



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

UTILIZACIÓN DE MODELOS EMPÍRICOS Y FILTROS DE PARTÍCULAS SENSIBLES
AL RIESGO PARA DETECCIÓN TEMPRANA DE ANOMALÍAS

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELECTRICISTA

GABRIEL FELIPE OSSA GÓMEZ

PROFESOR GUÍA:
MARCOS ORCHARD CONCHA

MIEMBROS DE LA COMISIÓN:
HÉCTOR AGUSTO ALEGRIA
JORGE SILVA SÁNCHEZ

SANTIAGO DE CHILE
JULIO 2013

RESUMEN DE LA MEMORIA
PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL ELECTRICISTA
POR: GABRIEL FELIPE OSSA GÓMEZ
FECHA: 9 DE JULIO DE 2013
PROF. GUÍA: MARCOS ORCHARD CONCHA

UTILIZACIÓN DE MODELOS EMPÍRICOS Y FILTROS DE PARTÍCULAS SENSIBLES AL RIESGO PARA DETECCIÓN TEMPRANA DE ANOMALÍAS

El presente trabajo de memoria postula como objetivo el estudio del desempeño de filtros de partículas en el área del pronóstico y diagnóstico de fallas, modelando sistemas dinámicos no-lineales por medio de formas paramétricas. También se desarrollan diferentes estrategias de filtrado, las cuales evalúan la validez del estado estimado an base a la información pasada de la operación. Todas las estrategias se evalúan analizando el efecto de la variabilidad de los parámetros del modelo sobre el rendimiento en estimación, predicción y detección de anomalías

Los filtros de partículas corresponden a técnicas basadas en representaciones empíricas de la distribución de probabilidad del estado en modelos Bayesianos. En el contexto de un modelo dinámico, tales técnicas sirven para estimar los estados y parámetros variantes en el tiempo de dicho modelo. El uso de curvas de ajuste paramétricas permite expandir el uso previamente mencionado a modelos que además de incluir las componentes mencionadas poseen parámetros fijos en su definición.

El rendimiento de los métodos estudiados fue evaluado mediante el uso de datos empíricos provenientes de pruebas experimentales. En particular, los datos usados corresponden a una estimación directa del crecimiento de una fractura axial en una placa del engranaje planetario de un helicóptero.

Para abordar el problema, se determinó usar la función arcotangente hiperbólica como curva de ajuste, con el fin de modelar la propagación de la fractura del componente mecánico en el tiempo. Además, se establecieron mecanismos novedosos basados en la generación de proyecciones de trayectorias en el pasado de la curva de ajuste a partir del vector de estados del modelo para un instante determinado, mediante la cual se evalúa el verosimilitud histórica de la estimación proporcionada por el algoritmo haciendo uso de la distribución chi-cuadrado y su distribución acumulada complementaria. De esta forma, se analizó el desempeño de cinco filtros: uno que sólo incluye la curva de ajuste en su operación, y otros cuatro que generan proyecciones de trayectorias en el pasado, para distintas ventanas de tiempo histórico y distribuciones de evaluación.

El análisis de los filtros entregó resultados regulares. El filtro sin proyección pasada tuvo un mejor desempeño a la hora de filtrar, y la predicción ofrecida por el mismo se mantuvo estable. Por otra parte, los filtros basados en el uso de proyecciones de trayectorias en el pasado y que evaluaban con la distribución chi-cuadrado presentaron signos de degeneración muestral para variabilidades muy altas en los parámetros de la curva de ajuste, pero ofrecieron capacidades de detección importantes. Por último se observó un alto nivel de degeneración muestral en los filtros que hacían uso de las proyecciones en el pasado pero que evaluaban con la distribución acumulada complementaria de la distribución chi-cuadrado, bajo todas las condiciones experimentales estudiados.

Se concluye a partir del trabajo realizado que la variabilidad de los ruidos involucrados al usar formas paramétricas en el modelo de sistema influye fuertemente en los desempeños de los filtros, y que, en términos de desempeño, el filtro de partículas sólo con curva de ajuste tiene mejor desempeño que aquellos que hacen uso de proyecciones de trayectorias en el pasado.

Agradecimientos

Quiero comenzar agradeciendo a mi familia que me ha acompañado y soportado durante todos estos años en esta larga carrera. Me han apoyado desde mi nacimiento y lo siguen haciendo, y a pesar de todas las dificultades, siempre han estado ahí, en casi todas las cosas que alguna vez he hecho. Los quiero mucho.

También me encantaría agradecer a mis amigos de colegio: Guti, Allan, Andrés, Fango, Aco, Mella, Memo, Guarda, Dani, Meli, Vilma, Mónica, Pipe. Mención aparte y especial merecen aquellos que me acompañaron sobre el escenario, mis compañeros Melosseanos: Seba, Cabi, Mendo y nuestro sonidista oficial Campos. Sin la compañía de ustedes mi vida estaría muy vacía y no hubiera aguantado esos momentos difíciles que alguna vez tuve.

Agradezco a mis amigos que conocí en la universidad: los del Bachillerato: Porras, Chigó, Pizo, Ruth, Contente, Compa, Dani, Nati, Loreto; los del laboratorio: Felipe, Alexis, Samir, Fuentealba, Tomás, Hugo, Daniel, Paul; aquellos del departamento: Dharma, Kirby, García, Cono, Valdivia, Andrejo, Hortencia, Con, Feña, Jano, Shaggy, Pao, Lumbi, Pablo, Chino, Daniel. Sin ustedes estos años hubiesen sido mucho más difíciles de lo que fueron. Muchas gracias por todos los buenos momentos, pizzatones y juntas eléctricas.

Me gustaría dedicar agradecimientos a mis profesores que me enseñaron, educaron y aportaron en mi vida durante el transcurso de mi carrera. En especial, quisiera destacar a Marcos Orchard, para quien trabajé durante años y que ha sido mi mentor desde que entré al departamento de ingeniería eléctrica —además de ser mi profesor guía durante este trabajo—, y que me apoyó con enseñanzas y conversaciones, tanto de forma académica como para la vida en general.

Por último, agradezco con un gran abrazo a Luz Godoy, ya que sin ella nada hubiera sido posible durante este último período. Te amo mucho.

Tabla de contenido

| | |
|--|-----------|
| Introducción | 1 |
| 1. Revisión bibliográfica | 3 |
| 1.1. Modelo dinámico y perspectiva del análisis | 3 |
| 1.2. El problema de filtrado para estados | 3 |
| 1.2.1. Filtro de partículas | 4 |
| 1.3. El problema de filtrado para estados y parámetros | 8 |
| 1.3.1. Evolución artificial de parámetros | 8 |
| 1.4. Predicción de eventos catastróficos | 9 |
| 1.4.1. Detección de fallas | 9 |
| 1.4.2. Preprocesamiento de los datos | 9 |
| 1.4.3. Extracción de la característica | 10 |
| 1.4.4. Fusión de los datos provenientes de los sensores y de las características | 10 |
| 1.4.5. Toma de decisiones | 10 |
| 1.5. Filtro de partículas aplicados al pronóstico de sistemas | 11 |
| 1.5.1. Generación de predicciones a plazos definidos | 11 |
| 1.5.2. Actualización de los pesos | 11 |
| 1.5.3. Proyección en el tiempo del valor esperado de los estados | 12 |
| 2. Problemática y metodología usada | 13 |
| 2.1. Descripción del problema en estudio | 13 |
| 2.1.1. Antecedentes técnicos | 13 |
| 2.1.2. Caracterización de los datos usados | 13 |
| 2.2. Descripción general de la modelación | 15 |
| 2.2.1. Curva a utilizar y modelo resultante | 16 |
| 2.2.2. Filtro de partículas | 17 |
| 2.2.3. Marco de pruebas | 22 |
| 2.3. Implementación del modelo y los filtros | 23 |
| 2.3.1. Herramientas utilizadas en el desarrollo del trabajo | 23 |
| 2.3.2. Consideraciones respecto a la programación | 23 |
| 2.3.3. Descripción del código implementado | 24 |
| 3. Discusión de resultados | 28 |
| 3.1. Análisis preliminar de las estrategias de filtrado | 29 |
| 3.1.1. Filtro de partículas “clásico” | 29 |

| | | |
|-----------|---|------------|
| 3.1.2. | Filtro con proyección móvil de la trayectoria en el pasado y verosimilitud basada en la densidad chi-cuadrado | 33 |
| 3.1.3. | Filtro con proyección acumulada de la trayectoria en el pasado y verosimilitud basada en la densidad chi-cuadrado | 36 |
| 3.1.4. | Filtro con proyección de trayectoria en el pasado y verosimilitud basada en la CCDF de la distribución chi-cuadrado | 40 |
| 3.2. | Análisis de los efectos de las varianzas de los ruidos y resultados numéricos | 44 |
| 3.2.1. | Desempeño de las estrategias de filtrado de partículas implementadas en labores de estimación | 44 |
| 3.2.2. | Desempeño de las estrategias de filtrado de partículas implementadas en labores de predicción | 45 |
| 3.2.3. | Detección de fallas en base a parámetros | 46 |
| 3.3. | Efecto de grandes variabilidades sobre el vector de parámetros | 61 |
| 3.4. | Comparación de las estrategias de filtrado y comentarios finales | 64 |
| | Conclusión | 65 |
| | Bibliografía | 68 |
| A. | Tablas de datos | I |
| A.1. | Efecto de la variación de las varianzas sobre la estimación de los filtros | I |
| A.2. | Efecto de la variación de las varianzas sobre el pronóstico de los filtros | IV |
| B. | Código de los programas | VII |
| B.1. | MAIN_atanh.m | VII |
| B.2. | tests.m | X |
| B.3. | Update_particles_atanh.m | XII |
| B.4. | gauss_sample.m | XIII |
| B.5. | gauss_likelihood.m | XIII |
| B.6. | chi2test_atanhV1.m | XIII |
| B.7. | chi2test_atanhV2.m | XIV |
| B.8. | chi2test_atanhV3.m | XV |
| B.9. | chi2test_atanhV4.m | XV |
| B.10. | resampling.m | XVI |
| B.11. | residualresample.m | XVI |
| B.12. | output_mean_atanh.m | XVIII |
| B.13. | D_atanh.m | XVIII |
| B.14. | funcPred.m | XVIII |
| B.15. | estErr.m | XVIII |
| B.16. | predErr.m | XIX |
| B.17. | confRange.m | XX |
| B.18. | paramGen.m | XX |
| B.19. | currentStatePrognosis.m | XX |

Índice de tablas

| | |
|---|-----|
| 2.1. Estructuras de datos utilizadas en los códigos fuente | 24 |
| 2.2. Estructura de archivos utilizadas | 25 |
| 2.3. Estructura de archivos utilizadas (continuación) | 26 |
| 3.1. Variables usadas en los filtros estudiados | 29 |
| 3.2. Variables modificadas usadas en las estrategias de filtrado | 61 |
| A.1. Influencia de la varianza del ruido de proceso en el error cuadrático medio para la estrategia de filtrado 1. | I |
| A.2. Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 1. | I |
| A.3. Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 1. | II |
| A.4. Influencia de la varianza del ruido de observación en el error cuadrático medio para la estrategia de filtrado 1. | II |
| A.5. Influencia de la varianza del ruido de proceso en el error cuadrático medio para la estrategia de filtrado 2. | II |
| A.6. Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 2. | II |
| A.7. Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 2. | II |
| A.8. Influencia de la varianza del ruido de observación en el error cuadrático medio para la estrategia de filtrado 2. | II |
| A.9. Influencia de la varianza del ruido de proceso en el error cuadrático medio para la estrategia de filtrado 3. | III |
| A.10. Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 3. | III |
| A.11. Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 3. | III |
| A.12. Influencia de la varianza del ruido de observación en el error cuadrático medio para la estrategia de filtrado 3. | III |
| A.13. Influencia de la varianza del ruido de proceso en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1. | IV |
| A.14. Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1. | IV |

| | |
|---|----|
| A.15. Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1. | IV |
| A.16. Influencia de la varianza del ruido de observación en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1. | V |
| A.17. Influencia de la varianza del ruido de proceso en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2. | V |
| A.18. Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2. | V |
| A.19. Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2. | V |
| A.20. Influencia de la varianza del ruido de observación en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2. | V |
| A.21. Influencia de la varianza del ruido de proceso en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3. | V |
| A.22. Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3. | VI |
| A.23. Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3. | VI |
| A.24. Influencia de la varianza del ruido de observación en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3. | VI |

Índice de figuras

| | |
|--|----|
| 2.1. Componentes mecánicos de la transmisión del helicóptero [7] | 14 |
| 2.2. Estimación ruidosa de la longitud de la grieta en función de los instantes de tiempo GAG [7] | 14 |
| 2.3. Comparación entre el conjunto de datos usado y la curva de ajuste escogida (arcotangente hiperbólica) | 16 |
| 2.4. Efecto de la variación de parámetros en la forma de la arcotangente hiperbólica | 17 |
| 2.5. Perfil de la verosimilitud basada en la distribución chi-cuadrado acumulada complementaria | 22 |
| 2.6. Funcionamiento básico del programa | 27 |
| 3.1. Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 1. | 30 |
| 3.2. Acercamiento a la salida estimada de la estrategia de filtrado 1 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG]) | 31 |
| 3.3. Predicciones generadas por cada una de las partículas para la estrategia de filtrado 1 | 31 |
| 3.4. Evolución de las componentes del vector de parámetros usado en el modelo . | 32 |
| 3.5. Evolución de los pesos de las partículas usadas en la estrategia de filtrado 1 . | 32 |
| 3.6. Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 2. | 33 |
| 3.7. Acercamiento a la salida estimada de la estrategia de filtrado 2 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG]) | 34 |
| 3.8. Predicciones generadas por cada una de las partículas para la estrategia de filtrado 2 | 34 |
| 3.9. Evolución de las componentes del vector de parámetros usado en el modelo . | 35 |
| 3.10. Evolución de los pesos de las partículas usadas en la estrategia de filtrado 2 . | 35 |
| 3.11. Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la de filtrado 3 | 37 |
| 3.12. Acercamiento a la salida estimada de la estrategia de filtrado 3 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG]) | 38 |
| 3.13. Predicciones generadas por cada una de las partículas para la estrategia de filtrado 3 | 38 |
| 3.14. Evolución de las componentes del vector de parámetros usado en el modelo . | 39 |
| 3.15. Evolución de los pesos de las partículas usadas en la estrategia de filtrado 3 . | 39 |
| 3.16. Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 4 | 41 |

| | |
|--|----|
| 3.17. Acercamiento a la salida estimada de la estrategia de filtrado 4 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG]) | 42 |
| 3.18. Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 5 | 42 |
| 3.19. Acercamiento a la salida estimada de la estrategia de filtrado 5 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG]) | 43 |
| 3.20. Evolución de los pesos de las partículas usadas por el filtro. Izquierda: Estrategia de filtrado 4. Derecha: Estrategia de filtrado 5 | 43 |
| 3.21. Evolución del parámetro a para diversos ruidos de proceso usando la estrategia de filtrado 1 | 47 |
| 3.22. Evolución del parámetro a para diversos ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 1 | 48 |
| 3.23. Evolución del parámetro a para diversos ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 1 | 48 |
| 3.24. Evolución del parámetro a para diversos ruidos de observación usando la estrategia de filtrado 1 | 49 |
| 3.25. Evolución del parámetro a para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 2 | 49 |
| 3.26. Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 2 | 50 |
| 3.27. Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 2 | 51 |
| 3.28. Evolución del parámetro a para diversas varianzas de ruidos de observación usando la estrategia de filtrado 2 | 52 |
| 3.29. Evolución del parámetro a para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 3 | 52 |
| 3.30. Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 3 | 53 |
| 3.31. Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 3 | 53 |
| 3.32. Evolución del parámetro a para diversas varianzas de ruidos de observación usando la estrategia de filtrado 3 | 54 |
| 3.33. Evolución del parámetro b para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 1 | 54 |
| 3.34. Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 1 | 55 |
| 3.35. Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 1 | 55 |
| 3.36. Evolución del parámetro b para diversas varianzas de ruidos de observación usando la estrategia de filtrado 1 | 56 |
| 3.37. Evolución del parámetro b para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 2 | 57 |
| 3.38. Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 2 | 58 |
| 3.39. Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 2 | 58 |

| | |
|--|----|
| 3.40. Evolución del parámetro b para diversas varianzas de ruidos de observación usando la estrategia de filtrado 2 | 59 |
| 3.41. Evolución del parámetro b para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 3 | 59 |
| 3.42. Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 3 | 60 |
| 3.43. Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 3 | 60 |
| 3.44. Evolución del parámetro b para diversas varianzas de ruidos de observación usando la estrategia de filtrado 3 | 61 |
| 3.45. Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad | 62 |
| 3.46. Predicción de la grieta, obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad | 63 |
| 3.47. Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad | 63 |
| 3.48. Predicción de la grieta, obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad | 64 |
| 3.49. Evolución de los pesos de las partículas usadas por el filtro para alta variabilidad en el vector de parámetros y baja variabilidad en la variable de estado. Izquierda: Estrategia de filtrado 2. Derecha: Estrategia de filtrado 3 | 64 |

Introducción

El presente trabajo se enmarca dentro de la problemática del pronóstico y diagnóstico de fallas en sistemas dinámicos. En particular, se estudiará el uso de curvas paramétricas, en conjunto con *filtros de partículas* (también llamados *métodos secuenciales de Monte Carlo*).

Los datos usados en los experimentos realizados para analizar el desempeño de los métodos se obtienen a partir del estudio del crecimiento de una fractura axial en una placa de engranaje planetario de un helicóptero UH-60A. Este problema es de importancia por sí mismo, dado que la aparición de este tipo de fracturas no es exclusiva al UH-60A, y por tanto afecta la seguridad que entregan diversos modelos de helicópteros. Este problema se instala dentro de la lógica del pronóstico de sistemas, puesto que guarda relación con la extensión de la vida útil de un equipo, con la reducción de tiempos de baja, con la determinación de tiempos óptimos de mantención así como la planificación de los costos de esta.

El filtro de partículas es un algoritmo de estimación Bayesiano, el cual se encarga de estimar el estado de un sistema por medio de la generación de una población de muestras, con pesos asociados los cuales representan la probabilidad de que tales muestras caractericen la distribución de probabilidad del estado. De esta forma, durante la estimación, los pesos son actualizados convenientemente —para cada partícula— en base a un modelo de estado y uno de medición.

El uso de modelos es común para el estudio de sistemas dinámicos. En estos, se describe el comportamiento de un sistema o una planta en base a términos variantes en el tiempo. Sin embargo, muchos de estos modelos hacen uso de componentes cuyo valor si bien es ajustable, no es temporalmente dependiente.

Si bien los filtros de partículas son utilizados en el contexto de un problema de diagnóstico, su uso en este campo se concentra en aplicaciones donde los parámetros asociados a los modelos involucrados dependen del tiempo. De esta forma, es de interés extender el uso de este algoritmo a modelos donde existen parámetros invariante en el tiempo, especialmente si tales parámetros representan la operación normal de una planta o sistema.

También dentro del mismo contexto, el proceso de actualización por medio del cual las partículas modifican su peso es de principal relevancia en este trabajo. La idea es el desarrollo de formas de mejorar la actualización mediante el uso de información pasada, lo cual está fuera del alcance del algoritmo original.

Dicho esto, el problema a abordar se puede formular de la siguiente manera: analizar el

desempeño de métodos basados en el filtro de partículas en conjunto con curvas paramétricas, dado el problema de pronóstico descrito, tanto en las dimensiones de estimación, predicción y detección de características. El uso de las curvas paramétricas se introduce como una forma de modelar la evolución de la planta en base a parámetros constantes. En forma particular, se asume que el ajuste de la variabilidad de las cantidades involucradas en la modelación de algoritmos usados es suficiente para estimar el valor real de la fractura, predecir su evolución y detectar anomalías en su ritmo de crecimiento.

Este documento se separa en 5 partes, además de un apartado de referencias bibliográficas y dos anexos. La primera parte consta del presente apartado. El resto se estructura de la siguiente forma: en el Capítulo 1 se hace una síntesis de los aspectos teóricos que soportan el trabajo. En el Capítulo 2 se describe el problema a abordar, tanto aspectos experimentales como teóricos, así como el implementación de los algoritmos. Los resultados y las discusiones acerca de los mismos se presentan en el Capítulo 3. Finalmente, en el último apartado se concluye a partir de los resultados obtenidos y se sugieren algunas líneas de investigación.

Capítulo 1

Revisión bibliográfica

1.1. Modelo dinámico y perspectiva del análisis

Sea modelo dinámico de Markov para una colección de datos observados de forma secuencial y_k , en el cual el vector de estados en el tiempo k es x_k y el vector de parámetros fijos es θ . El modelo se especifica en cada instante k por la ecuación de observación que define la densidad de observación

$$p(y_k|x_k, \theta) \tag{1.1}$$

y la ecuación de evolución Markoviana, o ecuación de estado, definida por la densidad de transición

$$p(x_k|x_{k-1}, \theta) \tag{1.2}$$

Cada y_k es condicionalmente independiente de los estados pasados y observaciones dado el estado actual x_k y el parámetro θ , y x_k es condicionalmente independiente de los estados pasados y las observaciones dado x_{k-1} y θ . Esto cubre una clase muy amplia de modelos interesantes en el sentido práctico [5].

1.2. El problema de filtrado para estados

Considérese un modelo de parámetros fijos, o en donde θ se asume desconocido. Entonces el modelo del sistema en el espacio de estados y en tiempo discreto queda dado por las siguientes ecuaciones:

$$x_k = f_k(x_{k-1}, \omega_k), \quad k \in \mathbb{N} \tag{1.3}$$

$$y_k = h_k(x_k, \nu_k), \quad k \in \mathbb{N} + \{0\} \tag{1.4}$$

donde para todo k , $x_k \in \mathbb{R}^{n_x}$ es el estado del sistema, $y_k \in \mathbb{R}^{n_y}$ es la observación, $u_k \in \mathbb{R}^{n_x}$ es el ruido de proceso, $\nu_k \in \mathbb{R}^{n_x}$ es el ruido de observación, y n_x, n_y son las dimensiones de x_k e y_k , respectivamente. Se asume que $\{u_k\}$ y $\{\nu_k\}$ son secuencias de variables aleatorias independientes e idénticamente distribuidas (i.i.d), las cuales son también independientes del estado inicial x_0 , el cual tiene por a p_0 como función de densidad de probabilidad. Por último, sea $p(x_k|x_{k-1})$ la forma de denotar la densidad de transición y $p(y_k|x_{k-1})$ la función de verosimilitud.

El problema de filtrado consiste en estimar las densidades condicionales

$$b_k(x_k) \triangleq p(x_k|y_{0:k}) \quad , \quad (1.5)$$

donde $y_{0:k}$ denota las observaciones desde el instante 0 al instante k . La densidad condicional $b_k(x_k)$ puede ser derivada de forma recursiva por medio de la ecuación de Chapman-Kolmogorov y la regla de Bayes como se muestra a continuación [12]:

$$b_k(x_k) = \frac{p(y_k|x_k)p(x_k|y_{0:k-1})}{p(y_k|y_{0:k-1})} \quad (1.6)$$

$$b_k(x_k) = \frac{p(y_k|x_k) \int p(x_k|x_{k-1})b_{k-1}(x_{k-1}) dx_{k-1}}{\int p(y_k|x_k)p(x_k|y_{0:k-1}) dx_{k-1}} \quad , \quad (1.7)$$

donde el denominador $\int p(y_k|x_k)p(x_k|y_{0:k-1})dx_{k-1}$ es sólo una constante de normalización.

Parece posible que dado $b_0(x_0)$ sea posible calcular $b_k(x_k)$ para cualquier k mediante la resolución recursiva de (1.7). Sin embargo, las integrales en (1.7) son por lo general intratables. Aún más, la densidad condicional $b_k(x_k)$ puede ser cualquier densidad de probabilidad, incluso si la densidad anterior $b_{k-1}(x_{k-1})$ es una distribución Gaussiana. Dado esto, la simulación de Monte Carlo es una técnica en extremo útil, y en particular, el muestreo por importancia es crucial debido a la dificultad ligada a la obtención de muestras desde distribuciones arbitrarias de forma directa.

1.2.1. Filtro de partículas

El *filtro de partículas* es una clase de filtro que utiliza simulaciones de Monte Carlo y muestreo por importancia para estimar las densidades condicionales. Éste método aproxima la densidad condicional usando un número finito de partículas (muestras) e imita la evolución de dicha densidad condicional por medio de la propagación de las partículas. De forma más específica, el filtro de partículas aproxima $b_k(x_k)$ por una función de pesos probabilística

$$\hat{b}_k(x_k) = \sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)}) \quad , \quad (1.8)$$

donde δ denota la función delta de Kronecker¹, $\{x_k^{(i)}\}_{i=1}^N$ son puntos de apoyo aleatorios, y $\{w_k^{(i)}\}_{i=1}^N$ son los pesos (probabilidades) asociados a cada punto.

¹La función delta de Kronecker se define como $\delta(z) = \begin{cases} 1 & \text{si } z = 0 \\ 0 & \text{si } z \neq 0 \end{cases}$

El proceso que incluye obtención de muestras $\{x_k^{(i)}\}_{i=1}^N$ y el cálculo de sus pesos correspondientes se presenta a continuación. De (1.7), se observa que

$$b_k(x_k) \propto \int p(y_k|x_k)p(x_k|x_{k-1})b_{k-1}(x_{k-1}) dx_{k-1} \quad . \quad (1.9)$$

Luego, para calcular la integral del lado derecho se necesitan muestras de (x_{k-1}, x_k) obtenidas a partir de $p(y_k|x_k)p(x_k|x_{k-1})b_{k-1}(x_{k-1})$. En lugar de tomar un ejemplar directamente de esta última expresión, se opta por introducir la distribuciones $q_k(x_k|x_{k-1}, y_k)$ y $g_{k-1}(x_{k-1}|y_{0:k-1})$ —llamadas *función de densidad de importancia* y *función de densidad de remuestreo por importancia*, de la cual se extraen muestras i.i.d. $(x_{k-1}, x_k)^{(i)} \triangleq (x_{k-1}^{(i)}, x_k^{(i)})$ de la siguiente manera:

$$x_{k-1}^{(i)} \sim g_{k-1}(x_{k-1}|y_{0:k-1}), \quad i = 1, \dots, N \quad (1.10)$$

$$x_k^{(i)} \sim q_k(x_k|x_{k-1}^{(i)}, y_k), \quad i = 1, \dots, N \quad . \quad (1.11)$$

Dado que $\{(x_{k-1}, x_k)^{(i)}\}_{i=1}^N$ son muestras i.i.d. extraídas de $q_k(x_k|x_{k-1}, y_k)g_{k-1}(x_{k-1}|y_{0:k-1})$, la integral en (1.9) se puede aproximar como:

$$\begin{aligned} & \int \frac{p(y_k|x_k)p(x_k|x_{k-1})b_{k-1}(x_{k-1})}{q_k(x_k|x_{k-1}, y_k)g_{k-1}(x_{k-1}|y_{0:k-1})} (q_k(x_k|x_{k-1}^{(i)}, y_k)g_{k-1}(x_{k-1}|y_{0:k-1})) dx_{k-1} \\ & \approx \sum_{i=1}^N \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})b_{k-1}(x_{k-1}^{(i)})}{q_k(x_k^{(i)}|x_{k-1}^{(i)}, y_k)g_{k-1}(x_{k-1}^{(i)}|y_{0:k-1})} \delta(x_k - x_k^{(i)}) \end{aligned} \quad (1.12)$$

Sean entonces los pesos normalizados como sigue:

$$w_k^{(i)} \propto \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})b_{k-1}(x_{k-1}^{(i)})}{q_k(x_k^{(i)}|x_{k-1}^{(i)}, y_k)g_{k-1}(x_{k-1}^{(i)}|y_{0:k-1})} \quad (1.13)$$

Luego $b_k(x_{k-1}) \approx \sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)})$

1.2.1.1. Remuestreo

Los filtros de partículas presentan un problema que, de no ser tratado adecuadamente, atenta considerablemente contra el desempeño del algoritmo [7]. Este problema, llamado *degeneración*, radica en que luego de algunas iteraciones sólo una de las partículas concentrará el peso relevante, mientras que en todas las demás será despreciable [1]. La degeneración es un problema que no se puede evitar dado que, por las características del algoritmo, la varianza de los pesos de las partículas sólo puede crecer en el tiempo [3]. No manejar correctamente este inconveniente se traduce en un desperdicio de cálculo computacional en partículas cuyo aporte es despreciable en la estimación [1].

