho de no poseer nodos ficticios. En esta heurística se utiliza Generación de Columnas Dinámica, por lo cual, al llegar un llamado se altera un subconjunto de las rutas obtenidas mediante la generación de columnas con el objetivo de generar nuevas columnas, las que serán utilizadas para volver a resolver el MIP.

En una primera instancia, se espera que la Heurística Propuesta obtenga mejores resultados que el TTP - Greedy (ruteo miope), pero no se tiene certeza sobre la relación entre el SCTTP - DCG y el TTP-DCG. Si bien el TTP-DCG realiza de por sí menos viajes (se ahorra el traslado a los nodos ficticios) no tiene la anticipación a los nuevos llamados que genera el SCTTP-DCG.

5.4.3. Resultados

Se ejecutan las 130 instancias en un computador Intel Core I3 (3M Cache, 2.13 GHz de 2 núcleos dobles) de 8 gigas de Ram, obteniéndose los resultados detallados en el ANEXO (XXX) para el SCTTP - DCG, el TTP-DCG y el TTP - Greedy.

Una vez implementado el modelo, se le agrega la opción de generar de forma inmediata las rutas con 1 y 2 vehículos. Con esto se logra reducir el tiempo de corrida para el Ruteo de Técnicos

con Ventanas de Tiempo.

Un resumen de dichos resultados se muestra en la tabla 5.1

Implementación	Tiempo(sec)	TW (min)	Transporte (min)	Espera (min)	Total (min)
TTP-Greedy	207.7	0	438.9	2860.9	1540.2
TTP-DCG	332.9	0	359.3	2719.2	1449.4
SCTTP-DCG	843.7	0	540.9	2629.5	1465.4

Tabla 5.1: Promedio de Costos de Instancias Corridas.

De la tabla anterior, se puede observar que, en promedio, tanto el TTP-DCG como el SCTTP-DCG consiguen costos menores que el TTP-Greedy. Este comportamiento corresponde a lo esperado, básicamente porque el TTP-Greedy corresponde a la implementación operacional más básica que se puede utilizar, la cual no aprovecha la posibilidad de insertar nodos dentro de las rutas (sólo al final) y tampoco utiliza la información que se tiene del problema. Se observa además, que es posible encontrar una ruta de costo mínimo que no incurra en violar la restricción de Ventanas de Tiempo para las tres implementaciones.

Del detalle de los resultados, se puede ver que si comparamos los costos del SCTTP-DCG y el TTP-Greedy, no existe ningún patrón marcado respecto a en qué tipo de instancias gana una respecto a la otra, pero se puede ver (Figura 5.2) que SCTTP-DCG tiene, en promedio, un un costo 2,6% menor que TTP-Greedy.

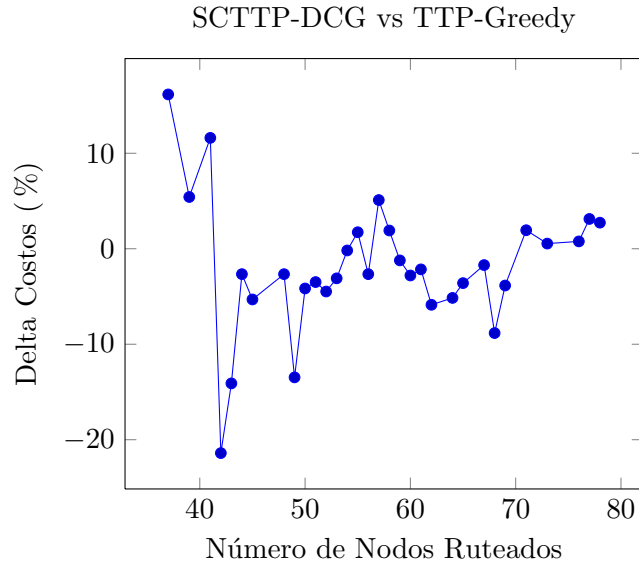


Figura 5.2: Brecha de costos en función del número de nodos.

Por otra parte, si comparamos los costos del TTP-DCG y el TTP-Greedy, TTP-DGC tiene mejor comportamiento en casi todas las instancias, lo cual se puede ver en la Figura 5.3.

Respecto a la comparación de costos entre SCTTP-DCG y TTP-DCG, podemos observar que, en promedio, TTP-DGC muestra un 5,9% menos de costos que el SCTTP-DCG. Una primera explicación se basa en entender el efecto que tiene el uso de nodos ficticios sobre una ruta: si bien se utiliza la información para antelarse a futuros llamados, el hecho de trasladarse a dichos nodos implica un costo extra que no posee el TTP-DCG. Bajo la misma premisa, el ruteo estático siempre debería tener menor costo para el TTP-DCG, pero a medida que llegan los llamados ficticios, la brecha entre los costos debería acortarse hasta llegar a un punto donde SCTTP-DCG logre mejores resultados que TTP-DCG.

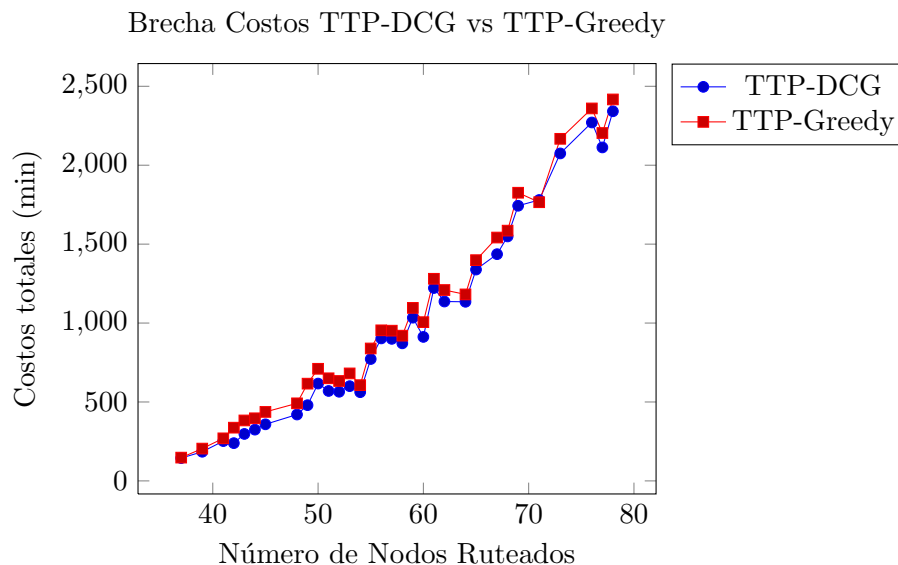


Figura 5.3: Comparación de costos totales TTP-DCG vs TTP-Greedy.