Un método básico para contrarrestar la degeneración de las muestras es usar un gran número de muestras [12], pero una forma más eficiente consiste en remuestrear de forma tal que las nuevas muestras tengan todas pesos razonablemente grandes. Un remuestreo óptimo

debería debería generar pesos iguales a todas las nuevas muestras; luego, la densidad de remuestreo por importancia óptima queda dada por

$$g_k^{\text{opt}}(x_k|y_{0:k}) = b_k(x_k) \quad . \quad (1.14)$$

Nótese que, por lo general, no se dispone de la distribución $b_k(x_k)$; pero, como se vio anteriormente, se posee una aproximación discreta de la misma, i.e., $\hat{b}_k(x_k) = \sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)})$. De esta manera, en la práctica, la densidad de remuestreo por importancia se determina de la siguiente manera:

$$g_k(x_k|y_{0:k}) = \hat{b}_k(x_k) \quad . \quad (1.15)$$

La desventaja principal asociada a esta densidad de remuestreo por importancia corresponde al “empobrecimiento de la muestra”², lo cual significa que que muestras de gran peso son propensas a tener muchas copias de sí mismas mientras que aquellas asociadas a pesos pequeñas probablemente no sobrevivan luego del remuestreo (con reemplazo) [12], y tiene como consecuencia la posible generación de un conjunto de partículas poco diverso [7]. El caso crítico se da cuando el ruido de proceso es pequeño, lo cual puede provocar que todas las partículas colapsen a un solo punto en pocas iteraciones [1].

En cuanto al criterio bajo el cual se define la degeneración del algoritmo, este se puede definir como un umbral N_{eff} , el que se basa en el incremento de la varianza de los pesos involucradas, como se muestra a continuación:

$$N_{\text{eff}} = \frac{N}{1 + \text{Var}_{b_k(\cdot)}(w_{0:k})} \quad , \quad (1.16)$$

donde N corresponde al número de partículas utilizadas.

Lamentablemente, (1.16) no se puede evaluar directamente [1]. Es por esto que en la práctica se utiliza un estimador para el valor deseado [8], i.e.

$$\hat{N}_{\text{eff}} = \frac{1}{\sum_{i=1}^N (w_k^{(i)})^2} \quad . \quad (1.17)$$

Para decidir si es necesario hacer un remuestreo se compara \hat{N}_{eff} con N_T , una constante escogida previamente. Así, cada vez que $\hat{N}_{\text{eff}} \leq N_T$ se procede a realizar el remuestreo. El valor de N_T puede variar en cada problema, pero basta con asignarle un valor $N_T = 0,885N$ o $N_T = 0,9N$ para comenzar a analizar el desempeño del filtro [7].

1.2.1.2. Función de densidad de importancia

La elección de la densidad de importancia $q_k(x_k|x_{k-1}^{(i)}, y_k)$ también es importante a la hora de abordar el problema de la degeneración. Dada un función de densidad de remuestreo por

²También conocido como “pérdida de diversidad”.

importancia óptima (1.14) y $x_{k-1}^{(i)}$, la densidad de importancia óptima —de acuerdo a [1]— es

$$q_k^{\text{opt}}(x_k|x_{k-1}^{(i)}, y_k) = p(x_k|x_{k-1}^{(i)}, y_k) \quad . \quad (1.18)$$

Sin embargo, $p(y_k|x_{k-1}^{(i)})$ es por lo general difícil de evaluar, y por lo mismo en la práctica se opta por usar densidades de importancia subóptimas. Una elección común es la densidad de transición, i.e.,

$$q_k(x_k|x_{k-1}^{(i)}, y_k) = p(x_k|x_{k-1}^{(i)}) \quad . \quad (1.19)$$

Esta elección es fácil de implementar, y también lleva a evaluación simple de los pesos. Reemplazando (1.19) en (1.13), y asumiendo que

$$\hat{b}_{k-1}(x_{k-1}^{(i)}) = b_{k-1}(x_{k-1}^{(i)}) \quad (1.20)$$

se tiene que los pesos normalizados quedan dados por

$$w_k^{(i)} \propto p(y_k|x_k^{(i)}) \quad , \quad (1.21)$$

la cual es la función de verosimilitud, y por consecuencia es fácil de evaluar.

1.2.1.3. Actualización recursiva de pesos

Nótese que si, en lugar de asumir (1.15) se asume que $g_k(x_k|y_{0:k})$ sigue una distribución uniforme, se obtiene:

$$g_k(x_k^{(i)}|y_{0:k}) = 1/K, \quad i = 1, \dots, N \quad , \quad (1.22)$$

donde K es una constante que tiene relación con el “tamaño” del espacio de estado. Sin pérdida de generalidad, se entonces que la expresión (1.13) se puede exponer de la siguiente forma:

$$w_k^{(i)} \propto \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})b_{k-1}(x_{k-1}^{(i)})}{q_k(x_k^{(i)}|x_{k-1}^{(i)}, y_k)} \quad (1.23)$$

Tómese en cuenta además que, por definición se tiene que

$$b_{k-1}(x_{k-1}^{(i)}) = w_{k-1}^{(i)}, \quad i = 1, \dots, N \quad (1.24)$$

con lo que, al imponer (1.20), (1.13) pasa a ser

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(y_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})}{q_k(x_k^{(i)}|x_{k-1}^{(i)}, y_k)} \quad (1.25)$$

La última expresión es de gran utilidad a la hora de implementar filtros pues provee una expresión recursiva capaz de encargarse de la actualización de los pesos. Cabe mencionar además que, para lograr la igualdad, basta con normalizar todos los pesos [7].

Por último, imponiendo (1.19) se tiene que

$$w_k^{(i)} \propto w_{k-1}^{(i)} p(y_k|x_k^{(i)}) \quad (1.26)$$

De esta forma, con las elecciones expuestas en (1.19) y (1.22), la iteración definida por el muestreo por importancia es equivalente a propagar las partículas a lo largo de la dinámica del sistema, y la evaluación de los pesos se reduce al cálculo de de la función de verosimilitud [12].

1.3. El problema de filtrado para estados y parámetros

En el modelo general con parámetros fijos θ , el marco de trabajo basado exclusivamente en muestras (ver Sección 1.2) se extiende de la siguiente forma: en el instante k , se tiene una colección de muestras combinadas

$$\{x_k^{(i)}, \theta_k^{(i)} : i = 1, \dots, N\} \quad (1.27)$$

y sus pesos asociados

$$\{w_k^{(i)} : i = 1, \dots, N\} \quad , \quad (1.28)$$

representando una aproximación de muestra de importancia a la densidad posterior

$$b_k(x_k, \theta) = p(x_k, \theta | y_{0:k}) \quad (1.29)$$

tanto para los parámetros como para el estado. Nótese que el sufijo k en las muestras de θ indican que estas provienen del instante k posterior, no que θ varía con el tiempo. Llegado el instante $k + 1$, se observa y_{k+1} , y en ese momento se genera una muestra a partir de $b_{k+1}(x_{k+1}, \theta)$. De acuerdo con la regla de Bayes, esto resulta en

$$b_{k+1}(x_{k+1}, \theta) \propto p(y_{k+1} | x_{0:k+1}, \theta) p(x_{k+1}, \theta | y_{0:k}) \quad (1.30)$$

$$\propto p(y_{k+1} | x_{0:k+1}, \theta) p(x_{k+1}, \theta | y_{0:k}) p(\theta | y_{0:k}) \quad (1.31)$$

donde la forma escogida para representar (1.30), hace explícita la noción de que la función de densidad $p(\theta | y_{0:k})$ es parte importante del proceso de actualización [5].

Si θ fuera conocido, (1.30) se simplifica: $p(\theta | y_{0:k})$ es degenerada, y por tanto se pueden ignorar los parámetros de los términos condicionales. Luego $b_{k+1}(x_{k+1}, \theta) \equiv b_k(x_k)$ y (1.30) se convierte en (1.6), en cuyo caso se puede usar el filtro de partículas tal como se describe en la Sección 1.2. En caso contrario, si se desconoce θ , se hace explícito el hecho que se tiene que lidiar con el problema relacionado al desconocimiento de la forma teórica de la función de densidad $p(\theta | y_{0:k})$ para poder filtrar respecto a la combinación entre el estado y los parámetros [5].

1.3.1. Evolución artificial de parámetros

En cuanto al hecho de lidiar con estados variantes en el tiempo, [4] sugiere un enfoque para reducir la degeneración muestral mediante la suma de pequeñas perturbaciones aleatorias al estado de las partículas en cada instante de tiempo, además de todas las contribuciones de ruido evolucionario existentes. Desde ese entonces, esta idea ha sido extrapolada a modelos

paramétricos en la literatura [5]. En este caso, la idea es sumar pequeñas perturbaciones aleatorias a todos los parámetros de las partículas relacionadas con la densidad posterior a cada instante previo a la evolución al instante siguiente. Este método específico tiene una interpretación que surge de un modelo extendido en el cual los parámetros del modelo se consideran como si fueran variando en el tiempo —una “evolución artificial”. Esto es, se toma en cuenta un modelo diferente en el cual θ es reemplazado por θ_k en el instante k , y simplemente se incluye θ_k en un vector de estados aumentado. Luego se suma un incremento independiente de media cero a los parámetros a cada instante [5]. Esto es,

$$\theta_{k+1} = \theta_k + \xi_{k+1} \quad (1.32)$$

$$\xi_{k+1} \sim N(0, \mathbf{W}_{k+1}) \quad (1.33)$$

para alguna matriz de covarianza específica \mathbf{W}_{k+1} y donde θ_k, ξ_{k+1} son condicionalmente independiente para un $y_{0:k}$ dado. Con el modelo modificado con el vector de estado correspondientes, los métodos de filtrado convencionales, como el descrito en la Sección 1.2, son aplicables. La idea motivadora clave es que la evolución artificial provee mecanismos para generar nuevos valores para los parámetros a cada instante de tiempo en las simulaciones, de forma que ayuda a atacar los problemas de degeneración muestral presente en métodos que mantienen el mismo conjunto de puntos paramétricos entre instantes.

Entre las desventajas de este enfoque, una clave es que los parámetros en un modelo de parámetros fijos debieran permanecer invariantes. El pretender que estos parámetros son variantes en el tiempo implica una “pérdida de información” artificial entre instantes, lo cual deriva en densidades posteriores que son, eventualmente, demasiado difusas en relación a las densidades posteriores teóricas para los parámetros fijos verdaderos [5]. Luego, si se adopta un modelo en el cual todos los parámetros están sujetos a alteraciones aleatorias a cada instante, la precisión de las inferencias obtenidas se limita inevitablemente.

1.4. Predicción de eventos catastróficos

1.4.1. Detección de fallas

La detección de fallas es de utilidad en diversos sistemas (mecánicos, eléctricos, etc.) pues juega un rol importante en el logro de una operación que sea fiable y eficiente en término de costos [8].

A continuación se presentan algunas etapas que, de acuerdo a [7], se pueden emplear a la hora de intentar detectar una anomalía o falla.

1.4.2. Preprocesamiento de los datos

Para intentar dar robustez al sistema de detección de fallas, evitando falsas alarmas, se hace necesario analizar previamente los datos de sus mediciones, puesto que para una buena

detección se debe tener una alta calidad en la información que proveen los sensores presentes en la planta en estudio [7].

La validación de las mediciones es seguida por el preprocesamiento de las señales, paso empleado, por ejemplo, para reducir la dimensión de los datos, mejorar su relación señal a ruido [6]. Entre las técnicas consideradas para el preprocesamiento se consideran, entre otras, la compresión de los datos, el filtrado de los datos, el cálculo de promedios de las señales y la aplicación de la transformada de Fourier [6].

1.4.3. Extracción de la característica

Se denomina *característica* a la señal originada a partir de pre-procesamiento de los datos adquiridos para efectos de realizar una correcta detección. Esta característica puede obtenerse desde una multitud de fuentes, dentro de las cuales se incluyen tanto los datos de evaluaciones como conocimiento del sistema y su historial de funcionamiento [7].

La correcta selección y extracción de la característica constituye la etapa más importante para lograr detectar y diagnosticar una falla de manera efectiva [6].

1.4.4. Fusión de los datos provenientes de los sensores y de las características

Por un lado, la fusión de los datos de medición que provienen de distintas fuentes permiten minimizar el efecto del ruido sobre ellos e incrementar la calidad de la señal. Lo anterior se traduce en una reducción en la aparición de falsas alarmas y en un aumento en la confiabilidad de la detección [6].

Por otra parte, la fusión de las características permite reducir su dimensión para obtener un indicador que incremente la confiabilidad de la detección y simplifique la utilización de indicadores de tipo umbral para una señal [6].

1.4.5. Toma de decisiones

El proceso de toma de decisiones se basa en la información que provee el diagnóstico de la falla y en la predicción de su evolución en el tiempo. Luego, la decisión se debe tomar a partir de una cadena de suposiciones, cálculos, modelos, todos falibles en alguna medida. Es por ello que una decisión certera debe apoyarse en algoritmos que sean capaces de interpretar correctamente las características relevantes extraídas del sistema [7].

1.5. Filtro de partículas aplicados al pronóstico de sistemas

Un pronóstico puede ser entendido como la generación de predicciones a largo plazo, en ausencia de nuevas mediciones, que describen la evolución temporal de una señal de interés en particular o un indicador de falla [6].

Como se expuso en la Sección 1.2.1, los filtros de partículas exigen para su funcionamiento la disponibilidad de mediciones. Luego, para eludir la falta de observaciones inherente en el contexto de un pronóstico, es necesario introducir algunas consideraciones adicionales en la implementación del algoritmo. Estas dicen relación con el desarrollo de un procedimiento capaz de proyectar las partículas en el tiempo, aún cuando no existan nuevas mediciones, con un nivel de incertidumbre lo más bajo posible [8].

1.5.1. Generación de predicciones a plazos definidos

La generación de predicciones a M pasos de la densidad condicional puede obtenerse a partir de la expresión (1.3), el modelo de la planta, el vector de parámetro θ y la estimación actual del estado [8]. Como resultado de esta combinación se tiene que

$$\begin{aligned} \tilde{p}(x_{k+M}|y_{1:k}) &= \int \tilde{p}(x_k|y_{1:k}) \prod_{j=k+1}^{k+M} p(x_j|x_{j-1}) dx_{k:k+M-1} \\ &\approx \sum_{i=1}^N w_k^{(i)} \int \dots \int p(x_{k+1}|x_k^{(i)}) \prod_{j=k+2}^{k+M} p(x_j|x_{j-1}) dx_{k+1:k+M-1} \end{aligned} \quad (1.34)$$

. Para poder dar solución a (1.34), a continuación se presentan dos alternativas: la actualización de los pesos y la proyección en el tiempo del valor esperado del estado.

1.5.2. Actualización de los pesos

En este caso, lo que se realiza en primera instancia es calcular el valor esperado de (1.3) para cada instante de tiempo k , considerando el valor asociado a cada partícula como condición inicial, i.e.,

$$x_{k+M}^{(i)} = \mathbb{E}[f_{k+M}(\hat{x}_{k+M-1}^{(i)}, \omega_{k+M})]; \hat{x}_k^{(i)} = \tilde{x}_k^{(i)} \quad . \quad (1.35)$$

Junto con lo anterior también se debe actualizar el peso de las partículas en cada instante k . Esto se realiza para reflejar que la función de densidad del estado pueda cambiar a causa del ruido y la no linealidad del problema [8].

Si se asume que los pesos iniciales $\{w_k^{(i)}\}_{i=1}^N$ representan adecuadamente la función de

densidad actual del estado, entonces

$$\tilde{p}(x_{k+r}|\tilde{x}_{1:k+r-1}) \approx \sum_{i=1}^N w_{k+r-1}^{(i)} \tilde{p}(x_{k+r}|\tilde{x}_{k+r-1}); r = 1, \dots, M \quad (1.36)$$

representa, a su vez, la densidad predicha del estado en instante de tiempo $k+r$ [8]. Nótese que (1.36) considera que $\hat{p}(x_{k+r}|\tilde{x}_{1:k+r-1})$ describe la distribución del estado en el instante de tiempo futuro $k+r$, $r = 1, \dots, M$ cuando se toma $\hat{x}_{1:k+r-1}$ como condición inicial.

1.5.3. Proyección en el tiempo del valor esperado de los estados

Esta alternativa plantea que (1.35) modela suficientemente bien la evolución de $\hat{x}_{1:k+r}^{(i)}$ y que por tanto no es necesario actualizar los pesos de las partículas, de modo que estos pesos se mantienen constantes [8].

El argumento que sostiene esta alternativa se basa en la suposición de que el error asociado a mantener constante los pesos de las partículas es despreciable con respecto a otras fuentes de error, tales como inexactitudes del modelo e hipótesis erróneas en los parámetros de ruido [8].

Capítulo 2

Problemática y metodología usada

2.1. Descripción del problema en estudio

2.1.1. Antecedentes técnicos

En el contexto de la sistema mecánico de un helicóptero, la *placa de engranaje planetario*¹ corresponde a unos de los componentes principales de la caja de transmisión del rotor principal (ver Figura 2.1), el cual se encarga de transmitir la energía mecánica del motor a su hélice principal. Una falla en este componente durante la operación puede llevar incluso a la pérdida de la aeronave[7].

La gravedad del problema fue así entendida luego de detectarse anomalías en un helicópteros modelo UH-60A . Esto obligó a dejar alrededor de mil de sus aeronaves (incluyendo algunas modelo UH-60Q, todas las UH60A y varios helicópteros derivados de este último modelo) en tierra para someterlas a revisión. La observación y registro del helicóptero que presentaba anomalías reveló que estas se debían a una fractura en su placa de engranaje planetario. Debido a que posteriormente se encontró el mismo tipo de falla en otras aeronaves, se comenzaron a financiar proyectos que se que se dedicaran a estudiar este inconveniente [7].

Por un tema de costos y complejidad, rediseñar la placa, reemplazarlas en todos los helicópteros en uso y darle mantenimiento no es una opción viable. Es por ello que estrategias que permitan diagnosticar la falla sin necesidad de sacar de servicio las aeronaves cobran vital importancia.

2.1.2. Caracterización de los datos usados

Tal como se explica en la Sección 1.4.3, en el estudio de fallas es usual la confección de una característica basada en datos empíricos para dimensionar la gravedad de la falla

¹Llamado *planetary gear carrier plate* en inglés.

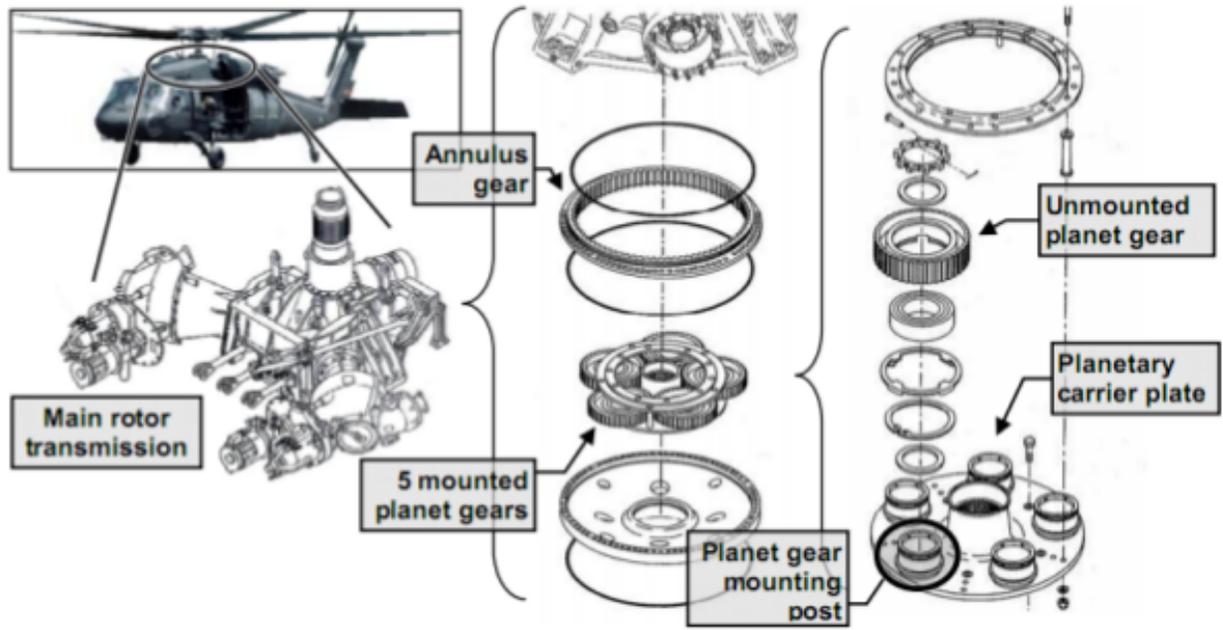


Figura 2.1: Componentes mecánicos de la transmisión del helicóptero [7]

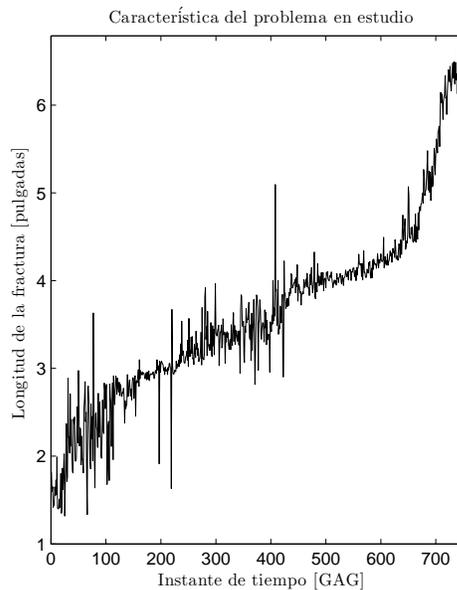


Figura 2.2: Estimación ruidosa de la longitud de la grieta en función de los instantes de tiempo GAG [7]

bajo análisis. En este caso, la presencia de una fractura en el engranaje planetario provoca vibraciones no esperadas en la caja de transmisión del helicóptero afectado, las cuales pueden usarse para estimar la longitud de la fractura.

Dado esto, se efectuó un experimento para estudiar la evolución de la falla descrita. En este experimento, la placa fue intervenida *a priori*, generando una grieta de 1.344 pulgadas en la misma. Acto seguido, se sometió la transmisión del helicóptero a un torque variable de

ciclo menor a 3 minutos. Cada ciclo de la prueba se denominó GAG², dado que representa el proceso de despegue y aterrizaje del helicóptero [7]. Se usaron dos perfiles de carga en la simulación: el primero se usó entre el GAG 1 y 320, en donde el torque variaba entre un 20 y un 120% de la carga de referencia; el segundo se usó desde el GAG 321 en adelante, con una variación reducida de torque entre un 20 y un 93% de la carga de referencia. Este cambio en los perfiles de carga se efectuó debido a que la grieta crecía con una rapidez mayor a la esperada, de modo que disminuyendo el torque máximo aplicado se buscaba poder obtener una mayor cantidad de datos antes de que la placa se fracturara completamente [8][7]. En la Figura 2.2 se presenta la estimación de la fractura obtenida a partir de las vibraciones sufridas por la caja de transmisión durante el procedimiento descrito.

El uso de los datos mostrados se justifica por los objetivos del trabajo a realizar. La estimación del largo de la fractura provee información que puede usarse como entrada de un algoritmo de pronóstico —en este caso el filtro de partículas— y además puede utilizarse para evaluar el desempeño mismo del algoritmo tanto en las criterios de filtrado como de predicción. Las medidas directas de la longitud de la grieta sirven para evaluar el desempeño de la estimación entregada por el filtro de partículas. Por último, la presencia de un cambio en el perfil de carga puede considerarse como una anomalía, y por tanto, pone a prueba las capacidades de detección de fallas del algoritmo usado.

2.2. Descripción general de la modelación

El algoritmo de pronóstico usado en este trabajo se basa en el uso de dos herramientas: curvas parametrizadas, que modela el progreso de la falla y representan el conocimiento previo que se tiene de la evolución de la misma; y filtro de partículas, usado para ajustar los parámetros de las formas paramétricas a los datos del problema a fin de que estas representen la trayectoria descrita por la falla, como también generar pronósticos de la longitud futura de la fractura en estudio. La idea básica es escoger una curva tal que se ajuste a la evolución del fenómeno de degradación a estudiar —la cual dependerá del grado de conocimiento que se posea sobre el fenómeno en cuestión— y que posea el menor número de parámetros variables posibles —a fin de no caer en problemas de observabilidad. A partir de esta curva —que se denominará de aquí en adelante como *curva de ajuste*— es posible construir un modelo del sistema donde el estado del sistema x_k está caracterizado por el valor de la función en cada instante k ; el vector de parámetros fijos θ queda determinado por los parámetros de definen dicha función; y la salida y_k corresponde sólo al valor de la función. Luego es posible procesar los datos haciendo uso de este modelo en conjunto con un filtro de partículas, obteniéndose estimaciones filtradas del conjunto de datos en estudio, así también como predicciones de acuerdo a los valores para un tiempo específico de los parámetros de la curva.

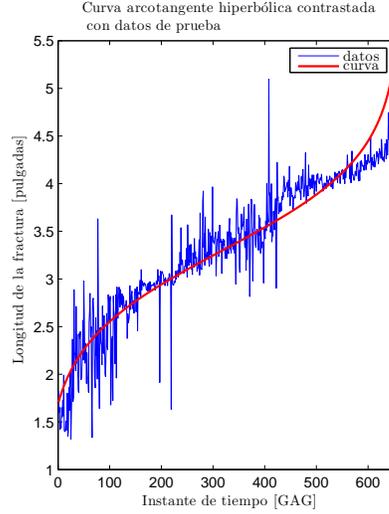


Figura 2.3: Comparación entre el conjunto de datos usado y la curva de ajuste escogida (arcotangente hiperbólica)

2.2.1. Curva a utilizar y modelo resultante

De acuerdo al conjunto de datos presentado en la Figura 2.2, se optó por usar la arcotangente hiperbólica como curva de ajuste, debido que al parametrizarla por una función lineal esta sólo posee dos parámetros a variar y que su forma se ajusta de forma razonable al conjunto de datos considerado (ver Figura 2.3). Tomando entonces esta curva, se tiene que el largo de la fractura para cada instante $L(t)$ queda dado por [11]:

$$L(t, a, b) = \operatorname{arctanh}(at + b) = \frac{1}{2} \ln \left(\frac{1 + at + b}{1 - at - b} \right) \quad (2.1)$$

La selección de parametrizar el argumento de la función de esta forma radica en que se se puede adecuar la forma de la curva a los datos en forma simple como se aprecia en la Figura 2.4: el parámetro a sirve para determinar el dominio de la función que define el proceso de degradación de la curva, mientras que b fija el centro de la curva a lo largo del eje de las abscisas. Por otra parte, el centro a lo largo del eje de las ordenadas no está definido; este se fijará en base al valor que tome la fractura en el instante cero. Nótese que esto, en conjunto con el esquema de modelación expuesto en el Capítulo 1, hace que la expresión (2.1), por ser continua, sea inadecuada. Una representación discreta, que resulta más adecuada para este caso, está dada por :

$$L_{k+1} = L_k + \Delta L(a, b, k) \quad (2.2)$$

$$\Delta L(a, b, k) = -\frac{a}{(ak + b)^2 - 1} \Delta k \quad . \quad (2.3)$$

Debido a que la diferencia entre los instantes es homogénea e igual a la unidad, se tiene

²*Ground-Air-Ground*, o Tierra-Aire-Tierra en español.