La figura 5.4 muestra la brecha de costos finales entre SCTTP-DCG y TTP-DCG ordenados por el número de nodos de la instancia.

Claramente, se puede observar que la brecha se reduce a medida que el número de nodos aumenta, lo que tiene estrecha relación con la ganancia en costos que muestra SCTTP-DCG a medida que se rutean los nuevos llamados. Además, se ve que existe una zona donde SCTTP-DCG posee mejores costos que TTP-DCG. Comprendida entre los 60 y los 70 nodos ruteados, dicha zona muestra la acotada efectividad de la heurística propuesta.

Dado el análisis anterior, el siguiente paso de la investigación corresponde a descubrir como ampliar la zona donde es efectivo el SCTTP-DCG. Las dos opciones que se barajan son:

- Covering Tiempo-Espacial.** La idea es ampliar el concepto de covering usado en el SCTTP-DCG para todo momento del tiempo, logrando así que a medida que mientras más llamados ficticios lleguen mejor se aproveche la reducción en tiempo de espera que genera el covering. Dicha idea se desarrolla en el Capítulo 6, donde se enuncia, se implementa y se muestran resultados del Problema de Ruteo de Técnicos con Covering Tiempo Espacial, o TSCTTP

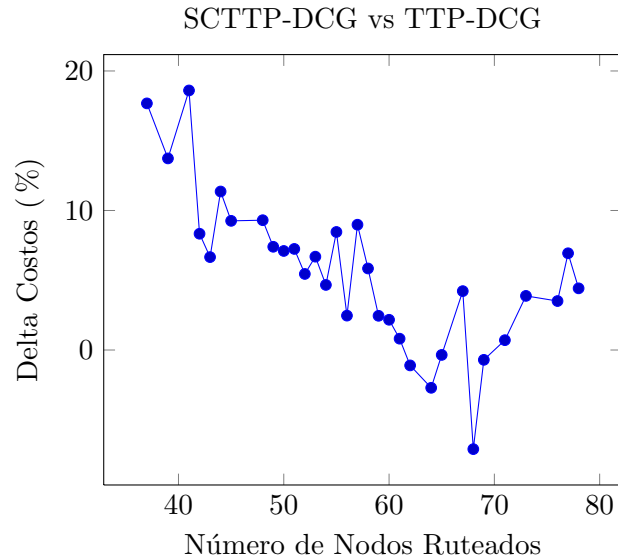


Figura 5.4: Brecha de costos en función del número de nodos.

(por su sigla en inglés).

- **Reducción de tiempos de viaje.** La idea es ampliar la zona de la curva donde gana SCTTP-DCG bajando la brecha de costos respecto al TTP-DCG. Esto se realiza eliminando la relocalización de los técnicos siempre y cuando el nodo ficticio esté en la misma zona que el nodo que lo precede.

Los detalles de esta última idea se muestran a continuación.

SCTTP-DCG sin Relocalización Innecesaria

Tal como se detalla en el Capítulo 4, una sub-zona es una de las 49 partes que conforman la zona de Santiago donde se realiza este estudio. Dado que el radio promedio de cada sub-zona es de 1,7 kms., el tiempo de viaje dentro de la zona está entre los 5 y los 10 min. Si el nodo ficticio de una ruta se encuentra en la misma zona que el predecesor (situación que ocurre la mayoría de las veces dada la minimización de costos de transporte), el técnico debe viajar estos 5 a 10 min dentro de la zona para situarse en el centro de la misma. La idea es eliminar dicho viaje, procurando que

el técnico espere la asignación de un llamado dinámico en el último nodo real atendido siempre y cuando la localización del nodo ficticio sucesor sea la misma que la del nodo atendido. En caso contrario, el técnico se mueve al nodo ficticio asignado.

El detalle de la comparación entre los costos de esta nueva implementación, llamada SCTTP-DCG-WLT (del inglés "Without Last Travel") y el TTP-DCG se muestran en la figura 5.5

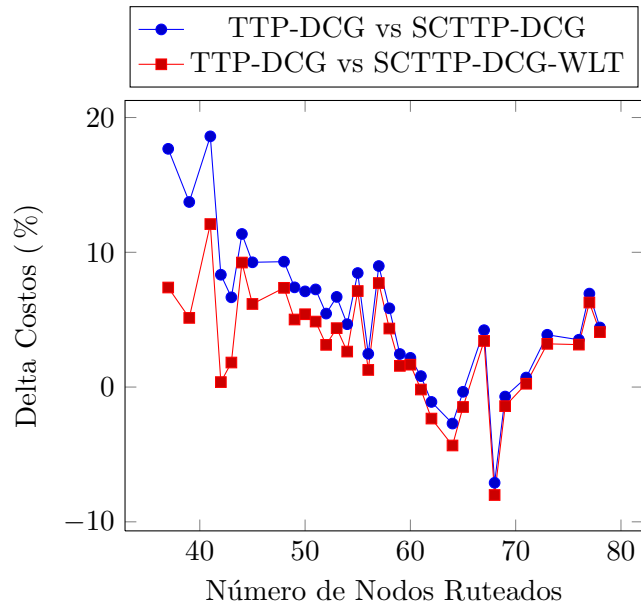


Figura 5.5: Brecha de costos en función del número de nodos.

Del gráfico, se puede observar que existe una baja importante en los costos del SCTTP-DCG, principalmente en las instancias con menos número de nodos, brecha que se va acortando a medida que el número de nodos aumenta. Esto se explica porque para las rutas con menos nodos es más probable llegar al nodo ficticio antes que un llamado dinámico lo reemplace. Si bien esta última implementación reduce la brecha, aún no se logran los resultados esperados. Debido a eso, en el siguiente Capítulo 6 se desarrolla la idea del Problema de Ruteo de Técnicos con Covering Tiempo Espacial.

Capítulo 6

Problema de Ruteo de Técnicos con Covering Tiempo Espacial

A partir de los resultados mostrados en el Capítulo 5, nace la pregunta si es posible entelar los llamados de los clientes de forma más efectiva y así tener una reducción de costos más significativa o, por lo menos, una que se presente en un mayor número de instancias. Esta inquietud promueve la extensión del modelo ya presentado a uno que no sólo asegure la correcta distribución de las rutas al comienzo de la operación, sino uno que tenga en consideración dicha distribución durante todo momento. A continuación, se detalla la idea del nuevo modelo y su posterior implementación, resultados y los desafíos futuros que presenta.