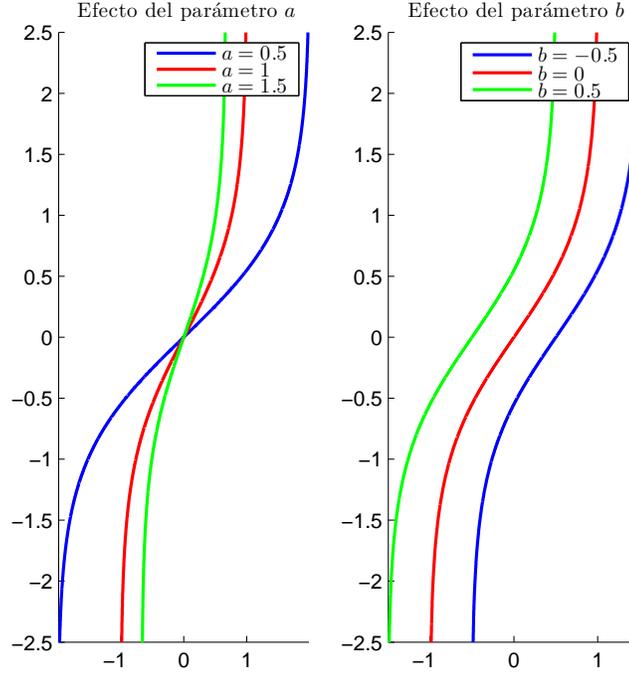


Figura 2.4: Efecto de la variación de parámetros en la forma de la arcotangente hiperbólica

que $\Delta k = 1 \forall k$. De esta forma, el sistema queda caracterizado por las siguientes ecuaciones de estado:

$$x_{k+1} = x_k - \frac{a}{(ak + b)^2 - 1} + \omega_{k+1} \quad (2.4)$$

$$y_k = x(k) + \nu_k \quad (2.5)$$

$$\theta = \begin{bmatrix} a \\ b \end{bmatrix}, \quad (2.6)$$

donde ω, ν son los ruidos de proceso y observación del modelo, respectivamente, tales que

$$\omega \sim \mathcal{N}(0, \sigma_\omega^2) \quad (2.7)$$

$$\nu \sim \mathcal{N}(0, \sigma_\nu^2), \quad (2.8)$$

y a, b son parámetros fijos del modelo.

2.2.2. Filtro de partículas

El uso del modelo previamente expuesto supone que el vector de parámetros θ es conocido. Luego, de acuerdo a lo presentado en la Sección 1.3, el modelo que usa el filtro de partículas

para definir la distribución de transición de estados debe ser el siguiente:

$$x_{k+1} = \begin{bmatrix} L_k - \frac{a}{(ak+b)^2-1} + \omega_{k+1}^{(L)} \\ a_k + \xi_{k+1}^{(a)} \\ b_k + \xi_{k+1}^{(b)} \end{bmatrix} \quad (2.9)$$

$$y_k = L_k + \nu_k \quad (2.10)$$

$$\xi^{(a)} \sim \mathcal{N}(0, \sigma_{\xi^{(a)}}^2) \quad (2.11)$$

$$\xi^{(b)} \sim \mathcal{N}(0, \sigma_{\xi^{(b)}}^2) \quad (2.12)$$

En lo referente a la implementación del filtro de partículas, se determinó que la función de densidad $q_k(x_k|x_{k-1})$ tomara la siguiente forma:

$$q_k(x_k|x_{k-1}) = p(x_k|x_{k-1}) \quad (2.13)$$

lo cual, de acuerdo a lo expuesto en la Sección 1.2.1.2, implica que la actualización de pesos queda dada por:

$$w(x_k) \propto w(x_{k-1})p(y_k|x_k) \quad (2.14)$$

La verosimilitud $p(y_k|x_k)$ se determina de acuerdo a lo expuesto en (2.10). Como el único término variable corresponde al ruido Gaussiano ν , se tiene que:

$$p(y_k|x_k) = \frac{1}{\sqrt{2\pi\sigma_\nu^2}} e^{-\frac{\nu^2}{2\sigma_\nu^2}} \quad (2.15)$$

Nótese que, si se tiene el valor del largo de la fractura para un tiempo determinado, ν se puede calcular a partir de (2.5) como:

$$\nu = y_k - L_k \quad (2.16)$$

por lo que (2.15) puede calcularse como:

$$p(y_k|x_k) = \frac{1}{\sqrt{2\pi\sigma_\nu^2}} e^{-\frac{\nu^2}{2\sigma_\nu^2}} \quad (2.17)$$

Con esto, la actualización de los pesos de cada partícula $w(x_k^{(i)})$, de acuerdo a lo expuesto en la Sección 1.2.1.3, queda dada por las siguientes condiciones:

$$w(x_k^{(i)}) = w(x_k^{(i)}) \frac{1}{\sqrt{2\pi\sigma_\nu^2}} e^{-\frac{\nu^2}{2\sigma_\nu^2}} \quad (2.18)$$

$$w(x_k^{(i)}) = \sum_{i=1}^N w(x_k^{(i)}) \frac{1}{\sqrt{2\pi\sigma_\nu^2}} e^{-\frac{\nu^2}{2\sigma_\nu^2}} \quad (2.19)$$

En lo referente a la generación de predicciones, se optó por proyectar en el tiempo el valor esperado de los datos (ver Sección 1.5.3). De esta manera, el espacio de estados toma la siguiente forma:

$$x_{k+1} = \begin{bmatrix} L_k - \frac{a}{(ak+b)^2-1} \\ a_k \\ b_k \end{bmatrix} , \quad (2.20)$$

lo que es equivalente a proyectar la función arcotangente con parámetros fijos conocidos para cada partícula.

2.2.2.1. Condiciones iniciales

Dada la naturaleza del algoritmo y los estudios realizados, se comprobó que el filtro es sensible a los valores iniciales asignados a los parámetros de la curva de ajuste. De acuerdo con esto, estos valores deben ser escogidos dentro de una vecindad tal que permita resultados coherentes con lo deseado i.e. deben cumplir con el requisito de ser valores típicos de la operación a estudiar. De esto último se extrae que es necesario un conocimiento *a priori* del proceso para efectuar este ajuste.

Teniendo en cuenta lo anterior, para el caso de estudio presentado en 2.1, los valores iniciales de los parámetros se obtuvieron de la siguiente manera: se generó la arcotangente hiperbólica que se ajustara mejor a la característica del problema, y en base a esta se obtuvo el valor de sus parámetros (que en adelante se denominarán a_{init} y b_{init}); el criterio para determinar la calidad del ajuste fue el del error mínimo cuadrático. Una vez obtenidos estos valores, estos se incorporan al modelo, y los valores iniciales a_0^i y b_0^i asociados a cada partícula i se calculan al comienzo de cada realización experimental de la siguiente forma:

$$a_0^i = a_{\text{init}} + \rho_a \quad (2.21)$$

$$b_0^i = b_{\text{init}} + \rho_b \quad , \quad (2.22)$$

donde ρ_a, ρ_b corresponden a ruidos uniformes en los rangos $[-1 \times 10^{-6}, 1 \times 10^{-6}]$ y $[-1 \times 10^{-5}, 1 \times 10^{-5}]$ respectivamente. Los órdenes de magnitud de los ruidos fueron escogidas de forma arbitraria.

En el caso del cálculo de los valores para L_0^i , de forma arbitraria se optó por que el valor asociado a este parámetro se encontrara entre el largo conocido de la fractura previo a la aplicación de carga y el primer valor del conjunto de datos a utilizar. De esta forma L_0^i queda como:

$$L_0^i = 1,344 + (y_1 - 1,344)\rho_L \quad , \quad (2.23)$$

donde la cifra 1.344 corresponde al valor de la grieta previo a la aplicación del torque mecánico (ver Sección 2.1.2), y_1 es el largo de la grieta en el primer instante de acuerdo a conjunto de datos usados y ρ_L es ruido uniforme en el rango $[-1, 1]$.

2.2.2.2. Variantes del proceso de actualización

Además de obtenerse resultados de pronósticos a partir del procedimiento anteriormente descrito, también se procuró generar resultados por medio de métodos alternativos. Estos métodos se diferencian principalmente por su mecanismo de actualización, siendo la idea clave siguiente: dado un instante K determinado, se proyecta el valor esperado del estado de cada partícula (ver Sección 1.5.3), pero en lugar de construir esta trayectoria en el rango $[K, K+r]$, $r > K$, ésta se arma en el rango $[0, K]$; en otras palabras, el propósito es generar una colección de N proyecciones $\{\hat{L}_{0:k}^{(i)}\}_{i=1,\dots,N}$, asumiendo el siguiente modelo:

$$\hat{L}_{k+1}^{(i)} = \hat{L}_k^{(i)} - \frac{a_K^{(i)}}{(a_K^{(i)}k + b_K^{(i)})^2 - 1} \quad (2.24)$$

donde $a_K^{(i)}, b_K^{(i)}$ son los valores que toman los parámetros para cada partícula i en el instante K de acuerdo al modelo definido por las expresiones (2.9) y (2.10). Nótese que la trayectoria está definida en el rango temporal $[0, K]$; tanto de denominará *proyección acumulada de una trayectoria en el pasado*. Esto abre la posibilidad de construir la proyección en el pasado en una ventana de tiempo móvil, i.e., que la trayectoria quede definida para un rango $[K-M, K]$, donde M es un número fijo y corresponde al número de muestras de la ventana. Esta variante se denominará *proyección móvil de una trayectoria en el pasado*, y se obtiene mediante el cálculo de la proyección móvil acumulada, seguido del descarte de todas las estimaciones asociadas a instantes de tiempo menores a $M-K$.

El objetivo tras el uso de las proyecciones en el pasado es evaluar, de acuerdo a la información pasada del estado, que tan buenas son las estimaciones de los parámetros. Esto tiene sentido puesto que, idealmente, los parámetros deberían ser fijos de acuerdo al modelo, y por consecuencia las proyecciones de trayectorias en el pasado de las partículas cuyos parámetros estén cercanos a los verdaderos debieran, en promedio, ajustarse bien a la característica en estudio.

2.2.2.2.1. proyección de una trayectoria en el pasado usando la distribución chi-cuadrado para el cálculo de la verosimilitud

De acuerdo con lo descrito, para cada partícula $x_k^{(i)}$ es posible construir una proyección de una trayectoria en el pasado; de la misma forma, se puede obtener el error $\varepsilon_k^{(i)}$ entre la proyección en el pasado de cada partícula y la colección de datos de la característica en estudio para una ventana de tiempo definida entre $k-M$ y k . Como la proyección consiste en un conjunto de estimaciones del estado, se tiene que:

$$\varepsilon_{k-M:k} = y_{k-M:k} - \hat{L}_{k-M:k} \quad (2.25)$$

Bajo el supuesto que las estimaciones $\hat{L}_{k-M:k}$ son insesgadas, se sigue que

$$\varepsilon_j^{(i)} \sim \mathcal{N}(0, \sigma_\nu^2), \quad j = k-M, \dots, k \quad (2.26)$$

Esto implica que, si se normalizan los errores por medio de

$$\bar{\varepsilon}_j^{(i)} = \frac{\varepsilon_j^{(i)} - \mu_\nu}{\sigma_\nu} \quad \forall j \in [k - M, k] \quad (2.27)$$

se obtiene que $\bar{\varepsilon}_j^{(i)} \sim \mathcal{N}(0, 1)$, $j = k - M, \dots, k$, y por consiguiente $\sum_{j=k-M}^k (\bar{\varepsilon}_j^{(i)})^2 \sim \chi_M^2$. De esta forma la verosimilitud del modelo, tomando en cuenta todos los estados entre cero y k , se puede considerar como la densidad de una distribución chi-cuadrado de M grados de libertad, y por tanto se puede calcular por medio de

$$\bar{p}(y_k | x_{k-M:k}^{(i)}) = \begin{cases} \frac{\bar{\varepsilon}^{(i)\frac{M}{2}-1} e^{-\frac{\bar{\varepsilon}^{(i)}}{2}}}{2^{\frac{M}{2}} \Gamma(\frac{M}{2})} & \text{si } \bar{\varepsilon}^{(i)} \geq 0 \\ 0 & \text{si } \bar{\varepsilon}^{(i)} < 0 \end{cases} \quad (2.28)$$

donde γ es la función Gamma incompleta³. Esto tiene sentido dado que la suma de Gaussianas distribuye chi-cuadrado. De esta manera, siguiendo la idea del filtro de partículas previamente presentado, es posible actualizar los pesos de las partículas de la siguiente forma:

$$w(x_k^{(i)}) = w(x_{k-1}^{(i)}) \bar{p}(y_k | x_{0:k}^{(i)}) \quad (2.29)$$

2.2.2.2. proyección de una trayectoria en el pasado usando la distribución acumulada complementaria de una chi-cuadrado para el cálculo de la verosimilitud

Téngase en cuenta que, a pesar de que el uso de la distribución chi-cuadrado aflora de un resultado conocido, esto no impide el uso de otras distribuciones para tratar con el error entre la proyección de una trayectoria en el pasado y la característica. De esta forma, se planteó otra variante, en la cual se calculó la verosimilitud a partir de la función distribución acumulada complementaria (CCDF⁴) de la distribución chi-cuadrado (ver Figura 2.5). Esta variante se escogió bajo el criterio que no castiga la existencia de errores muy pequeños para altos grados de libertad. De esta forma la verosimilitud queda definida como:

$$\begin{aligned} \check{p}(y_k | x_{k-M:k}^{(i)}) &= 1 - \text{bar}P(y_k | x_{k-M:k}^{(i)}) \\ \Rightarrow \check{p}(y_k | x_{k-M:k}^{(i)}) &= \begin{cases} 1 - \frac{\gamma(\frac{M}{2}, \frac{\bar{\varepsilon}^{(i)}}{2})}{\Gamma(\frac{M}{2})} & \text{si } \bar{\varepsilon}^{(i)} \geq 0 \\ 0 & \text{si } \bar{\varepsilon}^{(i)} < 0 \end{cases}, \end{aligned} \quad (2.30)$$

donde $\bar{P}(y_k | x_{0:k}^{(i)})$ es la distribución acumulada de una chi-cuadrado y γ es la función Gamma incompleta⁵ y, de forma análoga al caso anterior, la actualización de los pesos que dada por

$$w(x_k^{(i)}) = w(x_{k-1}^{(i)}) \check{p}(y_k | x_{0:k}^{(i)}) \quad (2.31)$$

³La función Gamma se define como $\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt$, para todo z complejo con parte entera mayor a cero.

⁴*Complementary Cumulative Distribution Function*

⁵La función Gamma incompleta se define como $\gamma(s, x) = \int_0^x t^{s-1} e^{-t} dt$.

Distribución χ^2 acumulada complementaria para diferentes grados de libertad

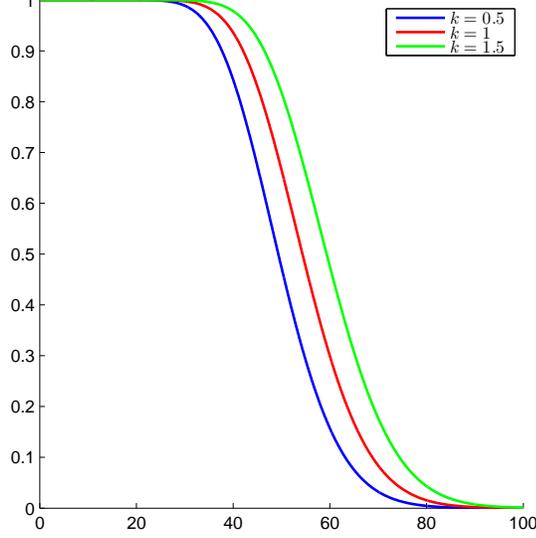


Figura 2.5: Perfil de la verosimilitud basada en la distribución chi-cuadrado acumulada complementaria

2.2.3. Marco de pruebas

El objetivo de las pruebas es el de evaluar el desempeño de las diversas dimensiones que componen el proceso de pronóstico de eventos catastróficos. Estas fueron la estimación del valor real de la fractura en el presente —también llamado proceso de filtrado, la predicción de la evolución de la grieta en el tiempo y por último la detección de eventos anómalos, que en este caso correspondería al cambio en el perfil de carga descrito en la Sección 2.1.2.

Para evaluar el desempeño del algoritmo en el área de filtrado, se usaron dos criterios. El primero consta de una inspección visual de la estimación entregada por el filtro, contrastada con la evolución de los datos. Dentro de este criterio también se añade la exposición del “intervalo de confianza” de los datos, donde los márgenes superior e inferior para cada instante k corresponden respectivamente a:

$$M_{\text{sup}}(k) = \sup\{L_k^{(i)} : 1 \leq i \leq N\} \quad (2.32)$$

$$M_{\text{inf}}(k) = \inf\{L_k^{(i)} : 1 \leq i \leq N\} \quad (2.33)$$

Nótese que de esta forma correspondería a un intervalo de confianza del 100%, lo cual no es muy correcto desde el punto de vista teórico pero entrega una buena aproximación de la incertidumbre involucrada en las mediciones.

Este criterio se complementa con el segundo, el cual consiste en el uso de error cuadrático medio (ECM), i.e. :

$$\eta_{\text{est}} = ECM = \frac{1}{L_e} \sum_{k=1}^{L_e} (y_k - \hat{y}_k)^2 \quad (2.34)$$

donde L_e corresponde al horizonte de filtrado.

El desempeño en cuanto a predicción del algoritmo se evaluó proyectando una curva de ajuste por cada partícula $x_k^{(i)}$ de acuerdo a los parámetros estimados de la partícula correspondiente en el tiempo k . Estas proyecciones fueron luego combinadas linealmente de acuerdo al peso de la partícula originaria para obtener una proyección definitivamente. Luego, de forma análoga al caso de la evaluación del filtrado, se contrastó esta proyección con el conjunto de datos de forma visual, para complementar esto se usó la siguiente fórmula como medida de desempeño:

$$\eta_{\text{pred}} = \sum_{k=k_0}^{L_p} \left(\frac{y_k - \hat{y}_k}{\max_{1 \leq i \leq N} y_k^{(i)} - \min_{1 \leq i \leq N} y_k^{(i)}} \right)^2 \quad (2.35)$$

donde el numerador es el error entre la predicción estimada y los datos de entrada, el denominador corresponde al intervalo de confianza obtenido en base a las predicciones del conjunto de partículas usado, y las cantidades k_0 y L_p definen el plazo de predicción usado en la medida.

Para finalizar, la detección se evaluó de forma visual, ya sea contrastando la salida en el tiempo del filtro con los datos a disposición, como también analizando la evolución de los parámetros de la curva de ajuste de acuerdo con las partículas usadas en el filtro.

2.3. Implementación del modelo y los filtros

2.3.1. Herramientas utilizadas en el desarrollo del trabajo

Para la implementación de procedimiento descrito, se usó el software MATLAB⁶. En particular se usaron las librerías que vienen por defecto en la versión 7.14, más las funciones `chi2pdf.m` y `cdf` del *Statistics Toolbox* (versión 8.0). Además, las rutinas se efectuaron en su mayoría en un computador con sistema operativo Linux, el cual posee 2[GB] de memoria RAM y un procesador Intel Core 2 Duo de 2.13[GHz].

2.3.2. Consideraciones respecto a la programación

Uno de los aspectos a tomar en cuenta a la hora de implementar las ecuaciones de estado del sistema, es que la función de ajuste a usar —la función arcotangente hiperbólica— es una función periódica. Esto hace necesario que su dominio sea limitado a fin de que el largo de la fractura de las partículas no pueda pasar de valores de gran magnitud positiva a valores de gran magnitud negativa entre instantes contiguos. De esta forma, se determinó empíricamente que para se obtiene resultados razonables implementando (2.3) de la siguiente forma:

$$\Delta L(a, b, k) = \begin{cases} -\frac{a}{(ak+b)^2-1} & \text{si } (ak+b) \leq 1 - 10^{-5} \\ 0 & \text{si } (ak+b) > 0 \end{cases} \quad (2.36)$$

⁶MATLAB es propiedad de la The MathWorks, Inc.

Por otra parte, existe la posibilidad de que al calcular la verosimilitud $p(y_k|x_k)$ en algún instante ésta sea nula, debido a las limitaciones computacionales asociadas al cálculo de las distribuciones involucradas. Este detalle se corrige de forma parcial, de acuerdo con [7], mediante la suma del valor 1×10^{-99} , lo cual previene una degeneración acelerada del algoritmo.

2.3.3. Descripción del código implementado

En conformidad con el trabajo realizado en [7], se agruparon los datos en base a *estructuras*⁷. De esta forma las variables usadas se aglomeran en torno a un “concepto” que las relaciona, ya sea en virtud de un rol específico dentro del programa o para detallar las variables a disposición del usuario de turno; este concepto da el origen a una variable “madre”, de la cual se desprenden variables “hijas”, como por ejemplo `particles.weights` (que muestra los pesos de las partículas) o `particles.value` (que muestra los valores que adquieren las partículas). Esta organización se efectuó para seguir la línea del código reutilizado de [7], en conformidad con los ideales de orden, legibilidad e intuición expuestos en el mismo trabajo. Las variables “madres” se presentan en la Tabla 2.1. Para la descripción de las variables “hijos” se puede consultar el código incluido en el Anexo B de forma directa.

Tabla 2.1: Estructuras de datos utilizadas en los códigos fuente

| Estructura | Descripción |
|------------------------|--|
| <code>noise_p</code> | Ruido de proceso (distribución normal) |
| <code>noise_o</code> | Ruido de observación (distribución normal) |
| <code>model</code> | Parámetros asociados al modelo |
| <code>particles</code> | Parámetros y variables asociadas con las partículas |
| <code>output</code> | VARIABLES DE SALIDA DE LA RUTINA <code>MAIN_atanh.m</code> |
| <code>PARAM</code> | Parámetros y variables de ajuste usadas en las pruebas |
| <code>OUTPUT</code> | VARIABLES DE SALIDA USADAS EN LAS PRUEBAS |
| <code>NOISE</code> | VARIABLES RELACIONADA CON LOS NIVELES DE RUIDO (DE PROCESO Y OBSERVACIÓN) UTILIZADOS |

En cuanto a los archivos usados, estos constan de rutinas modificadas del trabajo efectuado en [7], además de código nuevo escrito para satisfacer los requerimientos de los algoritmos usados en este trabajo. Sin embargo, en discrepancia con [7], se prefirió agrupar todos los filtros creados en una sola rutina (`MAIN_atanh.m`).

De esta manera, se puede identificar las rutinas principales como `MAIN_atanh.m`, la cual ejecuta el algoritmo relacionado con los filtros y cuyo diagrama de flujo se presenta en la Figura 2.6; y `tests.m`, la cual fija parámetros y hace uso de `MAIN_atanh.m` para efectuar las pruebas necesarias en el trabajo. Todas las rutinas usadas se explican de forma resumida en la Tablas 2.2 y 2.3.

⁷En conformidad con la definición en el contexto del lenguaje usado por el software MATLAB.

Tabla 2.2: Estructura de archivos utilizadas

| Nombre | Descripción |
|--------------------------|--|
| MAIN_atanh.m | Rutina principal para el filtro de partículas. |
| tests.m | Rutina usada para realizar las pruebas a los filtros. Los resultados obtenidos se guardan en archivos en disco |
| update_particles_atanh.m | Función que se ocupa de actualizar los valores y pesos de las partículas para el filtro de partículas. |
| gauss_sample.m | Función que obtiene una muestra a partir de una suma de <i>Gaussianas</i> . |
| gauss_likelihood.m | Función que calcula la verosimilitud de una predicción en base a una distribución normal. |
| chi2test_atanhV1.m | Función que computa la verosimilitud de una predicción a partir de la proyección móvil de una trayectoria en el pasado, usando una densidad de probabilidad chi-cuadrado de grados de libertad igual al número de muestras en la ventana usada |
| chi2test_atanhV2.m | Función que computa la verosimilitud de una predicción a partir de la proyección acumulada de una trayectoria en el pasado, usando una densidad de probabilidad chi-cuadrado de grados de libertad igual al número de muestras en la ventana usada |
| chi2test_atanhV3.m | Función que computa la verosimilitud de una predicción a partir de la proyección móvil de una trayectoria en el pasado, usando la CCDF de la distribución chi-cuadrado de grados de libertad igual al número de muestras en la ventana usada |
| chi2test_atanhV4.m | Función que computa la verosimilitud de una predicción a partir de la proyección acumulada de una trayectoria en el pasado, usando la CCDF de la distribución chi-cuadrado de grados de libertad igual al número de muestras en la ventana usada |
| resampling.m | Función que se ocupa del remuestreo de partículas |
| residualresample.m | Método que ocupa <code>resampling.m</code> para realizar el remuestreo |
| output_mean_atanh.m | Función que calcula la esperanza matemática de un conjunto de datos |

Tabla 2.3: Estructura de archivos utilizadas (continuación)

| Nombre | Descripción |
|-------------------------|---|
| D_atanh.m | Función que entrega el diferencial de la curve de ajuste para cada partícula y para un conjunto de instantes de tiempo determinado |
| funcPred.m | Función que genera la predicción de las curvas para cada instante de tiempo considerado en la simulación |
| estErr.m | Función que se ocupa de calcular el error de estimación para cada instante de tiempo |
| predErr.m | Función encargada de computar los errores de predicción para corto, mediano y largo plazo |
| confRange.m | Función encargada de generar un vector de rangos entre el estado de mayor valor y el estado de menor valor para una colección de partículas en un instante determinado. |
| paramGen.m | Función encarga de generar las colecciones ruidos de proceso y observación para la realización de las pruebas. |
| currentStatePrognosis.m | Función que entrega la proyección en el tiempo del valor esperado de la función para cada partícula |

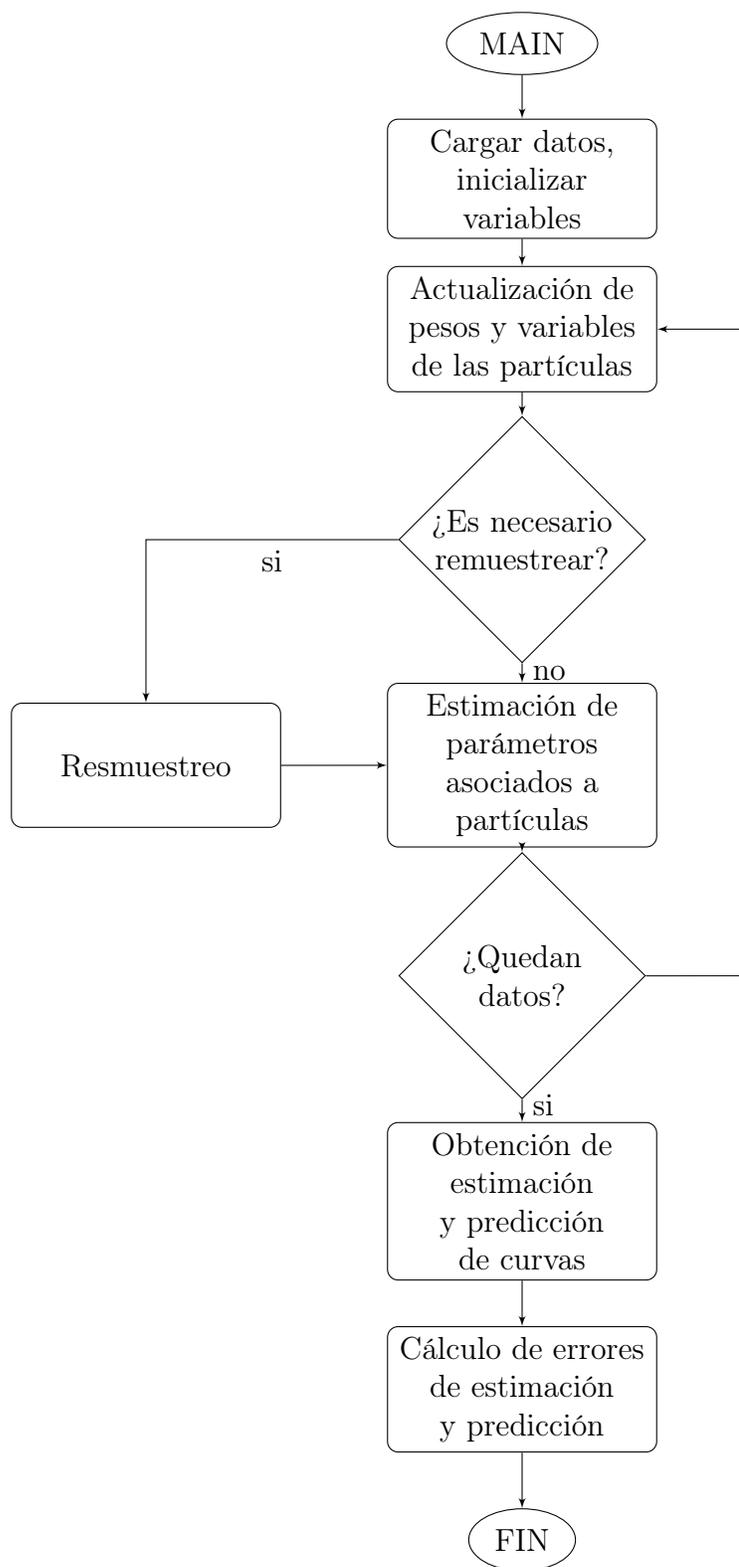


Figura 2.6: Funcionamiento básico del programa

Capítulo 3

Discusión de resultados

En este capítulo se expondrán los resultados obtenidos al procesar los datos correspondientes a la estimación de la longitud de una fractura en la placa de engranaje planetario de una helicópteros (ver Sección 2.1), usando filtros de partículas asumiendo un modelo de parámetros fijos y estados variantes (ver Sección 1.3). En particular, se asumió que los parámetros definen una curva de ajuste, que en este caso corresponde a una arcotangente hiperbólica (ver Sección 2.2). Los resultados se obtuvieron haciendo uso de cinco estrategias de filtrado diferentes, las cuales se describen a continuación:

Estrategia de filtrado 1: No considera la proyección de una trayectoria en el pasado de la curva de ajuste.

Estrategia de filtrado 2: Considera la proyección móvil de una trayectoria en el pasado de la curva de ajuste a lo largo de una ventana de 50 muestras, usando la distribución chi-cuadrado para el cálculo de la verosimilitud.

Estrategia de filtrado 3: Considera la proyección acumulada de una trayectoria en el pasado de la curva de ajuste, usando la distribución chi-cuadrado para el cálculo de la verosimilitud.

Estrategia de filtrado 4: Considera la proyección móvil de una trayectoria en el pasado de la curva de ajuste a lo largo de una ventana de 50 muestras, usando la CCDF de la distribución chi-cuadrado para el cálculo de la verosimilitud.

Estrategia de filtrado 5: Considera la proyección acumulada de una trayectoria en el pasado de la curva de ajuste, usando la CCDF de la distribución chi-cuadrado para el cálculo de la verosimilitud.

Las variables y parámetros usadas en los estrategias de filtrado se exponen en la Tabla 3.1 y que los resultados presentados en este capítulo corresponde al promedio de 30 simulaciones. Nótese de la misma que los ruidos asociados al vector de parámetros tienen poca variabilidad, mientras que la varianza asociada al ruido de proceso es alta. Esto fue una elección deliberada, con el fin de evitar la difusión de la densidades posteriores (ver Sección 1.3.1) y el empobrecimiento de la muestra (ver Sección 1.2.1.1). Por último considérese que los resultados

mostrados corresponden a un promedio obtenido a partir de 30 realizaciones, como forma de probar la consistencia de lo entregado por el algoritmo a lo largo de diversas instancias.

Tabla 3.1: Variables usadas en los filtros estudiados

| Variable | Valor |
|----------------------|---------------------|
| σ_ν^2 | 0, 15 |
| σ_ω^2 | 1×10^{-3} |
| $\sigma_{\xi_a}^2$ | 1×10^{-12} |
| $\sigma_{\xi_b}^2$ | 1×10^{-12} |
| Número de partículas | 60 |

En la Sección 2.1 se detalla la información relacionada con los datos usados como entrada de los filtros, como las condiciones experimentales usadas para su obtención y los perfiles de carga mecánica usados; si se quiere obtener mayor detalles acerca de la naturaleza de los estrategias de filtrado previamente enumeradas, se recomienda consultar la Sección 2.2.2; en cambio si se requiere conocer más a fondo los por menores de la implementación, se sugiere consultar las Secciones 2.3.2 y 2.3.3.

3.1. Análisis preliminar de las estrategias de filtrado

3.1.1. Filtro de partículas “clásico”

El primer uso que se le da al filtro se centra en el filtrado de la señal de entrada. Dicha señal es una estimación ruidosa; luego el objetivo es ver la posibilidad de extraer información relevante a través de la operación mencionada. Así la primera utilidad del filtro es estimar el valor real de la fractura, lo que se presenta gráficamente en la Figura 3.1.

En cuanto al desempeño en el filtrado destaca el hecho que distinguen dos zonas, delimitadas aproximadamente alrededor del GAG 400, donde en cada zona la curva que caracteriza la estimación parece tomar parámetros claramente distintos. Se aprecia además que en la primera zona la predicción es mejor que en la segunda. Esto es interesante puesto que tal diferenciación de las curvas ocurre después del cambio en el perfil de carga, y por lo tanto puede asumirse como un mecanismo de detección. La baja en el rendimiento luego del cambio también puede explicarse dentro de esos términos; la calidad del filtrado será baja puesto que los parámetros estaban ajustados primer perfil de carga usado, y el algoritmo muestra dificultades de adaptación en el segundo perfil usado.

En la Figura 3.3 se presenta la predicción entregada por cada una de las partículas del filtro en el GAG 350 (en verde). Este instante, a partir del cual se muestra la tendencia predicha por el filtro, se escogió de forma arbitraria con la condición de que ser mayor a 320 [GAG], momento en el cual se introduce un cambio en las condiciones del experimento. Se observa que las predicciones no poseen ruido, y que siguen la forma de una arcotangente hiperbólica, lo cual se debe a la ausencia de ruidos (tanto de proceso como de observación) en

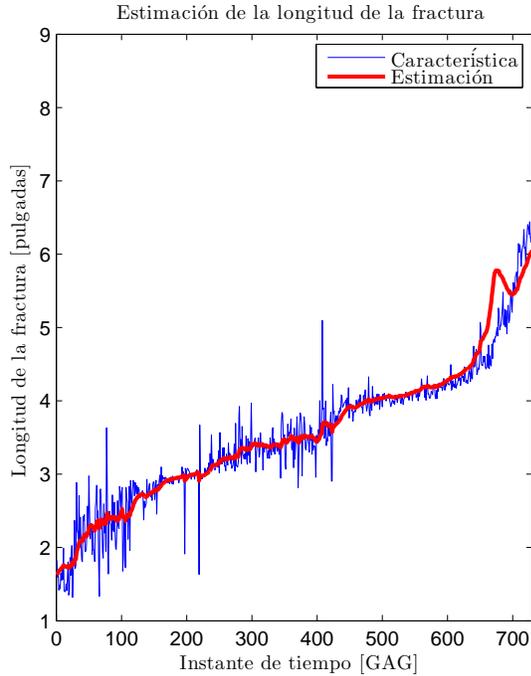


Figura 3.1: Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 1.

la confección de las predicciones, además de la fijación de los parámetros. Se aprecia también que, a pesar de que todas las proyecciones comienzan en una vecindad común, estas tienden a divergir a medida que aumenta el recorrido de su trayectoria. Tal efecto revela que el vector de parámetros es distinto para cada partícula —lo cual demuestra la variabilidad de éstos incorporada por el filtro— y que incluso las variaciones pequeñas en los valores del vector —evidenciada por el bajo valor asociado a la varianza de ξ_a y ξ_b — se puede traducir en un aumento, en este caso evidente aunque poco significativo, en la incertidumbre asociada al pronóstico de largo plazo.

En cuanto al vector de parámetros, en la Figura 3.4 se presenta la evolución de sus componentes. De esta última queda claro que la evolución de ambas muestra una alta estabilidad, evidenciado por la poca variabilidad exhibida por el valor que toman los parámetros. El parámetro a se muestra siempre cercano al valor 0.003, y presenta una ligera tendencia disminuir el valor, excepto a partir del GAG 600 donde tal tendencia se rompe y a disminuye su valor de forma brusca en casi 2%. Esto se explica como un intento desesperado del filtro por intentar demorar el crecimiento de la estimación en fin de ajustarse de mejor manera a la característica en esos instantes. En cuanto al parámetro b , este se mantiene también en una vecindad alrededor del valor 0.9552. Como punto interesante cabe decir que en este parámetro ocurren tres cambios de tendencia: la primera ocurre en un punto cercano al GAG 200, punto en donde la característica de la fractura exhibe muy poco ruido; el segunda aparece en el GAG 400, y coincide con el “cambio de curva” que se mencionó al analizar el filtrado de la señal; finalmente, el último cambio se presenta alrededor del GAG 600, y coincide con los cambios importantes en el valor del parámetro a y con la baja en el desempeño del filtro.

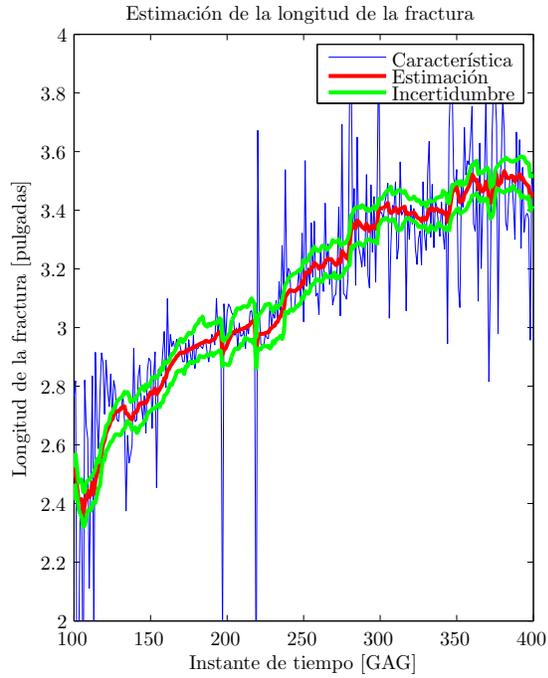


Figura 3.2: Acercamiento a la salida estimada de la estrategia de filtrado 1 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG])

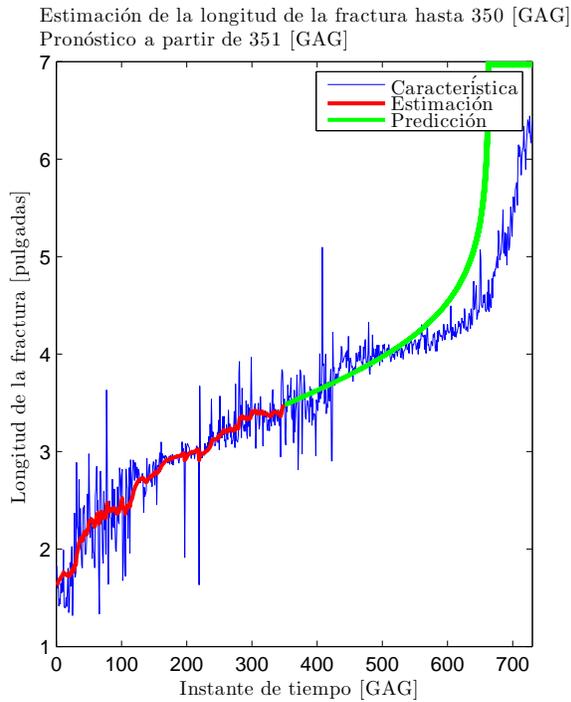


Figura 3.3: Predicciones generadas por cada una de las partículas para la estrategia de filtrado 1

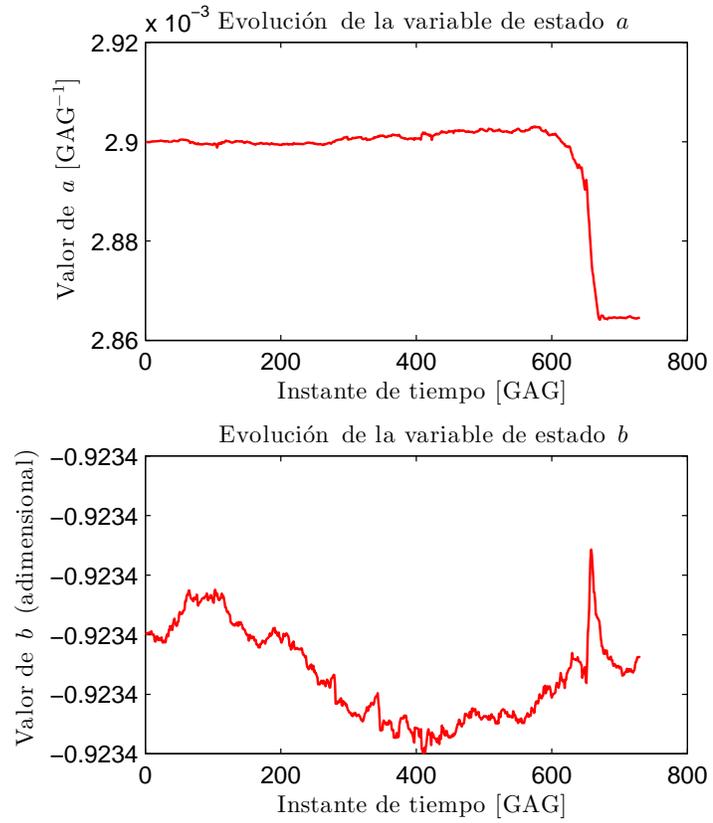


Figura 3.4: Evolución de las componentes del vector de parámetros usado en el modelo

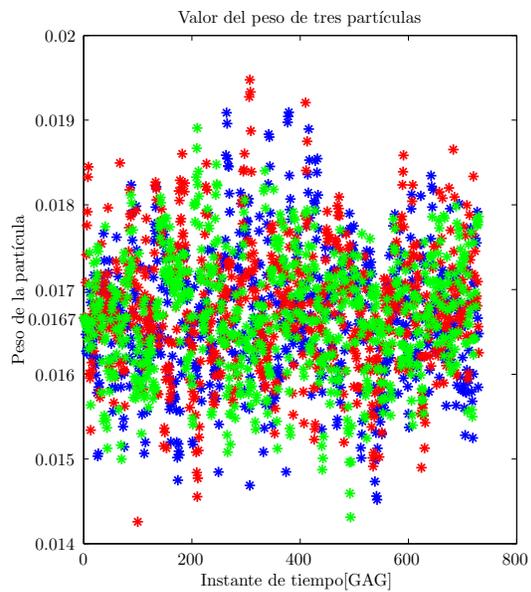


Figura 3.5: Evolución de los pesos de las partículas usadas en la estrategia de filtrado 1

Finalmente, la evolución de los pesos de tres partículas en el tiempo se expone en la Figura 3.5. Es este punto se evidencia que los pesos cambian de forma aleatoria; no se aprecia ningún patrón o tendencia en el tiempo. También se aprecia que en cada instante las partículas tienen diferente peso entre ellas. En conformidad con el funcionamiento del filtro, este comportamiento es el esperado y deseable, puesto que alguna aglomeración en torno al valor XX sería signo de degeneración¹. La inclusión de tales análisis será notoria en las siguientes secciones.

3.1.2. Filtro con proyección móvil de la trayectoria en el pasado y verosimilitud basada en la densidad chi-cuadrado

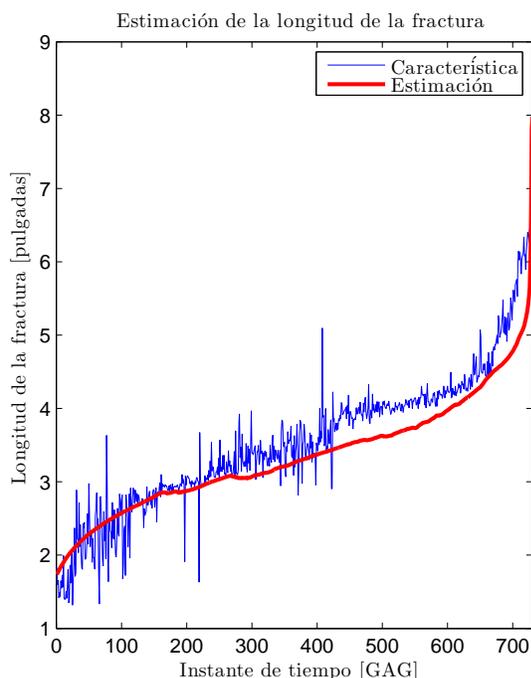


Figura 3.6: Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 2.

El filtrado de la señal ha cambiado con respecto a lo obtenido con el filtro “clásico”. Se aprecia en la Figura 3.6 que la estimación hecha a partir de la estrategia de filtrado 2 sigue una forma similar a lo que sería la trayectoria de una función arcotangente hiperbólica de parámetros constantes —aunque las variaciones con respecto a esa función aún persisten. En cuanto al desempeño como filtro, este decae con respecto a su contra par “clásico”; el algoritmo tiende a seguir más las directrices impuestas por el modelo en lugar de considerar la colección de datos de entrada. Como consecuencia se obtiene también el hecho que el filtro no se adapta al cambio del perfil de carga producido en el GAG 320, produciéndose así errores más grandes a medida que aumenta el instante de tiempo considerado, además de aumentar la incertidumbre asociada a la estimación como se observa en la Figura 3.11.

¹Recordar que $\frac{1}{60} \approx 0,0167$. Luego que las partículas se mantengan en una vecindad cercana a tal valor indica de que sus pesos asociados son similares.

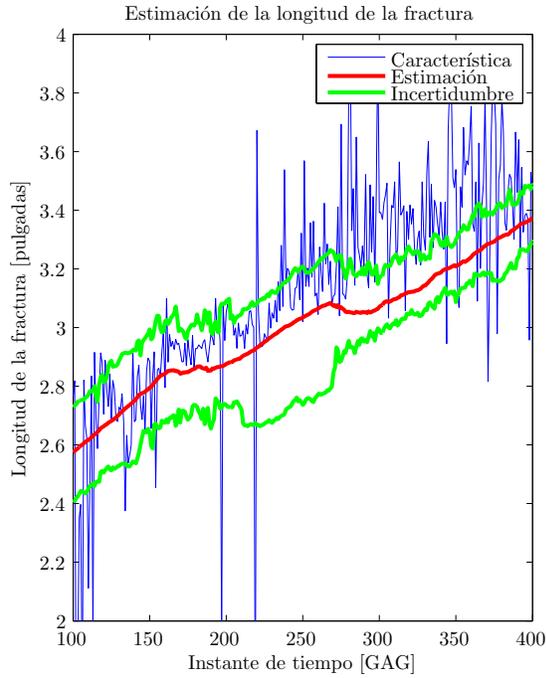


Figura 3.7: Acercamiento a la salida estimada de la estrategia de filtrado 2 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG])

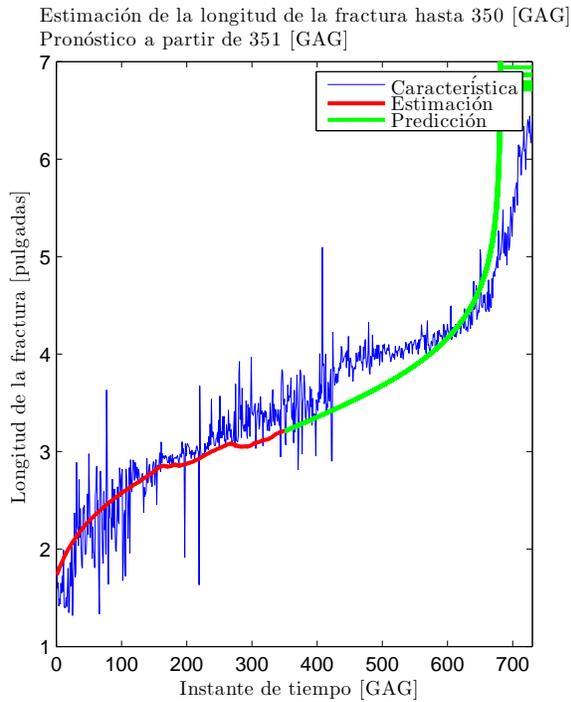


Figura 3.8: Predicciones generadas por cada una de las partículas para la estrategia de filtrado 2

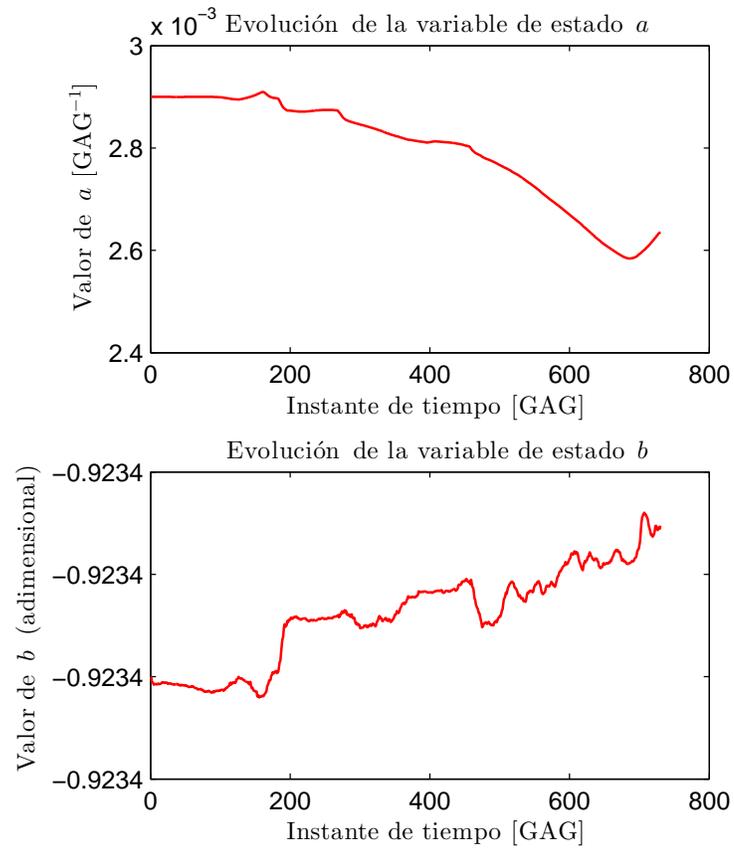


Figura 3.9: Evolución de las componentes del vector de parámetros usado en el modelo

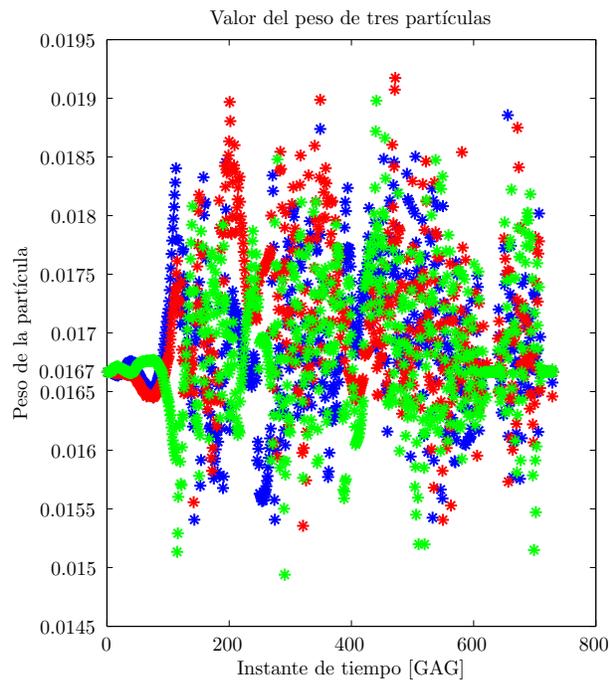


Figura 3.10: Evolución de los pesos de las partículas usadas en la estrategia de filtrado 2

El examen de la Figura 3.8 muestra una tendencia similar al caso “clásico” en cuanto al pronóstico entregado, aunque el desempeño es inferior debido a que el punto estimado desde donde se proyectan las predicciones es de inferior calidad.

El análisis de la evolución de los parámetros (ver Figura 3.9) muestra que los parámetros pierden estabilidad. La componente a presenta una evolución similar al caso “clásico”, presentando una disminución progresiva de su valor. Sin embargo, su caída total a lo largo de la estimación es de un 10 %, diferenciándose en este punto en que la baja se produce a un ritmo sostenido. El parámetro b expone también cambios mayores en cuanto a su valor inicial. Destaca el hecho que se repite el cambio de tendencia en el GAG 200. Por otra parte los otros cambios de pendiente promedio se producen alrededor de los GAG 300 y 500.

Con respecto a la evolución del peso de las partículas, es interesante observar que si bien toman valores aleatorios, durante los primeros 100 GAG se observan tendencias definidas para cada partícula en estudio. La Figura 3.10 expone que el filtro comienza su funcionamiento otorgándoles pesos iguales a todas las partículas, y las trayectorias observadas previo al GAG 100 muestran que la magnitud de los pesos varían de forma lenta en cada iteración, pero presentando una aceleración creciente. Esto muestra que el algoritmo presenta evidencia de degeneración muestral desde el comienzo de su operación, i.e., las partículas colapsan en un sólo estado en la primera iteración puesto que sólo uno de estos estados presenta una verosimilitud considerable. La explicación de este fenómeno radica en que la verosimilitud entregada por la distribución chi-cuadrado decrece asintóticamente para un grado de libertad, lo cual causa que estados que presenten una verosimilitud muy grande en relación a estados que poseen un error marginalmente mayor. Esta situación cambia conforme que aumentan las muestras en la ventana de tiempo, ya que aumentan los grados de libertad de la distribución y las verosimilitudes marginales no presentan tanta diferencia entre ellas. Esto también le da sentido a la tasa de cambio de los pesos, pues una variación muy alta en los pesos se traduce en una alta variabilidad de estados, y por consecuencia en una alta variabilidad en los errores, lo cual de acuerdo a una distribución chi-cuadrado con pocos grados de libertad es poco probable.

3.1.3. Filtro con proyección acumulada de la trayectoria en el pasado y verosimilitud basada en la densidad chi-cuadrado

Siguiendo la marca dejada por el filtro con proyección móvil de la trayectoria en el pasado, el filtrado desempeñado entrega características similares, caracterizado por una estimación muy cercana a la curva de ajuste usada, con poca influencia de los datos de entrada. La diferencia radica dos hechos: primero, en que toda noción de ajuste se pierde pasados los 500 GAG (ver Figura 3.16), lo cual demuestra un ajuste pobre al nuevo perfil de carga introducido en el GAG 320; segundo, en que el nivel de incertidumbre asociado a la estimación decrece, como se evidencia en la Figura 3.17.

La predicción, a su vez no presenta novedades con respecto a las estrategias de filtrado anteriores. Sin embargo llama la atención al comparar las Figuras 3.11y 3.13 que el pronóstico entregado sea en extremo similar a la estimación entregada como parte del filtrado.

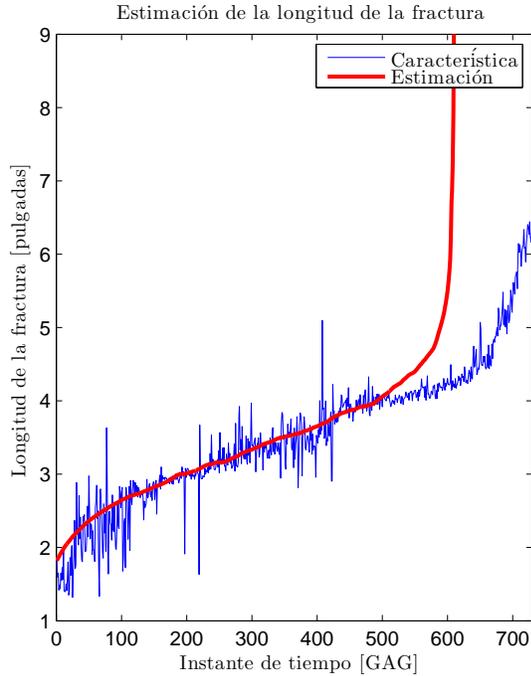


Figura 3.11: Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la de filtrado 3

Los parámetros muestran una evolución más interesante (ver Figura 3.14). Mientras que en las otras estrategias de filtrado la componente a tendía a decrecer, en esta el parámetro ofrece un aumento a medida que transcurre el tiempo a partir del GAG 200 y mantiene la tendencia a una tasa más o menos fija hasta el GAG 600; antes y después de este intervalo de crecimiento el valor de a se mantiene relativamente constante. El crecimiento obtenido fue de alrededor del 8%, en donde la intención del algoritmo era “acortar” el recorrido de la estimación. Esto explica porque la estimación de la fractura es deficiente, ya que, de acuerdo con las características de la arcotangente hiperbólica, una vez que la magnitud del argumento sobrepasa cierto valor se vuelve complicado “controlar” la tasa de crecimiento de la función.