6.1. Ruteo de Técnicos con Covering Tiempo Espacial

Observando el Ruteo de Técnicos con Covering Espacial presentado en el Capítulo 3, la mejora natural a este modelo es pedirle al problema de optimización que maximize no sólo el problema de covering asociado a los nodos finales de las rutas, sino que considere dentro de su función objetivo el covering a cada momento. Esta idea se basa en el trabajo de Weintraub [20] para

el ruteo de vehículos de emergencia de Chilectra. Como se puede ver en la figura 6.1, este enfoque intenta reducir las ineficiencias presentes en momentos intermedios (entre el comienzo de la ruta y la llegada al nodo ficticio), donde no se tiene cuantificado si los nodos están bien o mal distribuidos espacialmente.

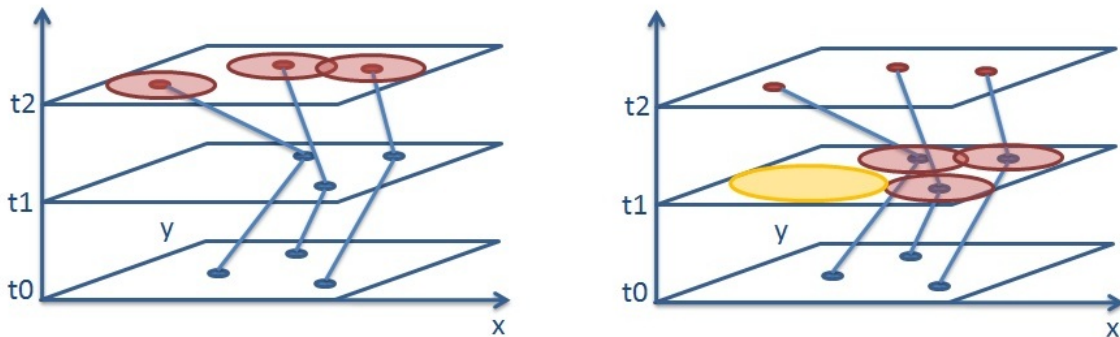


Figura 6.1: Espacio de mejora en Ruteo de Técnicos con Covering Tiempo Espacial.

Para esto, se crea un modelo que considera una matriz de covering Tiempo-Espacial, la cual indica si existe algún vehículo (o ruta, para el Problema Maestro) en la zona indicada para un momento determinado. Dado que una matriz como la descrita que considere el desarrollo minuto a minuto es inmanejable, se considera un covering mínimo por bloques horarios de 30 minutos.

La idea es mantener la estructura del modelo anteriormente implementado, pero agregando en el Problema Maestro las restricciones de covering Tiempo-Espacial y agregando en el Problema de Pricing los costos duales de las nuevas restricciones asociadas.

Por otra parte, se conserva la utilización de los nodos ficticios, pero ahora se utilizan para generar covering en todos los períodos considerados para la resolución los cuales no tengan asociados nodos reales. Cabe destacar que el número de períodos considerados para la optimización depende linealmente de la cantidad de llamados que estén ingresados al sistema.

6.2. Formulación

6.2.1. Problema de Programación Lineal

- I_0 : Conjunto de nodos ficticios generados
 I_1 : Conjunto de nodos iniciales
 I_2 : Conjunto de nodos a ser ruteados dentro del día
 Z : Conjunto de zonas.
 F : Hora en que finaliza la jornada laboral
 K : Número de técnicos disponibles
 A : Conjunto de todos los arcos existentes
 P : Conjunto de bloques de tiempo
 l_i : Momento de la llamada relativa a nodo i
 b_i : Ventana de tiempo superior para cada nodo
 s_i : Tiempo de reparación promedio para cada nodo
 u_i : posición del cliente i
 $d_{z,p}$: Densidad de clientes en zona z en bloque de tiempo p
 $t_{i,j}$: Tiempo de viaje entre el nodo el cliente i y el cliente j
 $Cov_{i,z}$: Cobertura que el nodo ficticio i ofrece a la zona z
 $MinCov_z$: Cobertura mínima requerida para la zona z

$$A = \{(i, j) / i \in I_1 \cup I_2, j \in I_0 \cup I_2, i \neq j\}$$

w_i^k : Variable de Decisión. Momento en que técnico k llega a nodo i

δ_i^k : Variable de Decisión. Violación de tiempo de llegada del técnico k al nodo i

$f_{z,p}$: Variable de Decisión. Cobertura ofrecida a la zona z en el bloque de tiempo p

$x_{i,j}^k$: Variable de Decisión

$$x_{i,j}^k = \begin{cases} 1 & \text{si técnico } k \text{ atiende a nodo } j \text{ inmediatamente después de nodo } i \\ 0 & \text{en caso contrario} \end{cases}$$

λ_i^k : Variable Auxiliar. Demora en atender al nodo i dentro de ventana de tiempo

$$\lambda_i^k = \begin{cases} (w_i^k - l_i - \delta_i^k) & \text{si } w_i^k \text{ es distinto de } 0 \\ 0 & \text{en otro caso} \end{cases}$$

$TS_{j,p}^k$: Variable Binaria. Si el vehículo k está en el cliente j en el bloque de tiempo p

$$TS_{j,p}^k = \begin{cases} 1 & \text{si } w_j^k \leq B_p < w_j^k + s_j \\ 0 & \text{en otro caso} \end{cases}$$

Donde B_p es el momento en que el bloque de tiempo p empieza.