En cuanto al parámetro b , este se presenta como más estable. Su período de mayor variabilidad se concentra en el mismo intervalo durante el cual a crece; fuera de ese rango de tiempo sus variación es muy baja. Nótese que la fijación de las componentes del vector de parámetros en instantes donde la estimación es mala muestra el poco efecto que poseen los datos de entrada sobre la salida del filtro.

Por último conviene examinar los pesos de las partículas. De acuerdo a lo expuesto en la Figura 3.15 es evidente que en los primeros GAG se produce el mismo efecto que se había visto en el filtro con proyección móvil de la trayectoria en el pasado. Pero, cabe notar que en este caso la tasa de crecimiento de la variabilidad de los pesos es mayor que en ese caso mencionado, lo cual es debido al efecto de que, al tener una ventana cuyo número de muestras siempre aumenta, los grados de libertad de la distribución chi cuadrado aumentan, con lo cual el efecto observado en el filtro con proyección móvil en el pasado —que tenía una ventana de sólo 50 muestras— se ve magnificado.

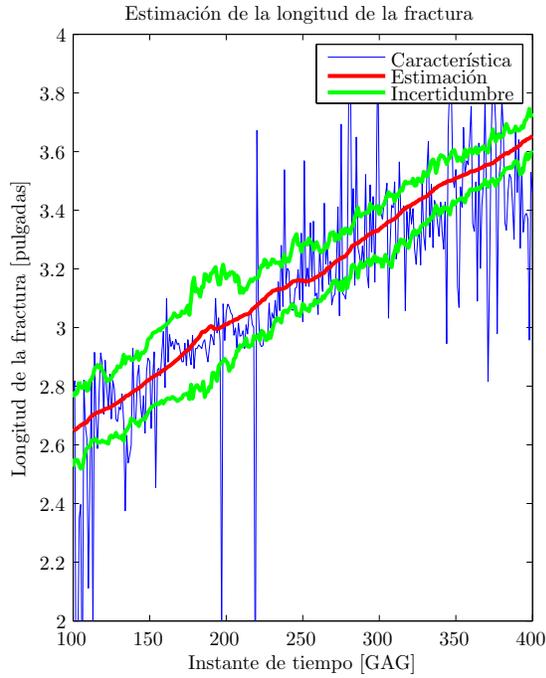


Figura 3.12: Acercamiento a la salida estimada de la estrategia de filtrado 3 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG])

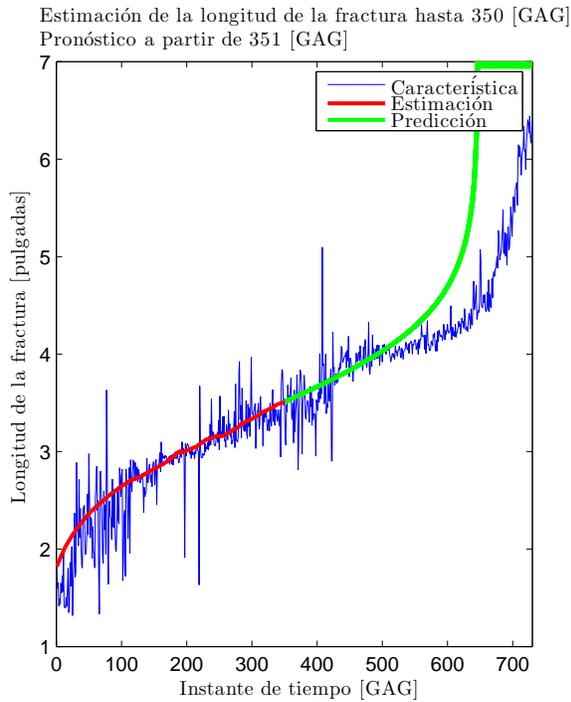


Figura 3.13: Predicciones generadas por cada una de las partículas para la estrategia de filtrado 3

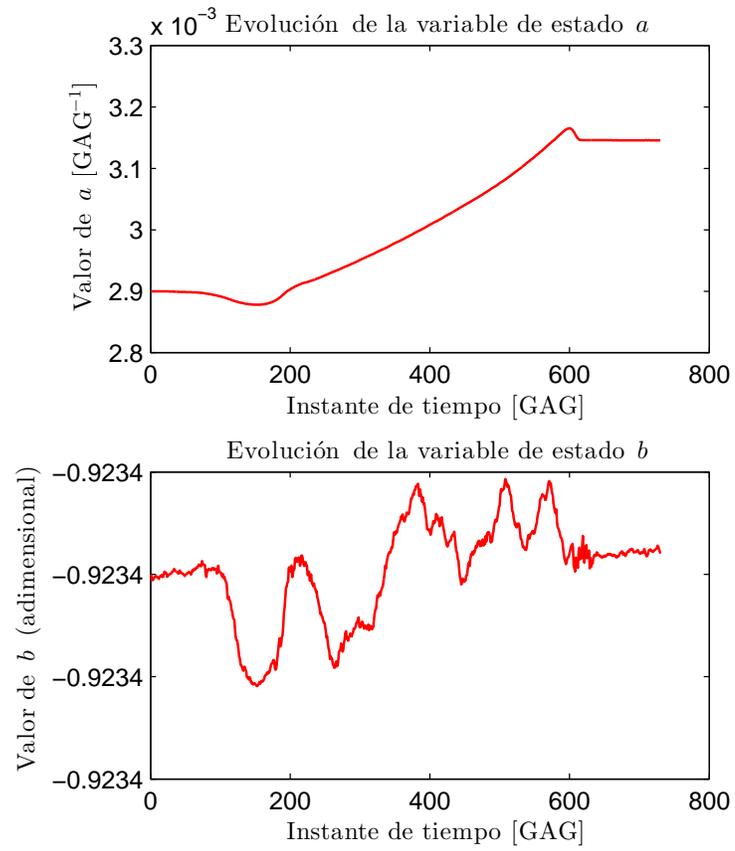


Figura 3.14: Evolución de las componentes del vector de parámetros usado en el modelo

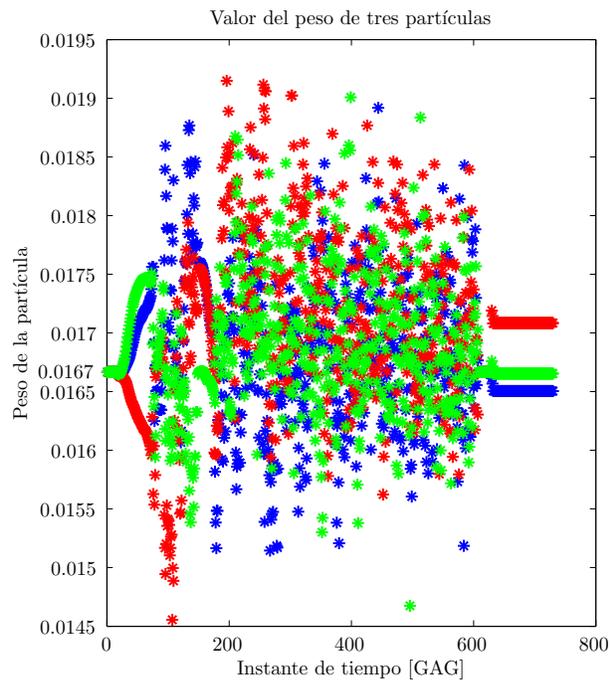


Figura 3.15: Evolución de los pesos de las partículas usadas en la estrategia de filtrado 3

Sin embargo lo más destacable de la Figura 3.15 es que pasado el GAG 600 los pesos se vuelven constantes. Esto explica porqué se pierde la variabilidad del vector de parámetros aún cuando el error de estimación —en relación a los datos de entrada— se dispara. La explicación para este fenómeno recae en la ventana de tiempo: el hecho que sea una ventana acumulada implica siempre se calcula el error entre las proyecciones de las trayectorias en el pasado y las observaciones obtenidas desde el instante inicial hasta el presente, con lo cual la colección de erres a considerar aumenta con cada iteración; esto a su vez que si el conjunto de errores es suficientemente grande, el aporte de la información nueva disminuya con cada iteración. Esto también se ve reflejado y complementado en la distribución chi-cuadrado, ya que la verosimilitud marginal se torna muy pequeña cuando los grados de libertad son mayores a cierto valor. Debido a esto, los pesos de las partículas se tornan constantes después de cierta iteración y más aún, los valores de los pesos son cercanos entre las diferentes partículas, como se parecía en la Figura 3.15. Luego es esperable que la salida del filtro no se adapte al nuevo perfil de carga, puesto que siempre va a considerar la información pertinente al perfil de carga antiguo.

3.1.4. Filtro con proyección de trayectoria en el pasado y verosimilitud basada en la CCDF de la distribución chi-cuadrado

En lo referente a este tipo de estrategias de filtrado, de la observación de las Figuras 3.16 y 3.18 se observa que a la hora de estimar el largo de la fractura la estrategia 4 ofrece un mejor desempeño que estrategia 5, específicamente entre los GAG 100 y 500. A pesar de esto, el desempeño de ambas estrategias de filtrado es similar.

Sin embargo, los problemas aparecen cuando se observa el intervalo de confianza asociado al filtrado. En las Figuras 3.17 y 3.19 se aprecia sólo unas de las curvas asociadas al intervalo de confianza, y no se distingue la señal estimada por los filtros. Esto se debe a que al hacer los gráficos, la curva de estimación se esboza antes que el intervalo de confianza, por lo que las curvas del intervalo de confianza, al ser trazadas, pueden ocultar sectores de la curva de estimación. Siguiendo este razonamiento, se tiene que, para estos filtros, la curva de estimación y sus respectivos límites del intervalos de confianza son la misma curva. Este hecho es claramente inesperado, y las razones de tal fenómeno y sus implicancias se explican a continuación.

Como se expuso en la Sección 2.2.3, el intervalo de confianza se definió como el rango entre el mayor valor de estado estimado por una partícula y el menor valor estimado para por una partícula para un instante determinado. De esta manera, el fenómeno que se observa significa que la partícula que estima el mayor valor de estado y su homóloga que estima el menor valor de estado estimaron el mismo valor para todos los GAG. Luego se tiene que los estados estimados por todas las partículas tienen el mismo valor.

Esta tesis se corrobora mediante el análisis de la Figura 3.20, donde se ve que los pesos de las partículas consideradas para la confección del gráfico son iguales en cada momento. Más aún, el valor que toman en todo instante sugiere que el peso de todas las partículas no incluidas en la Figura 3.20 también colapsa al valor que poseen las tres partículas estudiadas.

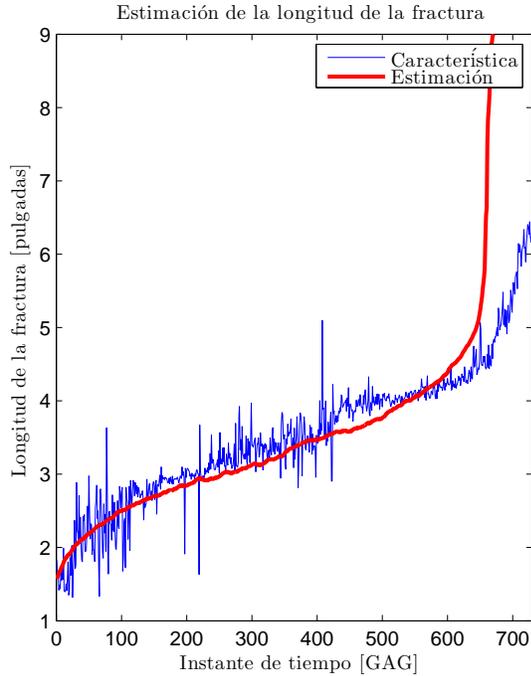


Figura 3.16: Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 4

Luego, el fenómeno observado se debe a que la muestra de partículas ha degenerado.

La degeneración presente se explica de acuerdo a la distribución usada en el cálculo de la verosimilitud. Análogamente al caso de usar la densidad chi-cuadrado, para pocos grados de libertad la verosimilitud marginal entregada por la CCDF de la chi-cuadrado es muy alta, lo cual hace que todas las partículas colapsen en un único peso. Sin embargo, el aumento en la cantidad de grados de libertad no cambia esta situación; diferencias pequeñas entre los $\varepsilon_{k-M:k}$ de diferentes partículas se traducen en grandes diferencias de verosimilitud, y por tanto se obtiene que estas partículas vuelven a colapsar en el mismo peso. Cabe mencionar que se intentó mitigar este efecto aumentando el número de partículas o usando otro grupo de valores para los ruidos de observación y proceso, pero no se tuvo éxito.

Como se ha visto, las estrategias de filtrado 4 y 5 presentan una degeneración muestral evidente y sostenida en el tiempo. Esto trae como consecuencia de que el intervalo de confianza es nulo, como se mostró en las Figuras 3.17 y 3.19, lo que implica que la estimación no es confiable. Más aún, dado que se cuenta con una sola muestra estimada, no es posible aproximarse a la distribución que gobierna el proceso —que es el propósito del filtro de partículas—, lo cual corrobora la noción de que las estrategias de filtrado 4 y 5 no son viables. Debido a esto, se ha decidido excluir estas estrategias de filtrado del resto de los análisis que componen este trabajo.

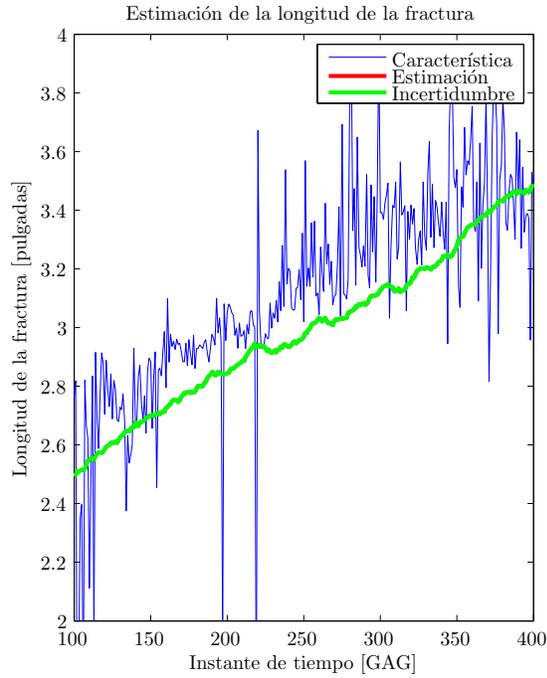


Figura 3.17: Acercamiento a la salida estimada de la estrategia de filtrado 4 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG])

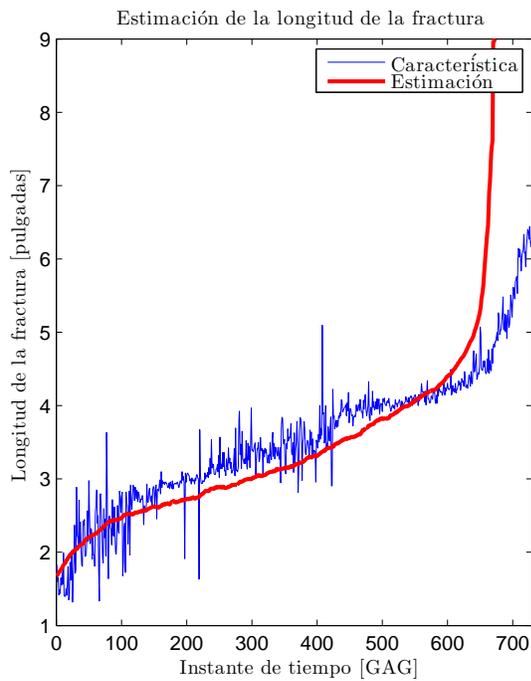


Figura 3.18: Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 5

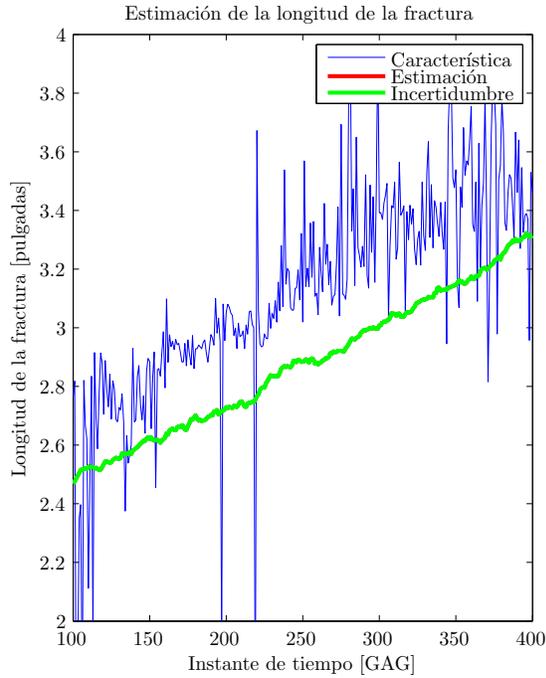


Figura 3.19: Acercamiento a la salida estimada de la estrategia de filtrado 5 y su intervalo de confianza (rango del acercamiento: [100,400] [GAG])

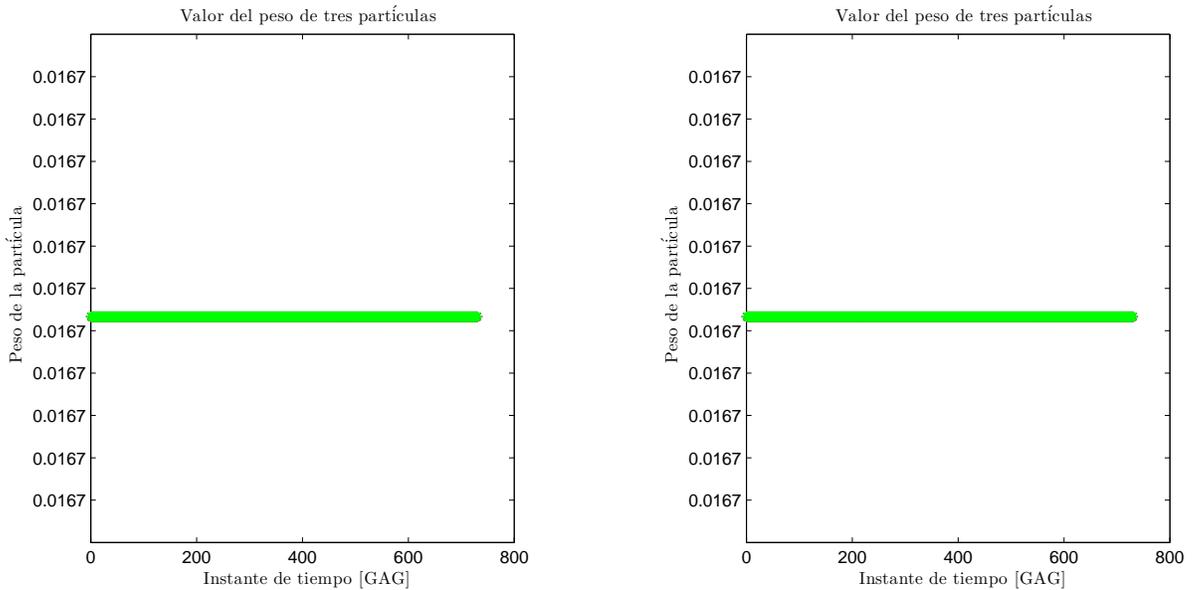


Figura 3.20: Evolución de los pesos de las partículas usadas por el filtro. Izquierda: Estrategia de filtrado 4. Derecha: Estrategia de filtrado 5

3.2. Análisis de los efectos de las varianzas de los ruidos y resultados numéricos

La totalidad de los datos experimentales obtenidos se encuentran en el Anexo A. Del estudio de estos datos se observa influencia que tienen la varianza de los ruidos involucrados en el algoritmo tanto en el proceso de filtrado como en el de pronóstico.

3.2.1. Desempeño de las estrategias de filtrado de partículas implementadas en labores de estimación

El estudio de los datos presentes en el Anexo A.1 permite mostrar la variabilidad en rendimiento fruto de la modificación de los valores de las varianzas asociadas dentro de una vecindad definida. De esta forma, se aprecia que la influencia producto de la mutación de la varianza del ruido de proceso es transversal a todas las estrategias de filtrado examinadas. De acuerdo a la Tabla A.1, se tiene que al aumentar σ_ω^2 se obtiene una mejora en la estimación producida por el filtro y se mejora la adaptación del mismo a los datos de entrada, lo que se aprecia en el bajo error que se obtiene al usar una varianza de 0.01 para la evaluación del período completo. Cabe destacar que, de los experimentos ejecutados, este fue el único donde el ECM extraído de la estimación completa de la característica es menor al obtenido en el período acotado.

Tales beneficios asociadas al incremento de σ_ω^2 tienen relación con el hecho de que, a mayor valor de esta varianza, mayor es la diversidad del conjunto de partículas generado durante cada iteración del algoritmo. Esto es útil, puesto que una muestra enriquecida tiende a tener mayores posibilidades de ajustarse la distribución condicional que gobierna la evolución de la característica, y por tanto, ofrece ventajas a la hora de ajustarse a los datos de entrada del filtro.

La Tabla A.2 exhibe el hecho que $\sigma_{\xi_a}^2$ posee una influencia similar a la desplegada por σ_ω^2 . Si bien esta influencia no es muy clara en el período A, se hace flagrante en B, lo que sugiere una mejora considerable en las capacidad de adaptación del filtro. Esto guarda relación con la capacidad de a de acrecentar el dominio de la curva de ajuste, lo cual ayuda a controlar la tasa de crecimiento de la arcotangente hiperbólica, y por tanto, mejora la respuesta ante la baja en el perfil de carga experimentada. Análogamente, el efecto queda claro al recordar lo exhibido en la Figura 3.4, donde al observar la evolución de a durante los últimos GAG se cae en cuenta que es el beneficioso poseedor de una mayor variabilidad, y por consecuencia, de una mayor tasa de cambio.

En cuanto a lo expuesto en la Tabla A.3, las mutaciones a $\sigma_{\xi_b}^2$ producen pocos efectos en la salida del filtro. La naturaleza de las consecuencias de tales variaciones no es apreciable en los datos recogidos, y dada la magnitud de las diferencias presentadas, se infiere que lo expuesto sea producto de la aleatoriedad del algoritmo. Esto es coherente con lo expuesto en la Figura 3.4, donde se aprecia que el parámetro b varía en torno a una vecindad acotada.

El peso que posee la varianza del ruido de observación se aprecia en la Tabla A.4. Queda

expuesta el hecho que a menor σ_ν^2 menor es el ECM. Este efecto se presenta en los dos períodos analizados, aunque el efecto destaca más en el intervalo extenso de estudio, lo que se toma como una muestra de la mejora en la adaptabilidad de la estrategia de filtrado 1. Tales atributos se entienden en base a que una menor varianza del ruido de observación aumenta la influencia de los datos de entrada en la respuesta del filtro, lo cual contribuye al ajuste de la señal filtrada a la característica utilizada.

En general, se aprecia que las reacciones de las estrategias de filtrado 2 y 3 ante la variación de la componente aleatoria de los ruidos presentes es similar. Por una parte los efectos producidos al cambiar la varianza del ruido de proceso o la varianza del ruido asociado a la componente a del vector de parámetros son opuestos a lo apreciado en la estrategia de filtrado 1. De acuerdo a las Tablas A.5 y A.9, se evidencia que el error cuadrático medio aumenta a medida que crece la varianza del ruido de proceso. Se advierte además que la tasa de mejoría decrece al disminuir la varianza en cuestión. De manera similar el aumento en $\sigma_{\xi_a}^2$ se traduce en un peor rendimiento del filtro en el tramo A para la estrategia de filtrado 2 como queda claro en la Tabla A.6, y en ambos tramos para la estrategia 3 (ver Tabla A.10). Además, cabe resaltar que los errores obtenidos con estos algoritmos son mayores que los obtenidos para el estrategia 1, especialmente al tomar en cuenta el período completo de estimación —lo cual es en extremo patente en el error producido por el estrategia 3.

Sin embargo, destaca la diferencia que se produce en relación a los cambios en $\sigma_{\xi_b}^2$ y σ_ν^2 . En cuanto a la estrategia de filtrado 3, la naturaleza de las observaciones obtenidas a través de las Tablas A.11 y A.12 es tal que no permite dilucidar alguna tendencia, y por tanto se considera que las pruebas efectuadas no son concluyentes. Sin embargo, en relación a lo percatado en la Tabla A.7 se desprende que $\sigma_{\xi_b}^2$ tiene efectos en rendimiento estimativo del filtro, el cual es más notorio en el período más extenso de estudio. También, de acuerdo con la Tabla A.8 se observa una leve tendencia a la disminución del ECM asociada al aumento de σ_ν^2 .

3.2.2. Desempeño de las estrategias de filtrado de partículas implementadas en labores de predicción

Una de las cosas principales que se distingue es que el error de predicción disminuye a medida que aumenta el horizonte de estimación. Esto se aprecia en todas las tablas exhibidas en el Anexo A.2 exceptuando la Tabla A.17, de donde se nota que la magnitud del error de predicción usando la estrategia de filtrado 2 para $\sigma_\omega^2 = 0,01$ es mayor tomando el GAG 600 como horizonte de estimación en lugar del GAG 90. En este aspecto, cabe resaltar que la estrategia 2 es la que presenta mayores errores para el uso del GAG 600 como horizonte de estimación, presentando diferencias de magnitud considerables con respecto al desempeño de las otras estrategias bajo el mismo horizonte.

En el caso de la estrategia de filtrado 1, se observa que la variabilidad de los ruidos tiene mayor efecto considerando el GAG 90 como horizonte de predicción. Esto se corrobora por medio de la Tabla A.13, donde se aprecia una relación directa entre los aumentos de σ_ω^2 y el error, o por medio de las Tablas A.14, A.15 y A.16, donde se aprecia una relación inversa

entre los aumentos de la varianza del vector de parámetros o del ruido de observación y el error.

Como era de esperar, las contribuciones generadas por los ruidos asociados a a y ω tienen consecuencias similares en las estrategias de filtrado 2 y 3. Considerando el GAG 600 como horizonte de estimación, aumentos en σ_ω^2 se traducen en mayores error de pronóstico. De la misma forma, tomando el GAG 90 esta vez como horizonte, crecimientos en $\sigma_{\xi_a}^2$ repercuten disminuyendo el error de predicción. Sobre las mismas varianzas juzgadas bajo los horizontes de estimación no comentados, no es posible esbozar conclusión alguna puesto que el punto medio de ambos experimentos corresponde al error máximo —en el caso de la estrategia 2— y mínimo —en el caso de la estrategia 3— registrados, lo que se cumple tanto para σ_ω^2 como para $\sigma_{\xi_a}^2$.

Siguiendo con el análisis del desempeño de las estrategias 2 y 3, de forma análoga al caso del filtrado, una de las diferencias notables guarda relación con lo aportado por $\sigma_{\xi_b}^2$, donde, para la estrategia de filtrado 2, se aprecia en una relación directa entre el aumento de esta varianza y el aumento del error (ver Tabla A.19). Sin embargo, no se puede concluir nada de el efecto producido por $\sigma_{\xi_b}^2$ sobre la estrategia 3 a partir de lo observado en la Tabla A.23.

Algo llamativo de los resultados de predicción es la presencia de un término infinito en la Tabla A.24. Tal término guarda relación con estabilización en la evolución de los pesos presentada en la Figura 3.15. Luego, la aparición de tal valor puede tomarse como la aparición de características degenerativas fruto del aumento en la varianza del ruido de observación. Esto tiene sentido puesto que tal crecimiento provocaría una disminución en la influencia que poseen los datos de entrada del filtro, como consecuencia de la normalización asociada al proceso de actualización; y por consiguiente, la poca influencia combinada con la gran presencia de datos pasados hace que se acelere el proceso de “estabilización” de los pesos.

3.2.3. Detección de fallas en base a parámetros

La información presentada en las Figuras 3.21 a la 3.44 corresponde a la evolución de las componentes del vector de parámetros en un espacio acotado de tiempo. La exposición de esto se hace con el fin de analizar el comportamiento de ajuste de estas variables y, en lo posible, reconocer en tales esfuerzos de ajuste detalles que puedan sugerir la presencia de anomalías, y por tanto, desemboque en una esfuerzo de detección. Cabe recordar que para las figuras mencionadas, los parámetros son los mismos descritos en la Tabla 3.1; en la Figura se enfatiza la varianza que ha sido modificada.

A partir de las las Figura 3.21 se desprende que no hay relación alguna entre la capacidad de detección asociada al parámetro a y la varianza del ruido de proceso. Nótese que sólo en el gráfico asociado a $\sigma_\omega^2 = 1 \times 10^{-4}$ ocurren cambios interesantes en la vecindad de la anomalía conocida (la cual ocurre en GAG 320). Sin embargo, los cambios de tendencias posteriores alrededor del GAG 380 ponen en duda la posibilidad de usar el parámetro a , bajo estas condiciones, como detector de anomalías. Cabe mencionar que también existe un cambio de tendencia llamativo en el gráfico superior, alrededor de los 400 GAG, pero tal hito se considera tardío como para ser considerado como una respuesta a una falla.

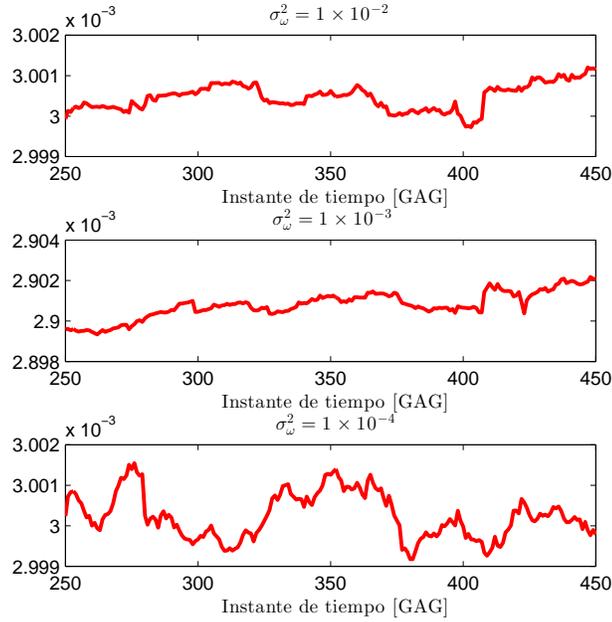


Figura 3.21: Evolución del parámetro a para diversos ruidos de proceso usando la estrategia de filtrado 1

Las evoluciones descritas en la Figura 3.22 poseen características similares a lo expuesto para σ_ω^2 , aunque de sentido contrario, dado que es ahora la varianza menor la que incita interés. Como se aprecia en la imagen superior, previo al GAG 350 comienza una tendencia ascendente que se mantiene hasta un poco después del GAG 400. La tasa de la trayectoria descrita se podría describir como relativamente constante lo cual se puede asociar a una respuesta definida y por tanto podría tomarse como una detección. En otro ámbito, de nuevo llama la atención la presencia de hito alrededor de GAG 400 para $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ (imagen inferior); sin embargo, como ya se mencionó, tal respuesta se aprecia como tardía, y por tanto no será considerada como de interés.

La influencia de $\sigma_{\xi_b}^2$ se observa a través de la Figura 3.23. Como se puede ver, no se distingue relación alguna entre las variaciones y la formación de algún patrón de detección, como tampoco se revelan detalles significativos para los parámetros probados. Esta baja trascendencia se correlaciona mucho con los resultados de estimación y predicción descritos en las Secciones 3.2.1 y 3.2.2, afirmando el concepto de la baja repercusión de esta componente al desempeño de la estrategia de filtrado 1.

En cuanto al ruido de observación, resulta interesante observar como en la imagen inferior de la Figura 3.28 se distingue un cambio de tendencia definido y sostenido, con lo cual la varianza $\sigma_\nu^2 = 0,5$ se plantea como otra candidata para respuestas detectoras. Aparte de esto, no aparece ningún indicio de que exista una relación más profunda entre la variación de la varianza mencionada y las capacidades detectoras del filtro. Por último se observa también la existencia de un cambio de tendencia leve alrededor del GAG 400 para $\sigma_\nu^2 = 0,45$.

El análisis ejecutado para la estrategia 1 es repetido para la estrategia 2. De acuerdo con

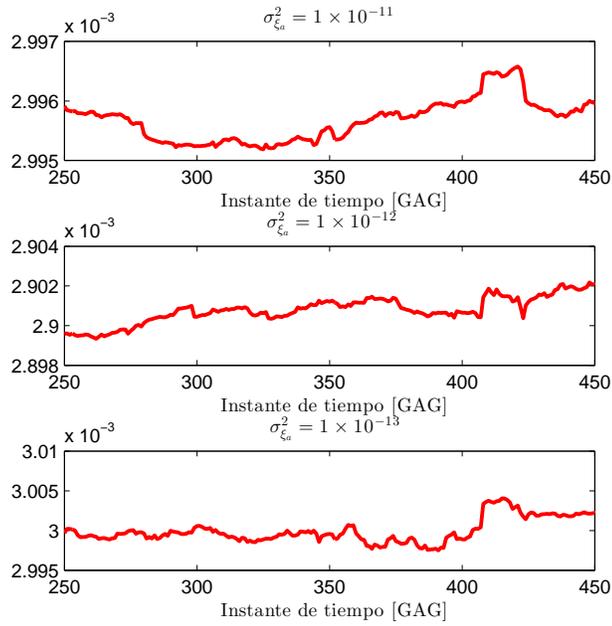


Figura 3.22: Evolución del parámetro a para diversos ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 1

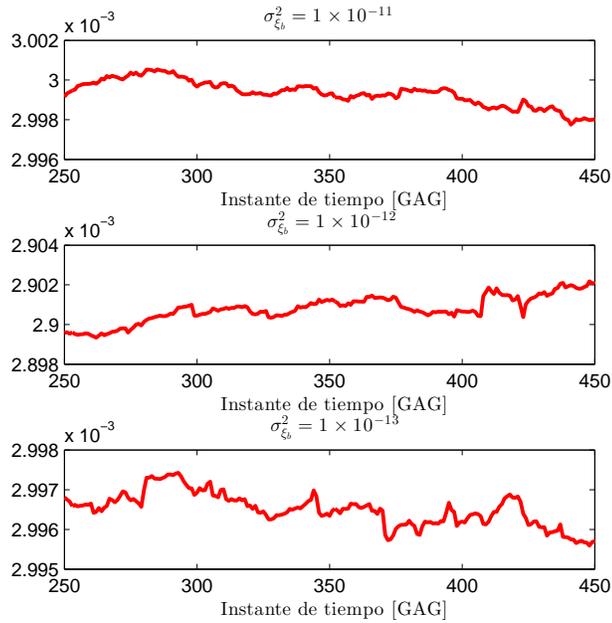


Figura 3.23: Evolución del parámetro a para diversos ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 1

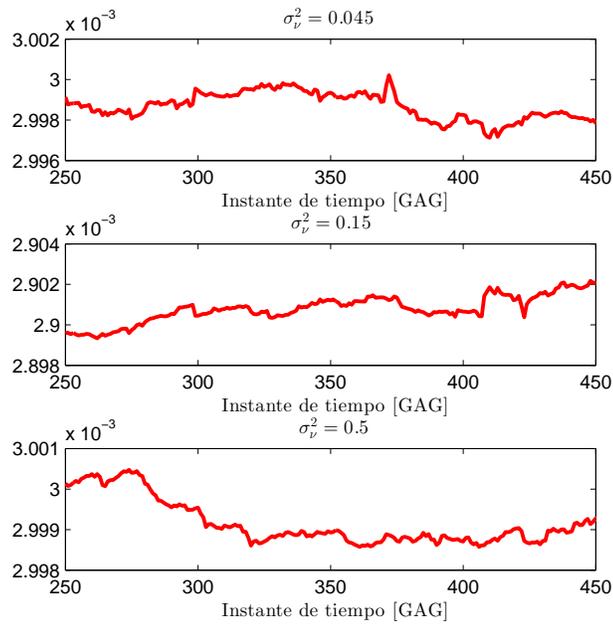


Figura 3.24: Evolución del parámetro a para diversos ruidos de observación usando la estrategia de filtrado 1

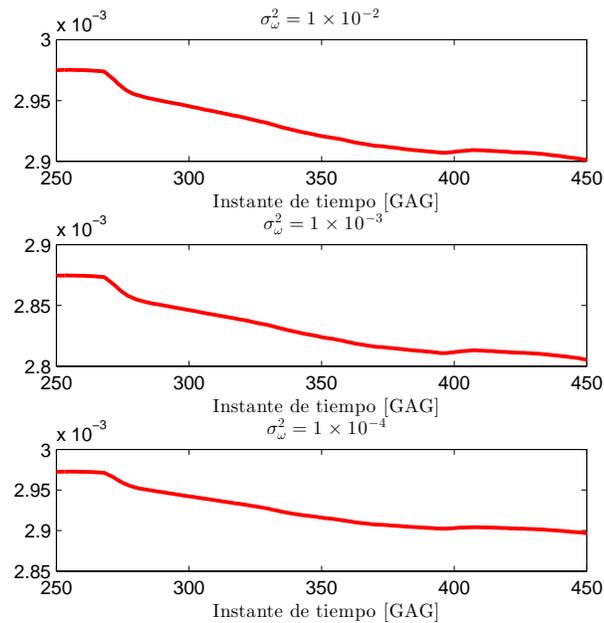


Figura 3.25: Evolución del parámetro a para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 2

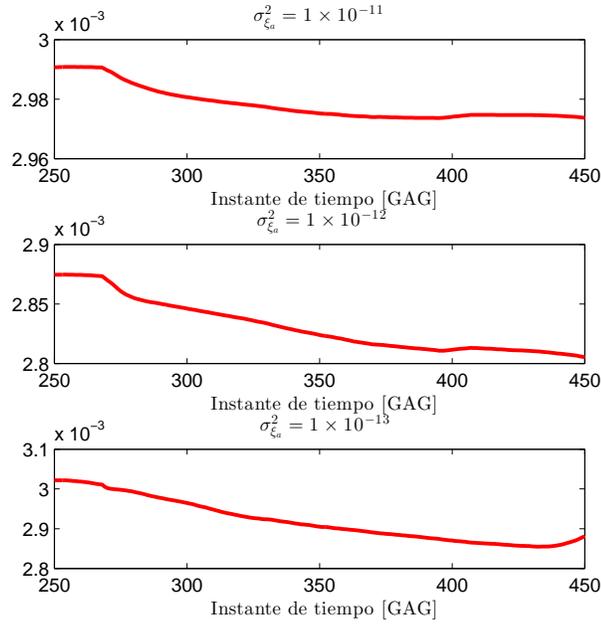


Figura 3.26: Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 2

las Figuras 3.29, 3.30 y 3.31, las varianzas asociadas al ruido de observación o al vector de parámetros no contribuyen en forma alguna al desarrollo de testimonio para la detección del cambio de perfil presente en la característica usada en los estudios. En cuando al efecto de la varianza del ruido de observación, sobresale el gráfico superior de la Figura 3.32, el cual deja revelar un cambio de tendencia importante asociado a $\sigma_v^2 = 0,045$. Esta alteración mantiene su carácter inconfundible durante más de 50 GAG, por lo que se plantea como modificación candidata a la hora de reforzar las capacidades detectoras de la estrategia de filtrado 2.

La situación expuesta por el examen a la estrategia de filtrado 2 se repite para estrategia 3: de lo manifestado por las Figuras 3.29, 3.30 y 3.31 se extrae que la variabilidad de la componente aleatoria asociada a la ecuación de estado o al vector de parámetros no tiene peso sobre la evolución del parámetro a . En relación a la componente aleatoria de la ecuación de observación, la Figura 3.32 enseña que si bien esta variable aporta de forma notoria, como se atestigua en el gráfico superior de la figura, su contribución no avanza en el sentido de mejorar las capacidades detectoras del filtro. De esta forma se tiene que, en ninguno de los casos de modificación estudiados se observa la presencia de algún hito cercano o posterior al GAG 320. Más aún, destaca dentro de los casos analizados que la inclinación evolutiva del parámetro a se mantiene inmutable.

Como manera de abordar la evolución del parámetro b y su aporte a las labores de detección de anomalías, en las Figuras 3.33 a la 3.44 se presenta el desarrollo de la variable en el tiempo.

Comenzando por la estrategia de filtrado 1, las Figuras 3.33 y 3.34 resaltan que tanto σ_v^2 como $\sigma_{\xi_a}^2$ no contribuyen a la detección del cambio en el perfil de carga; de hecho, los cambios en la trayectoria de las partículas se aprecian más cerca del GAG 400, de forma

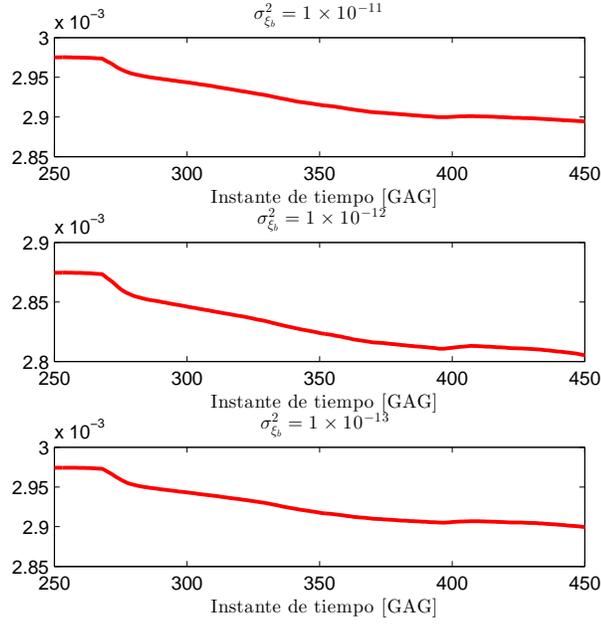


Figura 3.27: Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 2

similar a lo encontrado para la evolución de a bajo el funcionamiento de este filtro. De esta forma, sólo en para las varianzas σ_ν^2 y $\sigma_{\xi_b}^2$ se perciben hitos que podrían considerarse útiles en términos de detección, los cuales están asociados a los valores $\sigma_\nu^2 = 0,5$ y $\sigma_{\xi_b}^2 = 1 \times 10^{-13}$ como se aprecia en las 3.35 y 3.36. Cabe destacar en que las propiedades detectoras asociadas a la componente b fruto de las varianzas mencionadas son de carácter bastante débil, y por lo mismo no constituyen buenas candidatas a la hora de diseñar un sistema de detección en torno a estas.

Al analizar el progreso de b bajo la operación del estrategia de filtrado 2 se distinguen diversas propiedades. Primero, destaca que para la colección de datos de control ya se perciben características deseables para la detección del cambio de perfil. Se aprecia un cambio en el desarrollo de la componente muy definido previo al GAG 350, el cual se puede ver en la Figura 3.37 por ejemplo. En la misma figura destaca que un aumento en el valor de σ_ω^2 produce cambios más evidentes, lo cual sugiere una relación entre la magnitud de la varianza y la capacidad detectora del algoritmo.

En lo referente al vector de parámetros, se aprecia también características deseables para $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ y $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$, como se observa en las Figuras 3.38 y 3.39 respectivamente. En el primer caso se aprecia un cambio en al trayectoria de la partícula alrededor del GAG 350, donde este recorrido se estabiliza cerca del GAG 400. La variante en la evolución de b incurrida en el caso 2 guarda relación con una baja abrupta en la tasa de descenso de la variable alrededor del GAG 350, luego de mantener una pendiente relativamente constante hasta poco antes del GAG 450. Dado que estas características son definidas y carecen de una componente aleatoria considerable, se tiene que funcionan como herramientas de detección visual de anomalías. Nótese que, de acuerdo a lo observado, es posible inferir relaciones entre

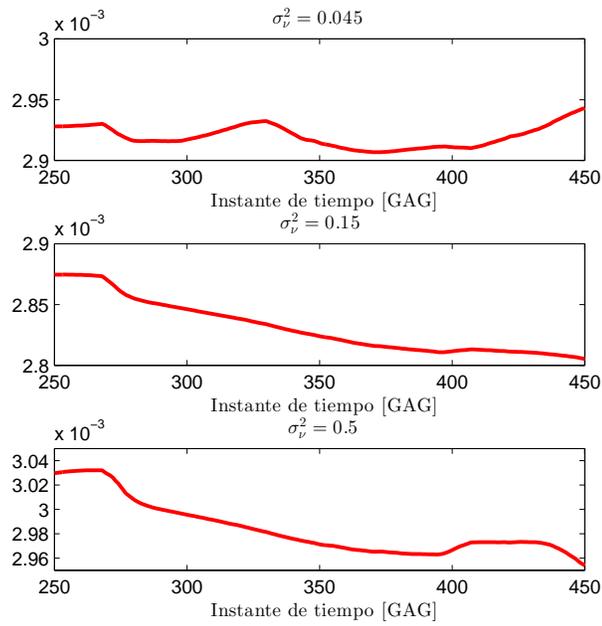


Figura 3.28: Evolución del parámetro a para diversas varianzas de ruidos de observación usando la estrategia de filtrado 2

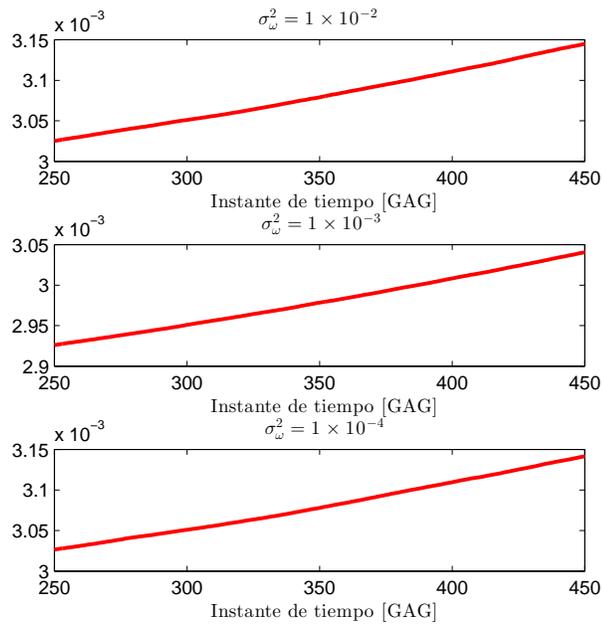


Figura 3.29: Evolución del parámetro a para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 3

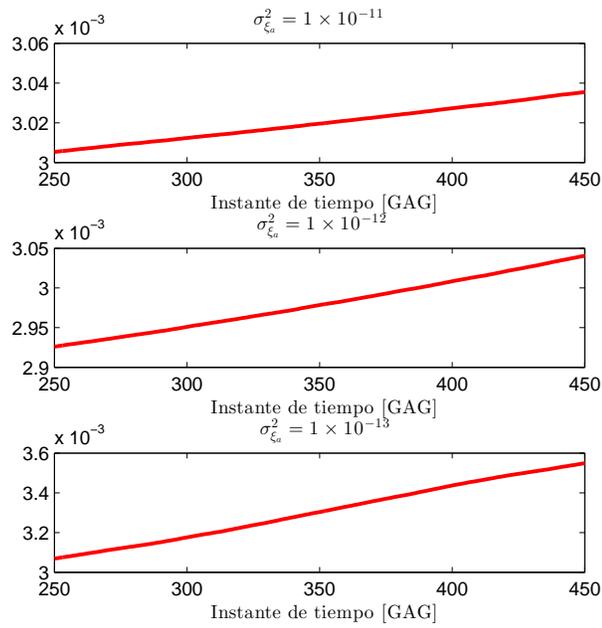


Figura 3.30: Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 3

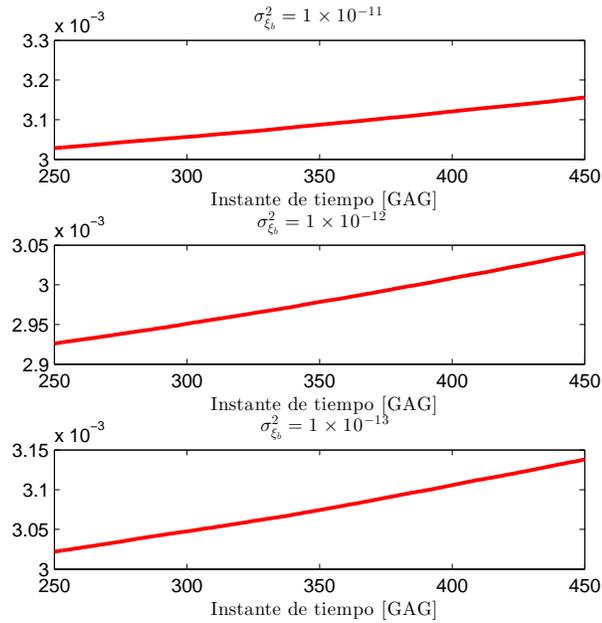


Figura 3.31: Evolución del parámetro a para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 3

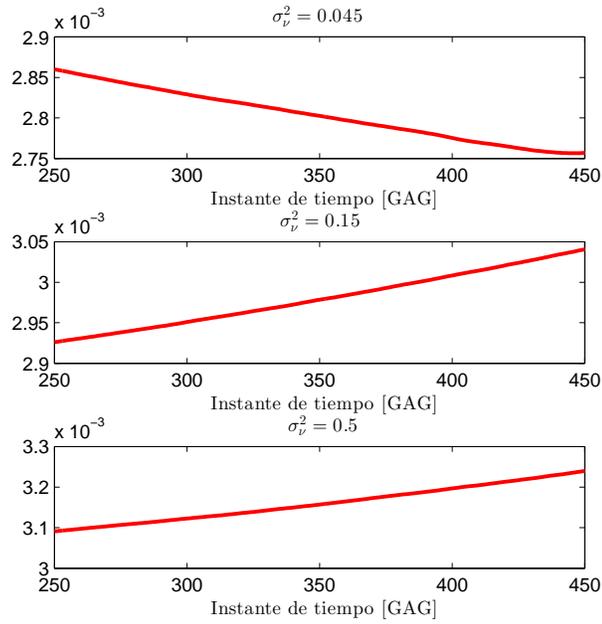


Figura 3.32: Evolución del parámetro a para diversas varianzas de ruidos de observación usando la estrategia de filtrado 3

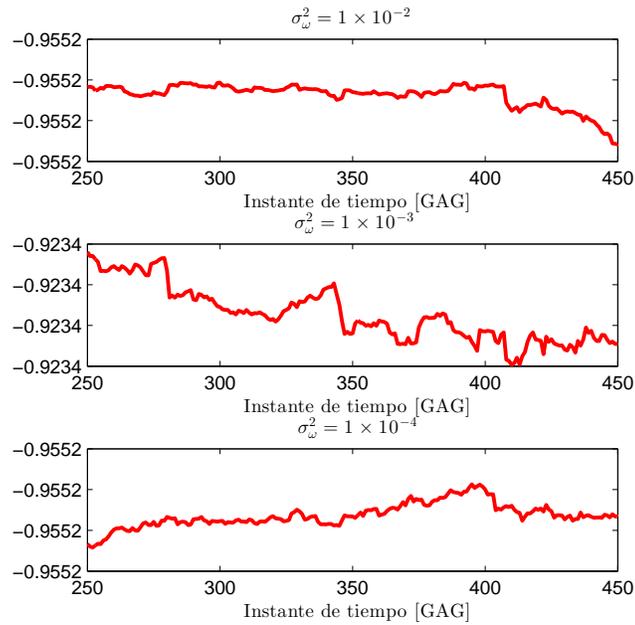


Figura 3.33: Evolución del parámetro b para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 1

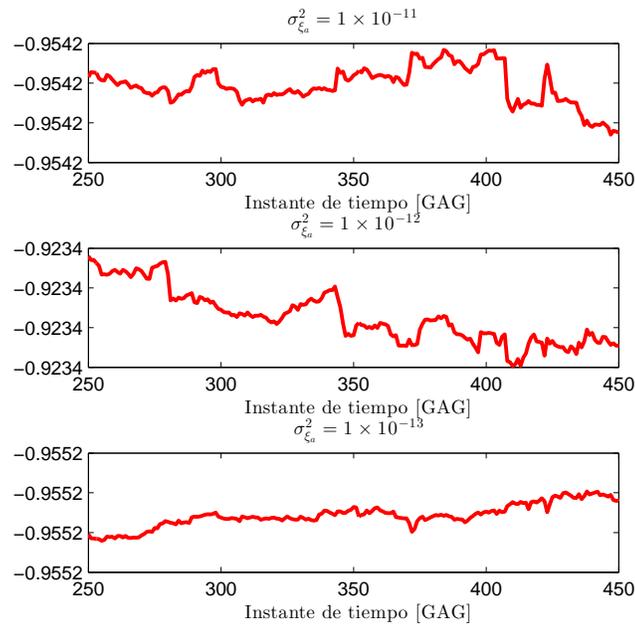


Figura 3.34: Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 1

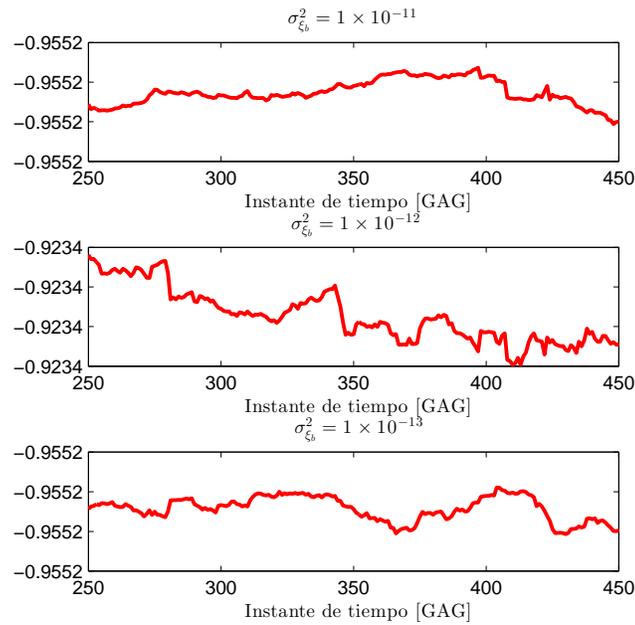


Figura 3.35: Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 1

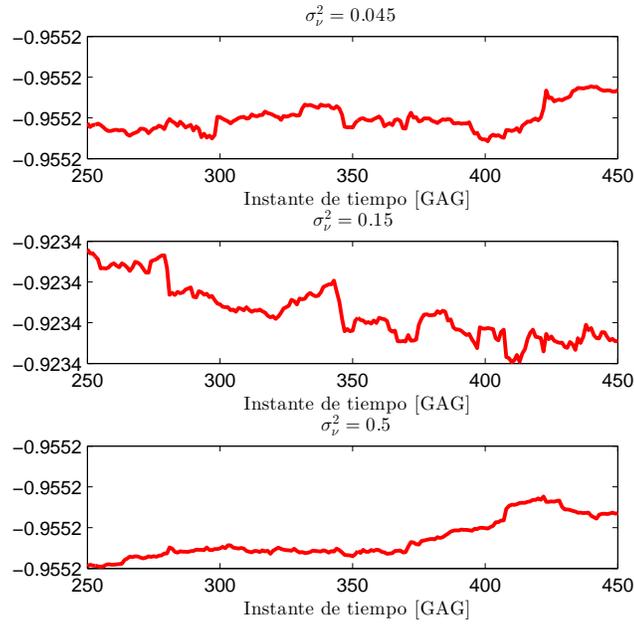


Figura 3.36: Evolución del parámetro b para diversas varianzas de ruidos de observación usando la estrategia de filtrado 1

la la magnitud y la aparición de características indicadoras, tales como como que a menor varianza del ruido de a o mayor varianza del ruido de b se tiene acceso a facilidades de detección visual.