Usando los parámetros α , β and φ (with $\alpha \in [0, 1]$, $\beta \in [0, 1]$, $\varphi \in [0, 1]$) y las variables de decisión definidas, el problema queda expresado como:

Se define el parámetro W_i^k como:

$$W_i^k = \begin{cases} 1 & \text{si } i = k, \text{ con } k \in K \\ 0 & \text{en otro caso} \end{cases}$$

El parámetro $N \gg 1$ es un parámetro auxiliar usado como cota superior de la restricción (6.7). La función objetivo (6.1) toma la penalidad asociada a la violación de la ventana de tiempo

$$\min_{x,w,\delta,f} \sum_{k \in K} \sum_{i \in I_2} \delta_i^k + \alpha \sum_{k \in K} \sum_{i \in I_1 \cup I_2} \lambda_i^k - \beta \sum_{z \in Z, p \in P} d_{z,p} f_{z,p} + \varphi \sum_{k \in K} \sum_{(i,j) \in A} t_{i,j} x_{i,j}^k \quad (6.1)$$

$$\text{s.a.} \quad \sum_{k \in K} \sum_{j: (i,j) \in A} x_{i,j}^k = 1 \quad \forall i \in I_1 \cup I_2 \quad (6.2)$$

$$\sum_{(i,j) \in A / j \in I_0} x_{i,j}^k = 1 \quad \forall k \in K \quad (6.3)$$

$$\sum_{j: (i,j) \in A} x_{i,j}^k - \sum_{j: (j,i) \in A} x_{j,i}^k = W_i^k \quad \forall k \in K, i \in I_1 \cup I_2 \quad (6.4)$$

$$w_i^k \leq F \sum_{j: (j,i) \in A} x_{j,i}^k \quad \forall i \in I_0, k \in K \quad (6.5)$$

$$w_i^k - \delta_i^k \leq b_i \quad \forall i \in I_2, k \in K \quad (6.6)$$

$$w_i^k + s_i + t_{i,j} - w_{j,k} \leq (1 - x_{i,j}^k) N \quad \forall (i,j) \in A, k \in K \quad (6.7)$$

$$\sum_{k \in K} \sum_{j: (i,j) \in A} TS_{j,p}^k Cov_{j,z} \geq f_{z,p} \quad \forall z \in Z, \forall p \in P \quad (6.8)$$

$$f_{z,p} \geq MinCov_{z,p} \quad \forall z \in Z, \forall p \in P \quad (6.9)$$

como costo primario y el costo de transporte de los vehiculos, y el tiempo de espera de los clientes como costo secundario. Además, la función objetivo tiene un premio por incrementar la cobertura de las zonas. La restricción (6.2) exige que cada nodo sea visitado una vez. La restricción (6.4) asegura la conservación de flujo de técnicos. La restricción (6.5) exige que cada ruta llegue a su último nodo antes del fin del período de trabajo. La restricción (6.6) fija el valor de la variable relacionada con la violación de las ventanas de tiempo, mientras la restricción (6.7) asegura la consistencia temporal de las rutas. Las restricciones (6.8) y (6.9) aseguran que la cobertura total por zona sea mayor al mínimo exigido en cada bloque de tiempo.

6.2.2. Descomposición para Generación de Columnas

Problema Maestro

El problema maestro considera un problema de *Set Partitioning*, y se formula de la siguiente manera:

- R^* : Conjunto de todas las rutas generadas
- θ_r : Variable de decisión binaria; se usa o no la ruta r
- C_r : Costo asociado a ruta r
- $H_{i,r}$: Matriz binaria; nodo i se encuentra en ruta r , o no
- $Cov_{z',z}$: Cobertura que la zona z' ofrece a la zona z
- $TS_{r,z',p}$: Matriz Binaria; si la ruta r está en la zona z' en el bloque de tiempo p

$$\min_{\theta, f} \sum_{r \in R^*} C_r \theta_r - \beta \sum_{z \in Z, p \in P} d_{z,p} f_{z,p} \quad (6.10)$$

$$\text{s.a.} \quad \sum_{r \in R^*} \theta_r \leq K \quad (6.11)$$

$$\sum_{r \in R^*} H_{i,r} \theta_r \geq 1 \quad \forall i : (i, j) \in A \quad (6.12)$$

$$\sum_{r \in R^*} \theta_r \sum_{z' \in Z} TS_{r,z',p} Cov_{z',z} \geq f_{z,p} \quad \forall z \in Z, \forall p \in P \quad (6.13)$$

$$f_{z,p} \geq MinCov_{z,p} \quad \forall z \in Z, \forall p \in P \quad (6.14)$$

La función objetivo (6.10) minimiza los costos totales de las rutas utilizadas, añadiendo un

premio por la cobertura lograda por cada ruta. La restricción (6.11) asegura que el número de rutas usadas no excede al número de vehículos disponibles. La restricción (6.12) exige que cada nodo sea visitado al menos una vez. Las restricciones (6.13) y (6.14) aseguran que las rutas generen una cobertura mínima de las zonas para cada bloque de tiempo. Se define π como el costo dual asociado a la restricción (6.15), mientras que γ_i es el costo dual asociado a la restricción (6.12). Además, se define la matriz $\psi_{z,p}$, que representa los costos duales de la restricción (6.14).

Problema de Pricing

La siguiente formulación genera una ruta de costo mínimo, considerando los costos duales obtenidos en el Problema Maestro. Una vez obtenida, la ruta se agrega al conjunto general después de cada iteración del algoritmo.

Para la correcta implementación del algoritmo, es necesario definir las funciones $\Gamma(x_{i,j})$ and $\Theta(w_i)$ para asegurar la correspondencia entre los costos duales del Problema Maestro y las variables de Problema de Pricing.

$$\Gamma(x_{i,j}) = \begin{cases} u_i & \text{si } x_{i,j} > 0 \\ 0 & \text{en otro caso} \end{cases} \quad \Theta(w_i) = \begin{cases} \lfloor w_i/\kappa \rfloor & \text{si } x_{i,j} > 0 \\ 0 & \text{en otro caso} \end{cases}$$

Donde κ es el largo de un bloque de tiempo medido en minutos.

$$\kappa = F/\text{card}(P)$$

Con estas funciones, es posible asociar las variables duales de la restricción (14) a los subíndices del Problema de Pricing.

$$\min_{x,w,\delta} \quad \pi + \sum_{i \in I_1 \cup I_2} \delta_i + \alpha \sum_{i \in I_1 \cup I_2} \lambda_i + \sum_{(i,j) \in A} t_{i,j} x_{i,j} \quad (6.15)$$

$$- \sum_{i:(i,j) \in A} x_{i,j} - \sum_{i:(i,j) \in A, z \in Z} \Psi_{\Gamma(x_{i,j}), \Theta(w_i)} \text{Cov}_{\Gamma(x_{i,j}), z}$$

$$\text{s.a.} \quad \sum_{i \in I_1} \sum_{j \in I_2} x_{i,j} = 1 \quad (6.16)$$

$$\sum_{j:(i,j) \in A} x_{i,j} - \sum_{j:(j,i) \in A} x_{j,i} = W_i^* \quad \forall i : (i,j) \in A \quad (6.17)$$

$$w_i \leq F \sum_{j:(j,i) \in A} x_{j,i} \quad \forall i \in I_0 \quad (6.18)$$

$$w_i - \delta_i \leq b_i \quad \forall i \in I_2 \quad (6.19)$$

$$w_i + s_i + t_{i,j} - w_j \leq (1 - x_{i,j})N \quad \forall (i,j) \in A \quad (6.20)$$

$$\sum_{i \in I_0} \sum_{j \in I_0 \cup I_1 \cup I_2} x_{j,i} \leq M \quad (6.21)$$

Parámetro W_i^* es definido como:

$$W_i^* = \begin{cases} 1 & \text{si } i \in I_1 \\ -1 & \text{si } i \in I_0 \\ 0 & \text{en otro caso} \end{cases}$$

Cabe destacar que la formulación recién presentada es altamente no lineal, por lo cual, su implementación computacional puede ser desde engorrosa hasta imposible. Por esta situación, se refuerza la idea de implementarla en *Constraint Programming*, donde es posible utilizar estructuras de esta índole sin mayor esfuerzo computacional.

6.3. Resultados

Capítulo 7

Conclusiones

7.1. Conclusiones Generales

Se logra desarrollar e implementar un modelo capaz de resolver el Problema de Ruteo Dinámico de Vehículos, teniendo como principal objetivo la minimización del tiempo de espera del cliente e utilizando la información histórica del problema. Esto se realiza por medio de la incorporación de nodos de espera ficticios al problema de ruteo estático, y de resolver el posterior problema de inserción de nuevos llamados por medio de la técnica llamada “Generación de Columnas Dinámico” (DCG por su sigla en inglés). Dicha técnica utiliza las bondades del método de Generación de Columnas implementado para la resolución estática del problema para dar soluciones rápidas y de muy buenos resultados.

Se destaca la implementación y uso exitoso del DCG dentro de la inserción dinámica, puesto que es una técnica enunciada en 2006 y, a pesar de eso, ha sido poco utilizada en la resolución de problemas de ruteo en la literatura. Se logra con ella la inserción rápida y eficaz de los llamados dinámicos, logrando buenos resultados en comparación a la resolución MIP.

En cuanto a los resultados numéricos, el método propuesto, SCTTP-DCG, obtiene mejores

resultados que la implementación dinámica más intuitiva realizable: el ruteo estático de nodos más la agregación Greedy de llamados dinámicos (TTP-Greedy). Por otra parte, el TTP-DCG (ruteo estático de nodos con la agregación de nodos dinámicos por medio de DCG) obtiene mejores resultados en comparación al TTP-Greedy y al SCTTP-DCG. La explicación a este fenómeno tiene que ver con 2 efectos: TTP-DCG gana las bondades de la inserción por medio de DCG y, además, ahorra en tiempos de desplazamiento pues no tiene que viajar a los nodos ficticios. Si bien era lógico que este fenómeno existiera, se esperaba que el SCTTP-DCG lograra recuperar los costos extra de transporte por medio de menor desplazamiento gracias a la buena ubicación frente a los llamados venideros. Esto sólo se produce para un espacio muy acotado de llamados, entre 60 y 70 nodos ruteados, y la posterior mejora del algoritmo (SCTTP-WLT-DCG) reduce la diferencia promedio de costos respecto a TTP-DCG, pero no logra aumentar la zona donde el desempeño es mejor que dicha técnica.

Debido a lo anterior, se concluye que es necesario un desarrollo más profundo de la noción de nodos de espera para lograr una técnica que, utilizando la información histórica del problema, logre mejores resultados. En base a eso, se propone el Trabajo Futuro propuesto a continuación.

7.2. Trabajo Futuro

Tal como se indica en el Capítulo 6, el trabajo futuro que se deriva directamente de este trabajo es la implementación exitosa del Ruteo de Técnicos con Covering Tiempo Espacial, problema que asegura que las rutas construidas cumplan con cierto criterio de covering durante toda la ejecución de las rutas.

Si bien el modelo es implementado en la etapa final de este trabajo, el tamaño del modelo (en cuanto a variables de decisión) es más grande de los que puede manejar la actual implementación, por lo cual se obtienen resultados para pocas instancias, los cuales no son satisfactorios en cuanto a función objetivo. La explicación radica en el hecho que la búsqueda de soluciones factibles implementada para el SCTTP no alcanza a encontrar buenas soluciones antes que los criterios de parada

se cumplan. La idea es reprogramar la misma idea y estructura de datos en un lenguaje de más bajo nivel, que permita un manejo más eficiente de los recursos computacionales, para así lograr resoluciones más rápidas y, se espera, de más y mayores instancias.

Para poder enfrentar el problema dinámico, es necesario además crear un método de inserción que respete los criterios de covering tiempo-espacial para asegurar que el trabajo computacional realizado no se pierde al insertar los nuevos llamados. La idea es mantener el uso de DCG, pero entendiendo que para un número grande de columnas la creación de nuevas columnas y su inserción al problema maestro aumenta los tiempos de ejecución exponencialmente, más aún para este nuevo problema, cuyas dimensiones son considerablemente más grandes que para el SCTTP-DCG.

REFERENCIAS

- [1] Cynthia Barnhart, Ellis L. Johnson, George L. Nemhauser, Martin W. P. Savelsbergh, and Pamela H. Vance. Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, 46(3):pp. 316–329, 1998.
- [2] Russell Bent and Pascal Van Hentenryck. A two-stage hybrid local search for the vehicle routing problem with time windows. Technical report, Transportation Science, 2001.
- [3] Russell Bent and Pascal Van Hentenryck. Waiting and relocation strategies in online stochastic vehicle routing. In *Proceedings of the 20th international joint conference on Artificial intelligence*, IJCAI'07, pages 1816–1821, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.
- [4] Olli Bräysy and Michel Gendreau. Vehicle routing problem with time windows, part i: Route construction and local search algorithms. *Transportation Science*, 39:104–118, February 2005.
- [5] John Gunnar Carlsson. Practical distributed vehicle routing for street-level map scanning. 2009.
- [6] Zhi-Long Chen and Hang Xu. Dynamic column generation for dynamic vehicle routing with time windows. *Transportation Science*, 40:74–88, February 2006.
- [7] J.-F. Cordeau, G. Desaulniers, J. Desrosiers, M. M. Solomon, and F. Soumis. *VRP with Time Windows*, pages 157–193. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.