En relación con la varianza del ruido de observación, se observan ciertos hitos en base a los cuales es posible efectuar diagnosticar una anomalía para $\sigma_v^2 = 0,045$. Téngase en cuenta que es posible detectar la anomalía de forma más directa haciendo uso de la varianza de control, por lo que tal σ_v^2 queda relegada a segunda opción.

Finalmente, el análisis realizado para la estrategia de filtrado 2 es idéntico al considerar estrategia 3. Se aprecia de forma análoga que la realización de control ofrece hitos de buena calidad que pueden ser usados como mecanismos de diagnóstico de anomalías. Esto puede ser observado en cualquiera de las Figuras 3.41 a la 3.44.

A partir de la Figura 3.41 se observan cambios en las tendencias para todas las varianzas mostradas. Se advierte también que existe cierta relación directa entre la magnitud de σ_v^2 y que tan tarde ocurre el cambio en el progreso de la partícula. Es importante que, aún así, ninguna de las modificaciones efectuadas a la varianza control facilitan la inspección visual de anomalías con respecto a la control.

En la Figura 3.42 se considera que en todos los gráficos es posible percibir el cambio de perfil de carga mecánica en base al desarrollo de b . En este caso, las modificaciones consideradas exponen con mayor claridad la anomalía conocida. La diferencia radica en la pendiente que deja en evidencia a la falla: en ambos casos, previo a la mutación, se tiene una tendencia relativamente constante; luego, la pendiente se hace decisivamente negativa

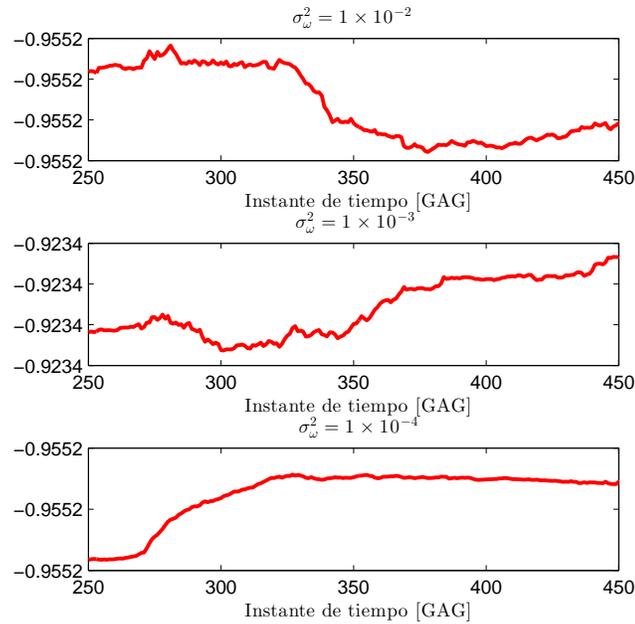


Figura 3.37: Evolución del parámetro b para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 2

—para $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$ — o positiva —para $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$. Destaca el hecho que el cambio de tendencia más suave es el visto en la curva de control, lo cual sugiere que podría ser una transición entre los otros dos esquemas de diagnóstico.

Para el caso de la varianza asociada a la componente b , se distingue en a Figura 3.43 el hecho que sólo la varianza $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$ provee características detectoras. Se aprecia además que la pendiente que evidencia la falla es de signo contrario al caso de control, dando un efecto similar al experimento asociado a $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$.

Finalmente, de la Figura 3.44 se desprende que en $\sigma_\nu^2 = 0,5$ es evidente el cambio de perfil de carga, destacado por un cambio de pendiente muy cerca del lugar donde ocurre la anomalía. Nótese además que la modificación en la evolución de b es de mayor magnitud que en el caso de control, lo que sugiere que una mayor confianza en el modelo en desmedro de los datos mejora el desempeño del filtro como herramienta detectora.

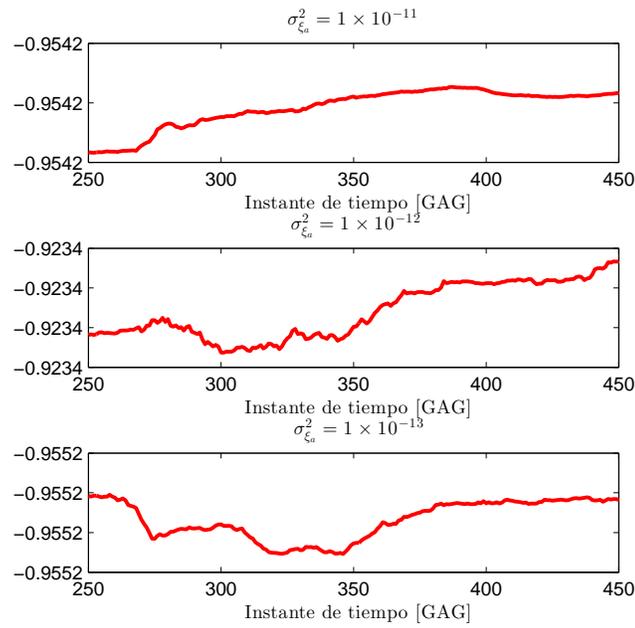


Figura 3.38: Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 2

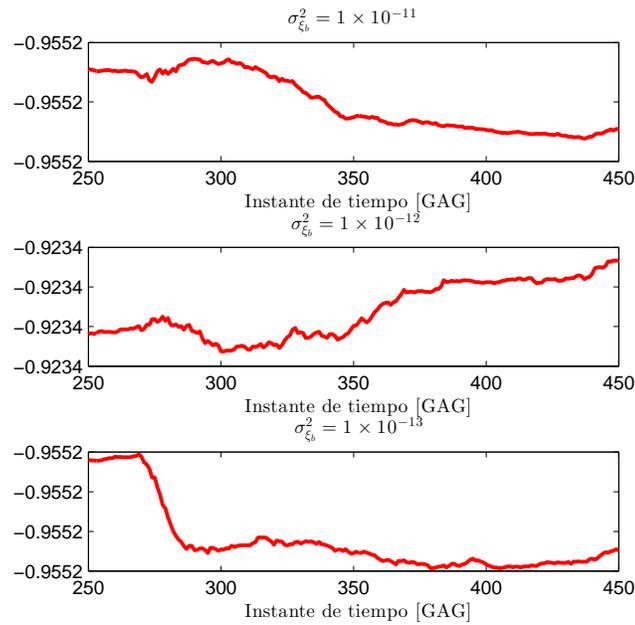


Figura 3.39: Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 2

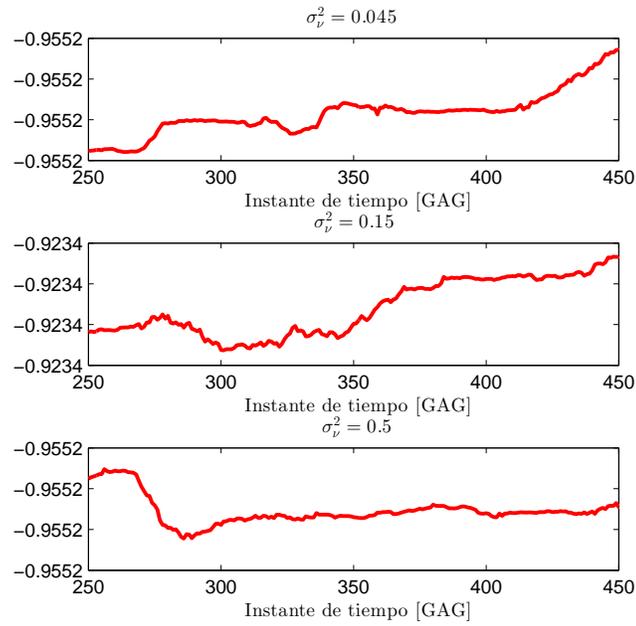


Figura 3.40: Evolución del parámetro b para diversas varianzas de ruidos de observación usando la estrategia de filtrado 2

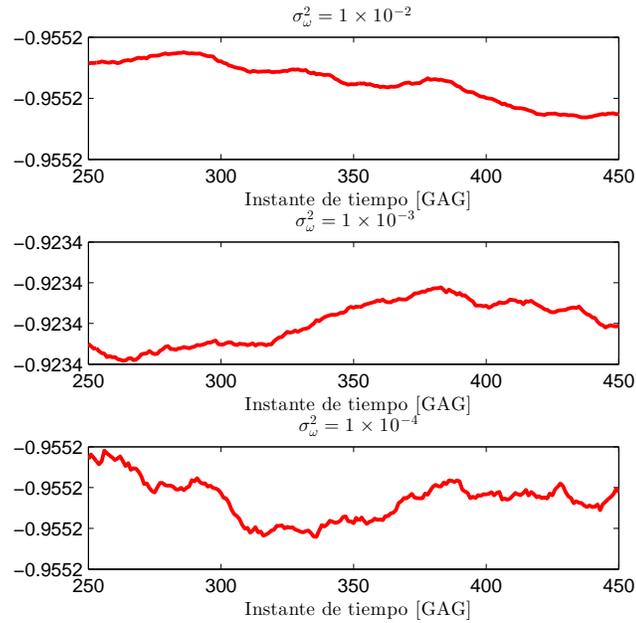


Figura 3.41: Evolución del parámetro b para diversas varianzas de ruidos de proceso usando la estrategia de filtrado 3

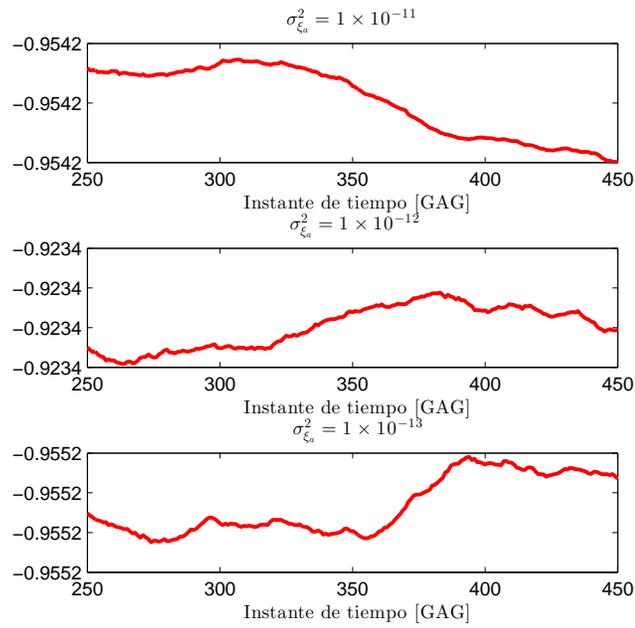


Figura 3.42: Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro a usando la estrategia de filtrado 3

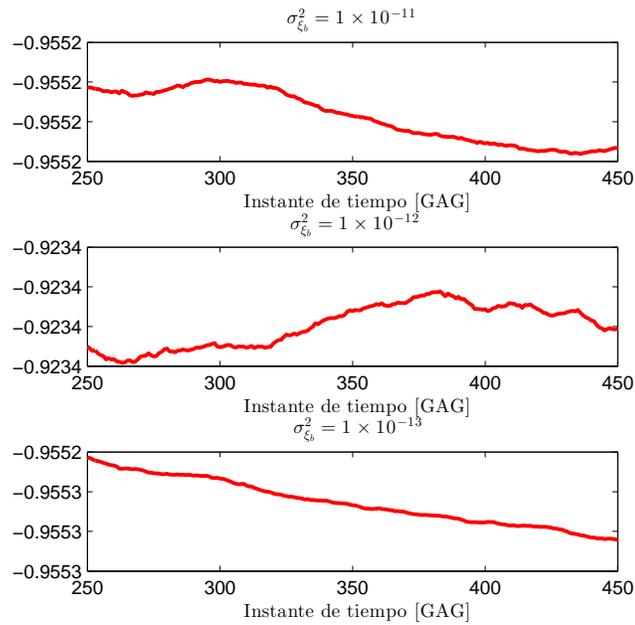


Figura 3.43: Evolución del parámetro b para diversas varianzas de ruidos asociados a la evolución artificial del parámetro b usando la estrategia de filtrado 3

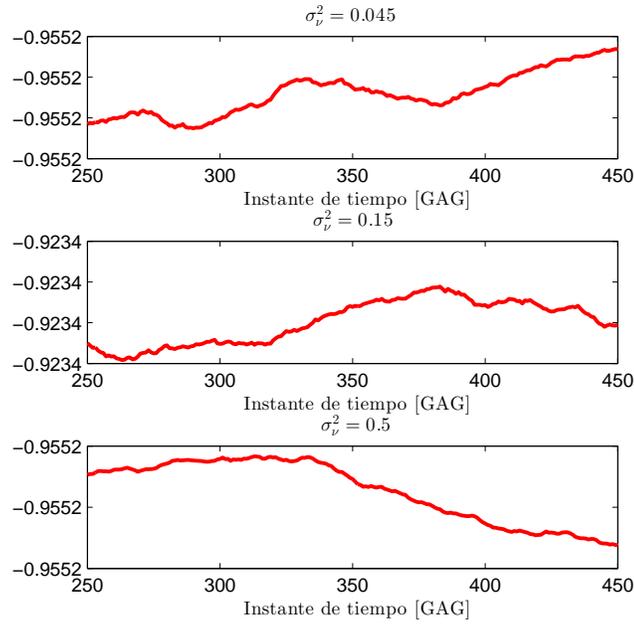


Figura 3.44: Evolución del parámetro b para diversas varianzas de ruidos de observación usando la estrategia de filtrado 3

3.3. Efecto de grandes variabilidades sobre el vector de parámetros

Como se mencionó al principio del presente capítulo, para el desarrollo de los experimentos mostrados se usa una alta variabilidad para el estado del sistema y una baja variabilidad en el vector de estados. Lo primero tiene por objetivo mantener una alta variabilidad en la muestra de partículas; lo segundo responde al supuesto de que los parámetros fijos. Dado que los algoritmos, una vez diseñados, no imponen restricciones *a priori* sobre la magnitud de la variabilidad asociada a los componentes del sistema, se ve la necesidad de exponer ciertos efectos encontrados durante los experimentos realizados en tal línea. Los datos usados para las estrategias de filtrado mostrados en esta sección se exponen en la Tabla 3.2.

Tabla 3.2: Variables modificadas usadas en las estrategias de filtrado

| Variable | Valor |
|----------------------|--------------------|
| σ_ν^2 | 0, 15 |
| σ_ω^2 | 1×10^{-9} |
| $\sigma_{\xi_a}^2$ | 1×10^{-7} |
| $\sigma_{\xi_b}^2$ | 1×10^{-8} |
| Número de partículas | 60 |

En la 3.47 se muestra la estimación producto del algoritmo. Se aprecia que esta es mejor que la mostrada anteriormente en la Figura 3.1, lo cual se debe a que la variabilidad pérdida

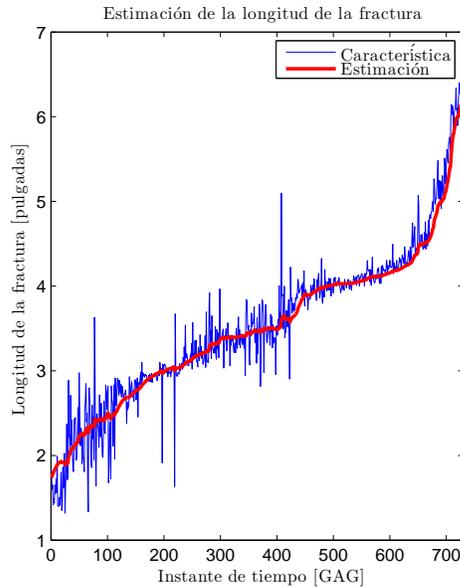


Figura 3.45: Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad

con la baja varianza de proceso se compensa de mejor forma con la alta variabilidad asociada al vector de parámetros. Esto último es responsable también de la alta incertidumbre asociada a la predicción, la cual se aprecia en 3.46.

En cuanto al desempeño de las estrategias de filtrado 1 y 2, en la Figura 3.47 se observa que el filtrado producido es de calidad inferior a lo obtenido para los valores antiguos de los parámetros (ver Figuras 3.6 y 3.11). De la misma forma, las predicciones también empeoraron, tanto en precisión como en incertidumbre, lo cual queda en evidencia al ver la Figura 3.48. Sin embargo lo más preocupante es lo expuesto en la Figura 3.49, donde se aprecia una degeneración abrupta en el peso de las partículas, la cual se consume previo al GAG 200. Tal efecto se entiende de acuerdo a la variabilidad del vector de parámetros, ya que los parámetros tienen mayor influencia sobre el valor del estado que ω . Esto muestra que las estrategias de filtrado 2 y 3 tienen que ser tratadas con cuidado, dado que una vez que degeneran su funcionamiento colapsa irremediablemente.

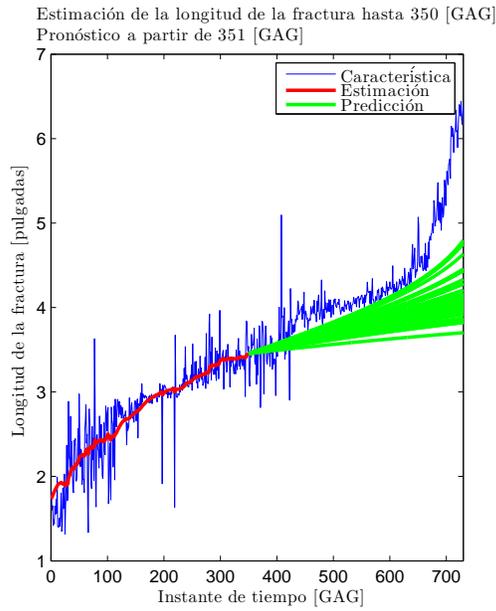


Figura 3.46: Predicción de la grieta, obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad

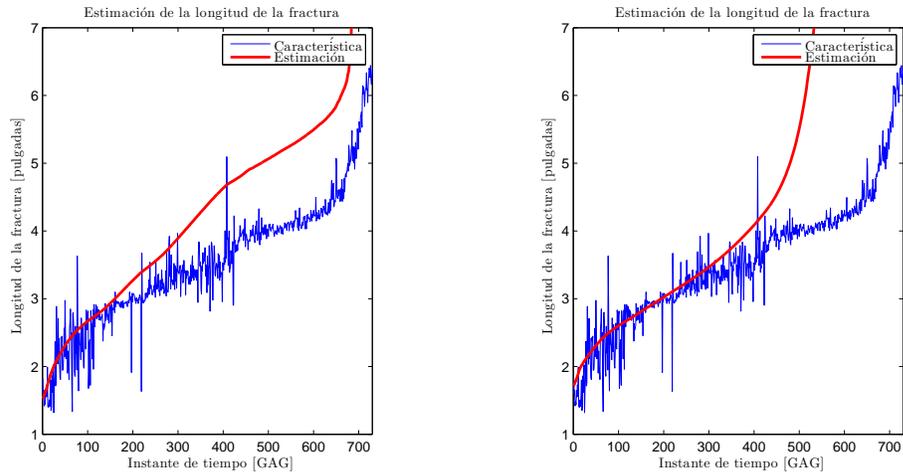


Figura 3.47: Estimación empírica de la grieta, contrastada con la estimación obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad

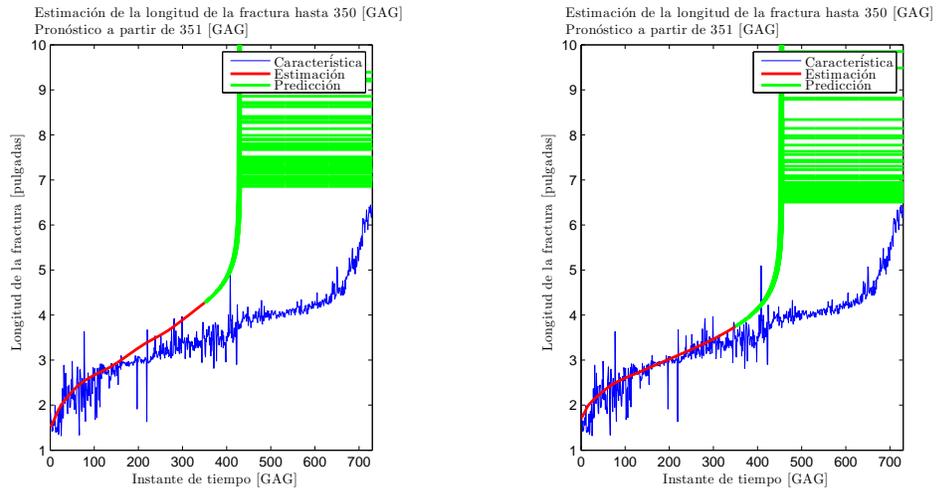


Figura 3.48: Predicción de la grieta, obtenida a partir del modelo para la estrategia de filtrado 1, usando alta variabilidad en el vector de parámetros y baja variabilidad

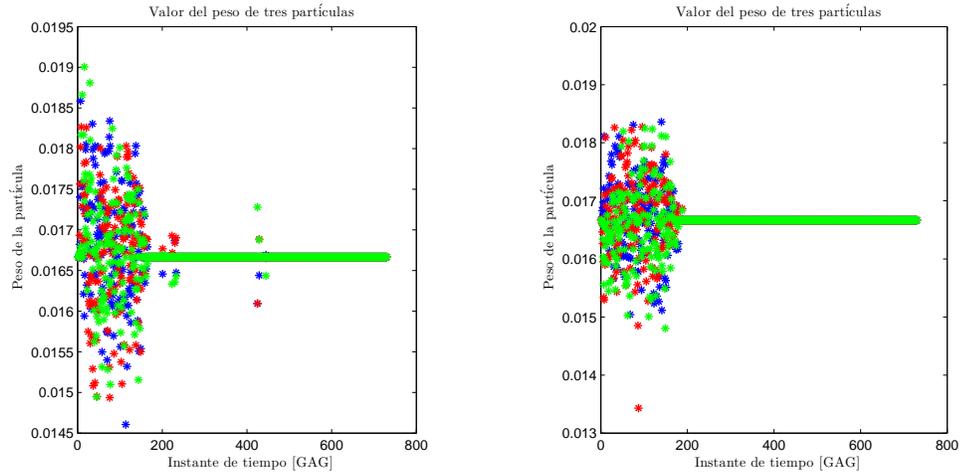


Figura 3.49: Evolución de los pesos de las partículas usadas por el filtro para alta variabilidad en el vector de parámetros y baja variabilidad en la variable de estado. Izquierda: Estrategia de filtrado 2. Derecha: Estrategia de filtrado 3

3.4. Comparación de las estrategias de filtrado y comentarios finales

En base a los resultados obtenidos, el filtro clásico con curva de ajuste ofrece una alternativa viable a la hora de filtrar datos. Su capacidades predictivas son limitadas así como su desempeño en detección. Por otra parte, las estrategias de filtrado con proyección de una trayectoria en el pasado sólo lograron ser alternativas viables como herramientas de detección, puesto que tanto en filtrado, pronóstico y robustez se vio ampliamente superado por el filtro clásico.

No obstante lo anterior, en caso de ser necesario escoger uno de las estrategias examinadas, se recomienda usar el filtro clásico. El criterio se centra, además de lo mencionado, en su rapidez de ejecución —debido a la menor cantidad de cálculos ejecutados— le dan ventaja en contextos prácticos. En particular, la simulación del estrategia de filtrado 2 demora 1.4 veces lo que demora la estrategia 1, y la de la estrategia 3 es de 1.7 veces la duración de la estrategia 1.

Conclusión

El análisis de los resultados expone que los filtros estudiados presentan aptitudes limitadas en lo que a estimación, predicción y detección de anomalías se refiere. Por otra parte, se mostró que el ajuste de la variabilidad asociada a las variables del modelo afecta de forma importante el desempeño de los algoritmos, si bien el afinamiento ejecutado no logró dar resultados esperados en cuanto a pronóstico y diagnóstico.

En lo referente a la demostración de las diferencias que se introducen al utilizar un método de actualización convencional a uno con construcción de proyecciones de trayectorias en el pasado, los resultados fueron satisfactorios. Se lograron destacar propiedades distintivas de los filtros estudiados, además de mostrar los efectos que poseen las diferentes varianzas sobre el rendimiento.

Destaca también el hecho de que las versiones basadas en proyecciones en el pasado son poco robustas y tienden a la degeneración, ya sea por la distribución de probabilidad usada (como al usar una distribución acumulada complementario de la chi-cuadrado) o por dar mucha variabilidad a los parámetros fijos del modelo.

En caso de tener que escogerse uno de los filtros analizados, en a los resultados, se sugiere el uso del filtro sin construcción de proyecciones de trayectorias en el pasado puesto que presenta mejor capacidades en filtrado y, en promedio, mejores capacidades de predicción. Además, posee consumo de recursos computacionales menor, y permite un mayor rango de ajuste en lo que a variabilidad de ruidos se refiere, lo que se traduce en un filtro más versátil.

En relación a la calidad de los resultados obtenidos, se debe reflexionar en torno al marco de pruebas usado. Puede ser que otro contexto experimental produzca resultados de mayor calidad y permita juicios de desempeño más concluyentes. Este nuevo contexto se define por la consideración de otros indicadores de falla. En particular, el uso de una métrica al momento de evaluar detección sería interesante. Por otra parte, el contexto también se vincula a un barrido más eficiente de las posibilidades de ajuste, el cual ponga en evidencia los alcances reales de tales modificaciones.

Finalmente, se concluye que lo que se puede desprender es de un alcance muy limitado, dado de que todo se dedujo a partir de una única curva de ajuste; mejorías sustanciales podrían obtenerse al hacer uso de otra función paramétricas más adecuada.

En cuanto a las líneas de trabajo futuro, lo principal es lograr mejorar los niveles de estimación, predicción y detección de los filtros asociados. Para lograr tal efecto, uno de los

frentes a estudiar sería considerar una curva de ajuste distinta. Hay que recordar que se usó la arcotangente hiperbólica, entre otras razones, debido al bajo número de parámetros de que definen la función. Por lo mismo, sería interesante observar el desempeño de los algoritmos estudiados bajo vectores de parámetros de mayor cardinalidad, los cuales pueden tener mayores recursos de ajuste dada la información contenida en los parámetros.

Otra sugerencia se relaciona con el proceso de actualización. En esta línea, sería interesante el uso de la variante sensible al riesgo del filtro de partícula, o simplemente usar otra función de densidad de importancia, una que no sea Gaussiana, como por ejemplo la distribución logística. Tales cambios permitirían el ajuste rápido ante la presencia de cambios en el comportamiento esperado de la partícula, y cambios de trayectorias instantáneos serían posibles.

Otro avance que se puede realizar es usar otro tipo de distribuciones para modelar los ruidos del problema. Esto puede ser de especial relevancia para los métodos de proyección en el pasado.

Como una alternativa más conservadora, se podría estudiar más a fondo las variantes con proyección de una trayectoria en el pasado. Por ejemplo, se podría ver el efecto de usar ventanas de distinto largo o, usar distribuciones de actualización distintas a las usadas en este trabajo. También es atractivo pensar como funcionarían estos métodos si al largo de la ventana usada variara con el tiempo.