-
- [8] Michel Gendreau, Francois Guertin, Jean-Yves Potvin, and Eric Taillard. Parallel tabu search for real-time vehicle routing and dispatching. *Transportation Science*, 33:381–390, April 1999.
- [9] Soumia Ichoua, Michel Gendreau, and Jean-Yves Potvin. Exploiting knowledge about future demands for real-time vehicle dispatching. *Transportation Science*, 40:211–225, May 2006.
- [10] Cordeau J-F., Laporte G., Mercier A., and Correspondence. A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 52(8):928–936, 2001.
- [11] Patrick Jaillet and Michael R. Wagner. Online Vehicle Routing Problems: A Survey. pages 221–237. 2008.
- [12] Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8):651–666, June 2010.
- [13] Gilbert Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3):345 – 358, 1992.
- [14] Allan Larsen, Oli B. G. Madsen, and Marius M. Solomon. The a priori dynamic traveling salesman problem with time windows. *Transportation Science*, 38:459–472, November 2004.
- [15] Daniel Leng. *Enfoque Basado en Simulacion y Ruteo para el Diseño de Flota del Problema de Despacho de Tecnicos con Demanda Estocastica*. DII, 2009.
- [16] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.
- [17] Stefan Ropke. Heuristic and exact algorithms for vehicle routing problems, 2005.
- [18] M.W.P. Savelsbergh. The vehicle routing problem with time windows: Minimizing route duration. *ORSA Journal on Computing*, 4(2):146–154, 1992.
- [19] Sebastian Souyris. *Enfoque Basado en Generación de Columnas y Constraint Programming para resolver el problema de Despacho dinámico de Técnicos*. DII, 2005.

- [20] Andres Weintraub, Julio Aboud, Claudio Fernandez, Gilbert Laporte, and Enrique Ramirez. Sistema para el despacho de vehículos de emergencia de una empresa eléctrica en Chile. *Revista Ingenieria de Sistemas*, 14:3–20, June 2000.

Apéndice

Apéndice A

Algoritmos

A.1. Modelo de programación

A.2. Tablas de Datos

A.3. Tablas de Resultados

A.4. Pseudocódigos para Algoritmo de Inserción Simple

A continuación, se presentan las definiciones, consideraciones y pseudocódigo para el Algoritmo de Inserción Simple (AIS), que corresponde a la primera forma de insertar las llamadas entrantes a las rutas ya obtenidas por medio del problema de optimización a priori.

A.4.1. Definiciones

Sea N el número de técnicos, defino $\mathcal{R} = \{r_1, r_2, \dots, r_N\}$ como el conjunto de rutas obtenidas a partir del Problema de Optimización a priori.

Cada ruta r_n es una clase computacional que representa un arreglo de nodos con los atributos mostrados a continuación.

Largo : Numero de nodos en la ruta, considerando nodos reales y ficticios

Costo : Costo total de la ruta

Ficticio : Boleano. Valor 1 si el último nodo de la ruta corresponde a un nodo ficticio. Valor 0

Posicion(l) : Área en la que se encuentra el nodo l . $l \in \{1, 2, \dots, Largo\}$

Ventana(l) : Ventana de tiempo perteneciente al nodo l . $l \in \{1, 2, \dots, Largo\}$

Servicio(l) : Tiempo de servicio asociada al nodo l . $l \in \{1, 2, \dots, Largo\}$

Tiempo(l) : Tiempo de llegada al nodo l . $l \in \{1, 2, \dots, Largo\}$

Llamada(l) : Tiempo de llegada de la llamada del nodo l . $l \in \{1, 2, \dots, Largo\}$

DEFINIR CLASE NEW!!!!!!

A.4.2. Algoritmos

Función *NvoCosto₁*

El algoritmo *NvoCosto₁* devuelve cuanto aumenta el costo (Int) de la ruta *r* al insertar el nuevo llamado (*new*). La función *NvoCosto₁* contiene la lógica de inserción que se utiliza en el Algoritmo de Inserción Simple.

Algorithm 1 *NvoCosto₁*

Require: Ruta *r*, Llamado *new*

```
1: if r.Ficticio=0 then
2:    $NCosto \leftarrow CostoRuta(r, new, r.Largo + 1)$ 
3: else {r.Ficticio=1}
4:   if (r.Tiempo(r.Largo-1)+r.Servicio(r.Largo-1))  $\geq$  new.Tiempo then
5:      $NCosto \leftarrow CostoRuta(r, new, r.Largo)$ 
6:   else
7:      $NCosto \leftarrow CostoRuta(r, new, r.Largo + 1)$ 
8:   end if
9: end if
10: return  $NCosto$ 
```

Función *CostoRuta*

El algoritmo *CostoRuta* devuelve el costo de una ruta (Int) que es alterada con un nuevo llamado. La función devuelve el valor 0 en caso de error (largo de ruta menor que 3, por ejemplo).

ARREGLAR PARA QUE FUNCIONES PARA TODO N

Función *CambioRuta*

El algoritmo *CambioRuta* devuelve una ruta con el nuevo llamado en la posición indicada.

Algorithm 2 *CostoRuta***Require:** Ruta r , Llamado new , Posicion p

```

1:  $CRuta \leftarrow 0$ 
2:  $CDist \leftarrow 0$ 
3:  $CEspera \leftarrow 0$ 
4:  $CTard \leftarrow 0$ 
5: if ( $p=r.Largo$ ) then
6:   while  $i \leq (r.Largo-2)$  do
7:      $CDist \leftarrow CDist + DIST(r.Posicion(i), r.Posicion(i+1))$ 
8:   end while
9:    $CDist \leftarrow CDist + DIST(r.Posicion(r.Largo-1), new.Posicion)$ 
10:  while  $i \leq (r.Largo-1)$  do
11:     $CTard \leftarrow CTard + |r.EspPrev(i) + r.Tiempo(i) - r.Ventana(i)|^+$ 
12:     $CEspera \leftarrow CEspera + r.EspPrev(i) + r.Tiempo(i) - |r.EspPrev(i) + r.Tiempo(i) - r.Ventana(i)|^+$ 
13:  end while
14:   $CTard \leftarrow CTard + |new.Tiempo - new.Ventana|^+$ 
15:   $CEspera \leftarrow CEspera + new.Tiempo - |new.Tiempo - new.Ventana|^+$ 
16:   $CRuta \leftarrow \alpha \times CTard + \beta \times CEspera + \gamma \times CDist$ 
17: else
18:  if ( $p=r.Largo+1$ ) then
19:     $CRuta \leftarrow r.Costo + \gamma \times DIST(r.Posicion(r.Largo), new.Posicion) + (\alpha - \beta) \times |new.Tiempo - new.Ventana|^+ + \beta \times new.Tiempo$ 
20:  end if
21: end if
22: return  $CRuta$ 

```

Algorithm 3 *CambioRuta***Require:** Ruta r , Llamado new , Posicion p

```

1: if ( $p=r.Largo$ ) then
2:    $r.Costo \leftarrow CostoRuta(r, new, p)$ 
3:    $r.Ficticio \leftarrow 0$ 
4:    $r.Posicion(p) \leftarrow new.Posicion$ 
5:    $r.Ventana(p) \leftarrow new.Ventana$ 
6: else
7:  if ( $p=r.Largo+1$ ) then
8:     $r.Largo \leftarrow (r.Largo + 1)$ 
9:     $r.Costo \leftarrow CostoRuta(r, new, p)$ 
10:    $r.Ficticio \leftarrow 0$ 
11:    $r.Posicion(p) \leftarrow new.Posicion$ 
12:    $r.Ventana(p) \leftarrow new.Ventana$ 
13:  end if
14: end if
15: return  $CRuta$ 

```

ID	Zonas	Mayo	Junio	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre	Enero	Febrero
1	La Pintana	11	11	15	11	16	13	11	12	8	5
2	Puente Alto - Centro	50	58	45	47	52	51	43	39	55	49
3	Puente Alto - Este	12	9	10	11	15	6	5	7	9	16
4	Puente Alto - Norte	39	16	39	24	17	20	18	18	17	18
5	La Florida - Sur	24	10	15	9	10	13	7	14	7	14
6	La Florida - SurPoniente	14	14	14	6	7	3	9	4	9	10
7	La Florida - Poniente	23	15	25	26	17	18	12	15	11	17
8	La Florida - Vespucio	114	76	95	84	72	132	86	72	106	94
9	San Joaquín - Sur	73	80	90	109	76	111	100	105	93	31
10	San Joaquín - Norte	83	70	76	73	47	96	56	49	58	49
11	San Miguel - Norte	60	49	63	68	53	56	40	60	63	43
12	San Miguel - Sur	42	38	54	29	27	46	36	35	43	41
13	P.A. Cerda	14	15	10	14	8	16	8	11	15	5
14	La Granja	10	16	14	12	7	8	8	6	3	10
15	San Ramon	10	12	11	16	11	14	4	4	1	7
16	La Cisterna	28	30	24	23	27	22	20	19	22	29
17	El Bosque	45	33	43	40	30	27	15	27	18	24
18	San Bernardo - Norte	54	35	69	45	33	46	50	27	40	39
19	Lo Espejo	20	13	19	16	12	26	11	9	16	12
20	Cerrillos - Norte	59	44	66	47	39	49	42	42	24	31
21	Cerrillos - Centro	30	23	38	24	19	38	36	22	31	35
22	Maipu-Pajaritos	21	24	26	20	18	28	31	20	28	21
23	Cerrillos - Sur	57	55	61	43	34	43	57	33	52	37
24	Maipu-Industrial	34	33	47	35	36	33	34	29	34	25
25	Maipu-Centro	51	71	56	60	61	63	51	25	38	41
26	Maipu-Rinconada	17	6	11	3	9	11	16	5	13	8
27	Padre Hurtado	22	24	28	17	29	32	17	23	14	18
28	San Bernardo - Sur	14	13	18	17	11	22	22	11	16	17
29	San Bernardo - Centro	63	68	86	72	58	64	92	62	56	75
30	La Reina	88	79	90	73	62	76	66	53	55	56
31	Las Condes - Alto	118	117	119	136	92	130	108	107	85	56
32	Las Condes - Apumanque	78	96	79	103	66	86	67	56	72	71
33	Las Condes - El Golf	239	189	266	172	182	196	155	162	200	144
34	Las Condes - Escuela	214	208	245	233	178	214	225	182	201	163
35	Las Condes - Isidora	263	229	356	275	278	286	299	248	204	225
36	Las Condes - Kennedy	72	55	71	70	53	63	58	39	40	44
37	Las Condes - Poniente	42	32	40	39	31	49	33	27	45	25
38	Las Condes - Sur	30	34	44	28	27	39	22	24	18	30
39	Lo Barnechea	59	44	48	63	35	59	42	41	33	41
40	Macul - Norte	46	46	45	55	45	63	50	51	47	51
41	Macul - Sur	92	65	84	83	79	96	60	70	47	65
42	Ñuñoa - NorEste	67	64	65	49	38	55	39	45	39	36
43	Ñuñoa - NorOeste	110	90	88	93	76	94	87	77	87	78
44	Ñuñoa - SurEste	31	40	32	39	25	28	29	22	23	22
45	Ñuñoa - SurOeste	164	156	186	165	133	178	123	120	116	127
46	Peñalolen	29	36	33	35	32	39	36	23	24	22
47	Vitacura - Centro	78	78	78	74	66	87	82	85	84	68
48	Vitacura - Este	31	25	37	26	33	27	32	28	22	13
49	Vitacura - Oeste	133	131	148	125	139	157	129	92	137	120

Tabla A.1: Llamados totales anuales por mes.

ID		7:00 - 8:00	8:00 - 9:00	9:00 - 10:00	10:00 - 11:00	11:00 - 12:00	12:00 - 13:00	13:00 - 14:00
	Totales	97	1773	5188	5064	4338	3680	2123
	Porcentajes %	0,29	5,31	15,55	15,18	13	11,03	6,36
1	La Pintana	0	6	13	28	34	15	9
2	Puente Alto - Centro	0	25	98	98	90	81	41
3	Puente Alto - Este	0	11	16	12	15	4	5
4	Puente Alto - Norte	0	6	24	44	51	41	24
5	La Florida - Sur	0	7	28	29	18	19	9
6	La Florida - SurPoniente	0	4	22	11	16	10	7
7	La Florida - Poniente	0	17	33	33	28	22	11
8	La Florida - Vespucio	1	54	177	135	139	133	88
9	San Joaquín - Sur	0	36	145	155	125	114	74
10	San Joaquín - Norte	2	43	137	131	107	103	53
11	San Miguel - Norte	0	52	111	119	85	56	44
12	San Miguel - Sur	0	23	77	72	61	54	34
13	P.A. Cerda	0	8	18	25	20	18	3
14	La Granja	1	7	19	19	9	15	5
15	San Ramon	0	13	20	16	16	14	6
16	La Cisterna	0	12	45	50	39	29	19
17	El Bosque	0	24	71	64	43	21	20
18	San Bernardo - Norte	0	37	86	76	70	49	29
19	Lo Espejo	0	16	22	25	26	17	11
20	Cerrillos - Norte	0	41	87	82	93	44	37
21	Cerrillos - Centro	0	15	50	64	40	29	30
22	Maipu-Pajaritos	0	20	35	28	51	35	14
23	Cerrillos - Sur	0	35	74	84	80	67	37
24	Maipu-Industrial	0	35	82	54	41	43	23
25	Maipu-Centro	0	30	98	93	67	63	49
26	Maipu-Rinconada	0	14	16	19	13	11	7
27	Padre Hurtado	0	22	50	40	35	36	18
28	San Bernardo - Sur	0	13	35	21	30	19	14
29	San Bernardo - Centro	0	34	126	128	109	83	51
30	La Reina	0	57	123	131	109	100	67
31	Las Condes - Alto	0	84	199	219	146	130	78
32	Las Condes - Apumanque	0	39	122	165	99	108	73
33	Las Condes - El Golf	0	105	384	339	311	267	144
34	Las Condes - Escuela	0	146	404	385	324	265	130
35	Las Condes - Isidora	3	112	448	475	428	391	199
36	Las Condes - Kennedy	2	34	101	89	93	75	50
37	Las Condes - Poniente	0	19	64	83	51	34	19
38	Las Condes - Sur	0	13	43	53	41	45	10
39	Lo Barnechea	0	29	82	81	86	74	45
40	Macul - Norte	15	37	95	86	76	58	37
41	Macul - Sur	68	70	155	115	94	75	48
42	Ñuñoa - NorEste	5	31	98	96	65	55	49
43	Ñuñoa - NorOeste	0	47	170	185	140	121	58
44	Ñuñoa - SurEste	0	14	46	46	51	49	17
45	Ñuñoa - SurOeste	0	108	314	273	236	189	104
46	Peñalolen	0	25	50	58	56	38	33
47	Vitacura - Centro	0	49	173	150	120	121	58
48	Vitacura - Este	0	20	51	51	46	50	20
49	Vitacura - Oeste	0	74	251	229	215	190	112

Tabla A.2: Llamados totales anuales por hora.

	Lunes	Martes	Miércoles	Jueves	Viernes	Promedio Diario	σ	$\sigma en\%$
Vitacura - Oeste	335	332	334	310	272	316,6	26,98	8,52
Vitacura - Centro	235	184	194	197	165	195	25,62	13,14
Vitacura - Este	78	77	65	76	57	70,6	9,24	13,09
Las Condes - Isidora	688	654	662	642	544	638	55,19	8,65
Las Condes - El Golf	486	470	475	466	404	460,2	32,3	7,02
Las Condes - Escuela	555	495	544	496	426	503,2	51,06	10,15
Las Condes - Apumanque	207	179	170	209	161	185,2	21,78	11,76
Las Condes - Kennedy	159	138	129	137	117	136	15,36	11,29
Las Condes - Alto	302	258	253	270	212	259	32,47	12,54
Las Condes - Poniente	117	97	69	90	68	88,2	20,54	23,29
Las Condes - Sur	79	73	74	48	60	66,8	12,64	18,92
Ñuñoa - NorOeste	238	209	209	205	208	213,8	13,63	6,38
Ñuñoa - NorEste	132	96	119	147	103	119,4	20,84	17,45
Ñuñoa - SurOeste	389	347	391	372	288	357,4	42,62	11,93
Ñuñoa - SurEste	75	81	76	67	63	72,4	7,27	10,04
La Reina	196	180	176	178	140	174	20,59	11,83
Maipu - Rinconada	36	21	26	16	13	22,4	9,07	40,49
Maipu - Centro	115	128	144	120	94	120,2	18,31	15,23
Maipu - Industrial	86	80	86	87	67	81,2	8,41	10,36
Maipu - Pajaritos	80	48	57	50	43	55,6	14,54	26,15
Cerrillos - Norte	126	100	106	118	82	106,4	16,99	15,97
Cerrillos - Centro	75	51	77	66	73	68,4	10,57	15,45
Cerrillos - Sur	126	128	102	112	84	110,4	18,19	16,48
Lo Espejo	33	31	36	52	21	34,6	11,24	32,49
P.A. Cerda	36	21	27	34	22	28	6,82	24,36
San Miguel - Sur	102	89	94	106	84	95	9,06	9,54
San Miguel - Norte	136	126	131	137	133	132,6	4,39	3,31
San Joaquín - Sur	241	215	189	181	197	204,6	23,93	11,7
San Joaquín - Norte	193	162	183	157	136	166,2	22,44	13,5
Macul - Norte	182	134	112	105	90	124,6	35,79	28,72
Macul - Sur	324	153	125	143	112	171,4	86,77	50,62
Peñalolen	96	82	72	68	73	78,2	11,19	14,31
La Cisterna	70	53	58	64	52	59,4	7,6	12,79
San Ramon	32	23	21	17	26	23,8	5,63	23,66
La Granja	26	27	20	23	12	21,6	6,02	27,87
La Florida - Vespucio	261	230	209	222	174	219,2	31,7	14,46
La Florida - Poniente	43	41	46	49	42	44,2	3,27	7,4
La Florida - Sur	30	34	32	35	19	30	6,44	21,47
La Florida - SurPoniente	22	22	27	25	15	22,2	4,55	20,5
El Bosque	81	76	73	86	68	76,8	6,98	9,09
La Pintana	40	31	35	32	19	31,4	7,77	24,75
Padre Hurtado	84	50	57	60	41	58,4	16,07	27,52
Puente Alto - Centro	140	113	143	108	107	122,2	17,8	14,57
Puente Alto - Norte	78	58	65	42	29	54,4	19,24	35,37
Puente Alto - Este	34	18	25	24	19	24	6,36	26,5
San Bernardo - Centro	182	163	154	172	144	163	14,87	9,12
San Bernardo - Norte	124	111	89	112	97	106,6	13,72	12,87
San Bernardo - Sur	59	37	42	35	31	40,8	10,92	26,76
Lo Barnechea	109	119	103	118	142	118,2	14,86	12,57

Tabla A.3: Llamados por día de la semana por zona.