Bibliografía

- [1] M.Sanjeev Arulampalam, Simon Maskell, Neil Gordon y Tim Clapp, *A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking*, IEEE Transactions on Signal Processing, 2, nº 50, February 2002, págs. 174–188.
- [2] James V. Candy, *Bayesian Signal Processing*, cap. Particle-Based Bayesian State-Space Processors, John Wiley & Sons, 2009.
- [3] A. Doucet, S. Godsill y C. Andrieu, *On Sequential Monte Carlo Methods for Bayesian Filtering*, Informe técnico, Department of Engineering, University of Cambridge, UK, 1998.
- [4] N. J. Gordon, D. J. Salmond y A. F. M. Smith, *Novel Approach to Non-Linear/Non-Gaussian Bayesian State Estimation*, IEE Proceedings-F, vol. 104, 1993, págs. 107–113.
- [5] Jane Liu y Mike West, *Combined parameter and state estimation in simulation-based filtering*, Sequential Monte Carlo Methods in Practice (Andrieu Doucet, Nando de Freitas y Neil Gordon, eds.), Springer-Verlag, New York, 2001.
- [6] Canh Ly, Kwok Tom, Carl S. Byington, Romano Patrick y George j. Vachtsevanos, *Fault Diagnosis and Failure Prognosis for Engineerign Systems: A Global Perspective*, 5th Annual IEEE Conference on Automation Science and Engineering, Agosto 2009, págs. 108–115.
- [7] Daniel Neira, *Aplicación de Métodos Secuenciales de Monte Carlo Sensibles al Riesgo en la Estimación de Vida Útil de Componentes*, Memoria de Título, Universidad de Chile, 2011.
- [8] Marcos E. Orchard, *A Particle Filtering-Based Framework for On-line Fault Diagnosis and Failure Prognosis*, Tesis de Doctorado, Georgia Institute of Technology, 2007.
- [9] Marcos E. Orchard, Liang Tang, Kai Goebel y George Vachtsevanos, *A Novel RSPF Approach to Prediction of High-Risk, Low-Probability Failure Events*, Annual Conference of the Prognostics and Health Management Society (San Diego, CA, USA), 2009.
- [10] Eric W. Weisstein, “*Chi-Squared Distribution.*” *From MathWorld—A Wolfram Web Resource*, <http://mathworld.wolfram.com/Chi-SquaredDistribution.html>.
- [11] ———, “*Inverse Hyperbolic Tangent.*” *From MathWorld—A Wolfram Web Resource*,

<http://mathworld.wolfram.com/InverseHyperbolicTangent.html>.

- [12] Enlu Zhou, Michael C. Fu y Steven I. Marcus, *A Particle Filtering Framework for Randomized Optimization Algorithms*, Winter Simulation Conference (College Park, MD, USA) (S. J. Mason, R. R. Hill, L. Münch, O. Rose, T. Jefferson y J. W. Fowler, eds.), 2008.

Anexo A

Tablas de datos

Nomenclatura:

A Denota al período comprendido entre el GAG 0 y el GAG 320

B Denota al período comprendido entre el GAG 0 y el GAG 730

A.1. Efecto de la variación de las varianzas sobre la estimación de los filtros

Tabla A.1: Influencia de la varianza del ruido de proceso en el error cuadrático medio para la estrategia de filtrado 1.

| Período | Longitud de la fractura [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\omega}^2 = 1 \times 10^{-2}$ | $\sigma_{\omega}^2 = 1 \times 10^{-3}$ | $\sigma_{\omega}^2 = 1 \times 10^{-4}$ |
| A | 0.230 | 0.254 | 0.277 |
| B | 0.203 | 0.268 | 0.392 |

Tabla A.2: Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 1.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ |
| A | 0.263 | 0.254 | 0.263 |
| B | 0.237 | 0.268 | 0.292 |

Tabla A.3: Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 1.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-13}$ |
| A | 0.263 | 0.254 | 0.263 |
| B | 0.277 | 0.268 | 0.274 |

Tabla A.4: Influencia de la varianza del ruido de observación en el error cuadrático medio para la estrategia de filtrado 1.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--------------------------------|-----------------------|----------------------|
| | $\sigma_\nu^2 = 0,045$ | $\sigma_\nu^2 = 0,15$ | $\sigma_\nu^2 = 0,5$ |
| A | 0.272 | 0.254 | 0.250 |
| B | 0.305 | 0.268 | 0.254 |

Tabla A.5: Influencia de la varianza del ruido de proceso en el error cuadrático medio para la estrategia de filtrado 2.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--------------------------------------|--------------------------------------|--------------------------------------|
| | $\sigma_\omega^2 = 1 \times 10^{-2}$ | $\sigma_\omega^2 = 1 \times 10^{-3}$ | $\sigma_\omega^2 = 1 \times 10^{-4}$ |
| A | 0.850 | 0.394 | 0.308 |
| B | 1.725 | 0.673 | 0.421 |

Tabla A.6: Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 2.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ |
| A | 0.401 | 0.394 | 0.336 |
| B | 2.108 | 0.673 | 2.244 |

Tabla A.7: Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 2.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-13}$ |
| A | 0.397 | 0.394 | 0.381 |
| B | 0.619 | 0.673 | 0.585 |

Tabla A.8: Influencia de la varianza del ruido de observación en el error cuadrático medio para la estrategia de filtrado 2.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--------------------------------|-----------------------|------------------------|
| | $\sigma_\nu^2 = 0,5$ | $\sigma_\nu^2 = 0,15$ | $\sigma_\nu^2 = 0,045$ |
| A | 0.378 | 0.394 | 0.430 |
| B | 0.828 | 0.673 | 1.589 |

Tabla A.9: Influencia de la varianza del ruido de proceso en el error cuadrático medio para la estrategia de filtrado 3.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\omega}^2 = 1 \times 10^{-2}$ | $\sigma_{\omega}^2 = 1 \times 10^{-3}$ | $\sigma_{\omega}^2 = 1 \times 10^{-4}$ |
| A | 0.813 | 0.377 | 0.329 |
| B | 3.773 | 3.300 | 2.000 |

Tabla A.10: Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 3.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ |
| A | 0.459 | 0.377 | 0.383 |
| B | 14.191 | 3.300 | 2.331 |

Tabla A.11: Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en el error cuadrático medio para la estrategia de filtrado 3.

| Período | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|---------|--|--|--|
| | $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-13}$ |
| A | 0.375 | 0.377 | 0.400 |
| B | 2.049 | 3.300 | 2.230 |

Tabla A.12: Influencia de la varianza del ruido de observación en el error cuadrático medio para la estrategia de filtrado 3.

| GAG de partida | $\sqrt{\text{ECM}}$ [pulgadas] | | |
|----------------|--------------------------------|-------------------------|--------------------------|
| | $\sigma_{\nu}^2 = 0,5$ | $\sigma_{\nu}^2 = 0,15$ | $\sigma_{\nu}^2 = 0,045$ |
| A | 0.448 | 0.377 | 0.479 |
| B | 2.153 | 3.300 | 1.602 |

A.2. Efecto de la variación de las varianzas sobre el pronóstico de los filtros

Tabla A.13: Influencia de la varianza del ruido de proceso en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--|--|--|
| | $\sigma_{\omega}^2 = 1 \times 10^{-2}$ | $\sigma_{\omega}^2 = 1 \times 10^{-3}$ | $\sigma_{\omega}^2 = 1 \times 10^{-4}$ |
| 90 | 23357.448 | 20374.044 | 19065.535 |
| 600 | 62.010 | 50.501 | 106.099 |

Tabla A.14: Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--|--|--|
| | $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ |
| 90 | 8513.322 | 20374.044 | 63939.215 |
| 600 | 21.754 | 50.501 | 200.876 |

Tabla A.15: Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--|--|--|
| | $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-13}$ |
| 90 | 20790.050 | 20374.044 | 21152.929 |
| 600 | 58.339 | 50.501 | 58.327 |

Tabla A.16: Influencia de la varianza del ruido de observación en la métrica de pronóstico a largo plazo para la estrategia de filtrado 1.

| Período | Magnitud del error [adimensional] | | |
|---------|-----------------------------------|-----------------------|------------------------|
| | $\sigma_\nu^2 = 0,5$ | $\sigma_\nu^2 = 0,15$ | $\sigma_\nu^2 = 0,045$ |
| 90 | 1702.963 | 20374.044 | 30826.266 |
| 600 | 61.872 | 50.501 | 66.820 |

Tabla A.17: Influencia de la varianza del ruido de proceso en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--------------------------------------|--------------------------------------|--------------------------------------|
| | $\sigma_\omega^2 = 1 \times 10^{-2}$ | $\sigma_\omega^2 = 1 \times 10^{-3}$ | $\sigma_\omega^2 = 1 \times 10^{-4}$ |
| 90 | 20484.073 | 21648.400 | 21454.111 |
| 600 | 34203.287 | 11483.378 | 4241.962 |

Tabla A.18: Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--|--|--|
| | $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ |
| 90 | 9041.748 | 21648.400 | 64538.166 |
| 600 | 2434.511 | 11483.378 | 6745.734 |

Tabla A.19: Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--|--|--|
| | $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-13}$ |
| 90 | 21935.193 | 21648.400 | 21242.168 |
| 600 | 11331.443 | 11483.378 | 7242.588 |

Tabla A.20: Influencia de la varianza del ruido de observación en la métrica de pronóstico a largo plazo para la estrategia de filtrado 2.

| Período | Magnitud del error [adimensional] | | |
|---------|-----------------------------------|-----------------------|------------------------|
| | $\sigma_\nu^2 = 0,5$ | $\sigma_\nu^2 = 0,15$ | $\sigma_\nu^2 = 0,045$ |
| 90 | 24762.142 | 21648.400 | 30795.025 |
| 600 | 8688.619 | 11483.378 | 5279.313 |

Tabla A.21: Influencia de la varianza del ruido de proceso en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--------------------------------------|--------------------------------------|--------------------------------------|
| | $\sigma_\omega^2 = 1 \times 10^{-2}$ | $\sigma_\omega^2 = 1 \times 10^{-3}$ | $\sigma_\omega^2 = 1 \times 10^{-4}$ |
| 90 | 26368.642 | 24748.158 | 26422.031 |
| 600 | 202.449 | 82.858 | 78.173 |

Tabla A.22: Influencia de la varianza del ruido asociada a la componente a del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--|--|--|
| | $\sigma_{\xi_a}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_a}^2 = 1 \times 10^{-13}$ |
| 90 | 12700.722 | 24748.158 | 79975.981 |
| 600 | 210.650 | 82.858 | 5301.158 |

Tabla A.23: Influencia de la varianza del ruido asociada a la componente b del vector de parámetros en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3.

| GAG de partida | Magnitud del error [adimensional] | | |
|----------------|--|--|--|
| | $\sigma_{\xi_b}^2 = 1 \times 10^{-11}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-12}$ | $\sigma_{\xi_b}^2 = 1 \times 10^{-13}$ |
| 90 | 28157.631 | 24748.158 | 26964.273 |
| 600 | 103.557 | 82.858 | 88.066 |

Tabla A.24: Influencia de la varianza del ruido de observación en la métrica de pronóstico a largo plazo para la estrategia de filtrado 3.

| Período | Magnitud del error [adimensional] | | |
|---------|-----------------------------------|-----------------------|------------------------|
| | $\sigma_\nu^2 = 0,5$ | $\sigma_\nu^2 = 0,15$ | $\sigma_\nu^2 = 0,045$ |
| 90 | 31404.004 | 24748.158 | 59610.501 |
| 600 | Inf | 82.858 | 2567.279 |

Anexo B

Código de los programas

B.1. MAIN_atanh.m

```
clearvars -except OUTPUT MSetot imag prjdir ...
    iTTest jTest kTest mTest nTest rt steps outPlotLeg...
    maintstart NOISE PARAM exp;
if exist('Crack_Data.mat','file')
    load Crack_Data;
else
    disp('ERROR: No se encontr?? el archivo Crack_Data.mat');
    return;
end
if (exist('PARAM','var') ~=1)
    PARAM.wm = input('Window mode? (0/1/2/3/4)');
    PARAM.n = input('No PARAM.n set; please set value default (default:730)');
end
if (isfield(PARAM,'wm') ~= 1)
    PARAM.wm = input('Window mode? (0/1/2/3/4)');
end

if (isfield(PARAM,'n') ~= 1)
    disp('No PARAM.n set; using default value');
    PARAM.n = 744;
end

if (exist('NOISE','var') ~=1)
    NOISE.a = input('No NOISE.a set; please set value (default:1e-8)');
    NOISE.b = input('No NOISE.b set; please set value (default:1e-8)');
    NOISE.f = input('No NOISE.f set; please set value (default:1e-10)');
    NOISE.obs = input('No NOISE.obs set; please set value (default:0.15)');
```

```

end

if (isfield(NOISE,'a') ~= 1)
    disp('No NOISE.a set; using default value');
    NOISE.a = 1e-8;
end

if (isfield(NOISE,'b') ~= 1)
    disp('No NOISE.b set; using default value');
    NOISE.b = 1e-8;
end

if (isfield(NOISE,'f') ~= 1)
    disp('No NOISE.f set; using default value');
    NOISE.f = 1e-10;
end

if (isfield(NOISE,'obs') ~= 1)
    disp('No NOISE.obs set; using default value');
    NOISE.obs = 0.15;
end

model.L = PARAM.n;
model.threshold = 0.85;
model.confInt = 0.95;

%-Ruido de proceso
cova = NOISE.a;
covb = NOISE.b;
covf = NOISE.f;
model.noise_p.tag = 'ruido de proceso (gaussiano)';
model.noise_p.mu = [0; 0; 0];
model.noise_p.cov = [cova, 0, 0; 0, covb, 0; 0, 0, covf];
%-Ruido de observaci??n
model.noise_o.tag = 'ruido de observaci??n (gaussiano)';
model.noise_o.mu = 0;
model.noise_o.cov = NOISE.obs;

%---Particles initialization
if(isfield(PARAM,'initF') ~= 1)
    disp('No PARAM.initF set; generating new values');
    particles.init.function = 1.344+(Y(1)-1.344)*rand(1,1); % longitud inicial de la f
else
    particles.init.function = PARAM.initF;
end
particles.init.a = 0.0029; % valor inicial de 'a'
particles.init.b = -0.9234; % valor inicial de 'b'

```

```

particles.N          = 60;    % n??mero de part??culas
particles.value      = zeros(3,particles.N);
particles.value(1,:) = 1e-6*(2*rand(1,particles.N)-1)+particles.init.a;
particles.value(2,:) = 1e-5*(2*rand(1,particles.N)-1)+particles.init.b;
particles.value(3,:) = particles.init.function;
particles.weights    = ones(1,particles.N)/particles.N;
%particles.function = ones(1,particles.N) * particles.init.function;
particles.chi2 = nan(1, particles.N);

if (PARAM.wm == 3 || PARAM.wm == 1)
    if (isfield(PARAM,'sampleN') ~= 1)
        PARAM.sampleN = input('Number of samples in window?');
    end
    particles.prev = zeros(particles.N, PARAM.sampleN, model.L);
else
    if (isfield(PARAM,'sampleN') ~= 1)
        PARAM.sampleN = 1;
    end
    particles.prev = zeros(particles.N, model.L, model.L);
end
PARAM.ws = [20 50 100]; % Window sizes for prediction error calculations

output.a.part      = zeros(model.L,particles.N); % part??culas
output.a.part(1,:) = particles.value(1,:);      % condici??n inicial
output.a.est       = zeros(model.L,1);          % estimaci??n
output.a.est(1,1)  = particles.init.a;         % condici??n inicial
output.b.part      = zeros(model.L,particles.N); % part??culas
output.b.part(1,:) = particles.value(2,:);      % condici??n inicial
output.b.est       = zeros(model.L,1);          % estimaci??n
output.b.est(1,1)  = particles.init.b;         % condici??n inicial
output.f.part      = zeros(model.L,particles.N); % part??culas
output.f.part(1,:) = particles.value(3,:);      % condici??n inicial
output.f.est       = zeros(model.L,1);          % estimaci??n
output.f.est(1,1)  = particles.init.function;

output.pred = zeros(model.L,model.L);

output.err.est = zeros(model.L,1);

output.err.pred.short = zeros(model.L-PARAM.ws(1),1);
output.err.pred.med = zeros(model.L-PARAM.ws(2),1);
output.err.pred.long = zeros(model.L-PARAM.ws(3),1);

output.part.weights = zeros(model.L,particles.N);
output.part.weights(1,:) = particles.weights;

%---Estimation of fracture's length based on observation data

```

```

for k = 2:model.L
    %-Update of particles weights and values
    if k < 321; load_u = 120; else load_u = 93; end;
    particles = update_particles_atanh(particles, model, Y, k, 1, PARAM.wm, PARAM.sampleN);
    %-Resampling
    particles = resampling(particles, model.threshold);
    %-Estimation of model parameters
    [output.a.est(k), output.b.est(k), output.f.est(k)] = output_mean_atanh(particles);
    %-Prediction based on current parameters
    output.pred(k+1:model.L,k) = funcPred(particles,output,k,model.L);
    %-Filtering error gathering
    output.err.est(k) = sqrt(sum((output.f.est(1:k)-Y(1:k)).^2)/k);
    %-Saving relevant data
    output.a.part(k,:) = particles.value(1,:);
    output.b.part(k,:) = particles.value(2,:);
    output.f.part(k,:) = particles.value(3,:);
    output.part.weights(k,:) = particles.weights;
end
%output.pred = funcPrediction1(output.pred,output,model.L);

output = predErr(Y,output,model.L,PARAM.ws); % Prediction error

}

```

B.2. tests.m

```

clear all;

for kTest = 0:4
    PARAM.wm = kTest;
    a = 1e-12;
    b = 1e-12;
    f = 1e-3;
    obs = 0.15;
    da = 0.1;
    db = 0.1;
    df = 0.1;
    dobs = 0.3;
    PARAM.iter = 30;
    PARAM.sampleN = 50;
    NOISE.param = paramGen([a,b,f,obs],[da,db,df,dobs]);
    PARAM.initF = 1.344+(1.8254-1.344)*rand(1,1);
    PARAM.n = 730;
    PARAM.particlesN = 60;
    maxPredWS = [20 50 100];
end

```

```

OUTPUT.a.part      = zeros(PARAM.n,PARAM.particlesN); % part??culas
OUTPUT.a.est       = zeros(PARAM.n,1);                % estimaci??n
OUTPUT.b.part      = zeros(PARAM.n,PARAM.particlesN); % part??culas
OUTPUT.b.est       = zeros(PARAM.n,1);                % estimaci??n
OUTPUT.f.part      = zeros(PARAM.n,PARAM.particlesN); % part??culas
OUTPUT.f.est       = zeros(PARAM.n,1);                % estimaci??n

OUTPUT.pred = zeros(PARAM.n,PARAM.n);

OUTPUT.err.est = zeros(PARAM.n,1);

OUTPUT.err.pred.short = zeros(PARAM.n-maxPredWS(1),1);
OUTPUT.err.pred.med = zeros(PARAM.n-maxPredWS(2),1);
OUTPUT.err.pred.long = zeros(PARAM.n-maxPredWS(3),1);

OUTPUT.part.weights = zeros(PARAM.n,PARAM.particlesN);
for iTest = 1 : length(NOISE.param(:,1))
    NOISE.a = NOISE.param(iTest,1);
    NOISE.b = NOISE.param(iTest,2);
    NOISE.f = NOISE.param(iTest,3);
    NOISE.obs = NOISE.param(iTest,4);
    for jTest = 1 : PARAM.iter
        MAIN_atanh;
        OUTPUT.a.part      = OUTPUT.a.part + output.a.part;
        OUTPUT.a.est       = OUTPUT.a.est + output.a.est;
        OUTPUT.b.part      = OUTPUT.b.part + output.b.part;
        OUTPUT.b.est       = OUTPUT.b.est + output.b.est;
        OUTPUT.f.part      = OUTPUT.f.part + output.f.part;
        OUTPUT.f.est       = OUTPUT.f.est + output.f.est;
        OUTPUT.pred = OUTPUT.pred + output.pred;
        OUTPUT.err.est = OUTPUT.err.est + output.err.est;
        OUTPUT.err.pred.short = OUTPUT.err.pred.short + output.err.pred.short;
        OUTPUT.err.pred.med = OUTPUT.err.pred.med + output.err.pred.med;
        OUTPUT.err.pred.long = OUTPUT.err.pred.long + output.err.pred.long;
        OUTPUT.part.weights      = OUTPUT.part.weights + output.part.weights;
    end
    OUTPUT.a.part      = OUTPUT.a.part ./ PARAM.iter;
    OUTPUT.a.est       = OUTPUT.a.est ./ PARAM.iter;
    OUTPUT.b.part      = OUTPUT.b.part ./ PARAM.iter;
    OUTPUT.b.est       = OUTPUT.b.est ./ PARAM.iter;
    OUTPUT.f.part      = OUTPUT.f.part ./ PARAM.iter;
    OUTPUT.f.est       = OUTPUT.f.est ./ PARAM.iter;
    OUTPUT.pred = OUTPUT.pred ./ PARAM.iter;
    OUTPUT.err.est = OUTPUT.err.est ./ PARAM.iter;
    OUTPUT.err.pred.short = OUTPUT.err.pred.short ./ PARAM.iter;
    OUTPUT.err.pred.med = OUTPUT.err.pred.med ./ PARAM.iter;
    OUTPUT.err.pred.long = OUTPUT.err.pred.long ./ PARAM.iter;

```

```

    OUTPUT.part.weights    = OUTPUT.part.weights ./ PARAM.iter;
    filename = ['OUTPUT_wm',num2str(PARAM.wm),'_exp',num2str(iTest)];
    save(filename, 'OUTPUT', 'PARAM','NOISE');
end
clearvars -except kTest
end
}

```

B.3. Update_particles_atanh.m

```

%-Actualizaci??n del valor de las part??culas y sus pesos
function particles = update_particles_atanh(particles, model, obsTot, iter, n, ...
    mem, sampleN)
% n es un par??metro que dice si se actualizan los pesos, n=1, o no, n=0
obs = obsTot(iter);
if(n)
    pNoise = gauss_sample(model.noise_p,particles.N);
else
    mu = model.noise_p.mu;
    pNoise = mu(:,ones(1,particles.N));
end;

particles.value = particles.value + pNoise;
%LIMITING VALUES
% particles.value(1,particles.value(1,:)<=0) = 1;
% particles.value(2,particles.value(2,:)<=0) = 1;

%CALULATING NEW FUNCTION VALUES CONSIDERING LIMITING
X = particles.value(1,:).*iter + particles.value(2,:);
particles.value(3,:) = particles.value(3,:)-...
    (particles.value(1,:)./(X.^2-1)).*(X<=(1-1e-5));

if(n)
    switch mem
        case 4 %proyeccion de curvas con (1-CDF), ventana creciente
            particles = chi2test_atanhV4(model.noise_o,particles, iter, obsTot);
            likelihood = particles.chi2 + 1e-99;
        case 3 %proyeccion de curvas con (1-CDF), ventana movil
            particles = chi2test_atanhV3(model.noise_o,particles, iter, obsTot, sampleN);
            likelihood = particles.chi2 + 1e-99;
        case 2 %proyeccion de curvas hacia adelante con ventana creciente
            particles = chi2test_atanhV2(model.noise_o,particles, iter, obsTot);
            likelihood = particles.chi2 + 1e-99;
    end
end

```

```

    case 1 %proyeccion de curvas hacia adelante con ventana movil
        particles = chi2test_atanhV1(model.noise_o,particles, iter, obsTot, sample);
        likelihood = particles.chi2 + 1e-99;
    case 0
        error = obs - particles.value(3,:);
        likelihood = gauss_likelihood(model.noise_o,error) + 1e-99;
    end
    particles.weights = particles.weights.*likelihood;
    particles.weights = particles.weights/sum(particles.weights);
end

end
}

```

B.4. gauss_sample.m

```

%-Obtención de una muestra de ruido que distribuye como una normal
function sample = gauss_sample(noise, N)
    M = size(noise.mu,1);
    sample = chol(noise.cov)*randn(M,N) + noise.mu(:,ones(1,N));
end}

```

B.5. gauss_likelihood.m

```

%-Cálculo de la verosimilitud en base a una distribución normal
function likelihood = gauss_likelihood(oNoise, error)
    nov = size(error,2);
    normfact = (2*pi)^(size(oNoise.mu,1)/2);
    XX = error - oNoise.mu(:,ones(1,nov));
    S = chol(oNoise.cov)';
    foo = S \ XX;
    likelihood = exp(-0.5*sum(foo.*foo, 1))./(normfact*abs(prod(diag(S))));
end}

```

B.6. chi2test_atanhV1.m

```

function particles = chi2test_atanhV1(oNoise,particles, iter, obs, sn)
%Proyección de curvas hacia adelante con ventana móvil
obs = obs(1:iter);
obs = obs(:,ones(particles.N,1));

```

```

aux = zeros(particles.N,iter);
st.a = particles.value(1,:)';
st.b = particles.value(2,:)';
aux(:,1) = particles.init.function*ones(particles.N,1);
df = D_atanh(1:iter-1,st);

for j=1:iter-1
    aux(:,j+1) = aux(:,j)+df(:,j);
end
if (iter < sn)
    error = obs-aux';
    XX = error - oNoise.mu(ones(iter,1),ones(particles.N,1));
    SE = sum(XX.^2)/oNoise.cov;
    particles.chi2 = chi2pdf(SE,iter);
else
    error = obs(iter-sn+1:iter,:)-aux(:,iter-sn+1:iter)';
    XX = error - oNoise.mu(ones(sn,1),ones(particles.N,1));
    SE = sum(XX.^2)/oNoise.cov;
    particles.chi2 = chi2pdf(SE,sn);
end
}

```

B.7. chi2test_atanhV2.m

```

function particles = chi2test_atanhV2(oNoise,particles, iter, obs)
%Proyección de curvas hacia adelante
obs = obs(1:iter);
obs = obs(:,ones(particles.N,1));
aux = zeros(particles.N,iter);
st.a = particles.value(1,:)';
st.b = particles.value(2,:)';
aux(:,1) = particles.init.function*ones(particles.N,1);
df = D_atanh(1:iter-1,st);
for j=1:iter-1
    aux(:,j+1) = aux(:,j)+df(:,j);
end
error = obs-aux';
XX = error - oNoise.mu(ones(iter,1),ones(particles.N,1));
SE = sum(XX.^2)/oNoise.cov;
particles.chi2 = chi2pdf(SE,iter);
}

```

B.8. chi2test_atanhV3.m

```
function particles = chi2test_atanhV3(oNoise,particles, iter, obs, sn)
%Proyección de curvas hacia adelante; evaluación con (1-CDF); ventana móvil
obs = obs(1:iter);
obs = obs(:,ones(particles.N,1));
aux = zeros(particles.N,iter);
st.a = particles.value(1,:)';
st.b = particles.value(2,:)';
aux(1) = particles.init.function;
df = D_atanh(1:iter-1,st);

for j=1:iter-1
    aux(:,j+1) = aux(:,j)+df(:,j);
end
if (iter < sn)
    error = obs-aux';
    XX = error - oNoise.mu(ones(iter,1),ones(particles.N,1));
    SE = sum(XX.^2)/oNoise.cov;
    particles.chi2 = 1-cdf('chi2',SE,iter);
else
    error = obs(iter-sn+1:iter,:)-aux(:,iter-sn+1:iter)';
    XX = error - oNoise.mu(ones(sn,1),ones(particles.N,1));
    SE = sum(XX.^2)/oNoise.cov;
    particles.chi2 = 1-cdf('chi2',SE,sn);
end
}
```

B.9. chi2test_atanhV4.m

```
function particles = chi2test_atanhV4(oNoise,particles, iter, obs)
%Proyección de curvas hacia adelante; evaluación con (1-CDF);ventana
%creciente
obs = obs(1:iter);
obs = obs(:,ones(particles.N,1));
aux = zeros(particles.N,iter);
st.a = particles.value(1,:)';
st.b = particles.value(2,:)';
aux(1) = particles.init.function;
df = D_atanh(1:iter-1,st);

for j=1:iter-1
    aux(:,j+1) = aux(:,j)+df(:,j);
end
```

```

error = obs-aux';
XX = error - oNoise.mu(ones(iter,1),ones(particles.N,1));
SE = sum(XX.^2)/oNoise.cov;
particles.chi2 = 1-cdf('chi2',SE,iter);

}

```

B.10. resampling.m

```

%-Remuestreo de partículas
function particles = resampling(particles, threshold)
    St = round(threshold*particles.N);
    if sum(particles.weights.^2) >= 1/St
        outIndex = residualresample(1:particles.N,particles.weights);
        particles.value = particles.value(:,outIndex);
        particles.weights = ones(1,particles.N)/particles.N;
    end
end}

```

B.11. residualresample.m

```

function outIndex = residualResample(inIndex, weights);

if (nargin ~= 2),
    error(' [ residualResample ] Not enough input arguments. ');
end

S = length(weights);          % S = Number of particles.

outIndex = zeros(1,S);      % setup output index buffer

%=== RESIDUAL RESAMPLING =====

N_kind= zeros(1,S);

% first integer part
weights_res = S*weights;
N_kind = fix(weights_res);

% residual number of particles to sample

```

```

N_res = S-sum(N_kind);

if N_res

    weights_res = (weights_res-N_kind)/N_res;
    cumDist = cumsum(weights_res);

    % generate N_res ordered random variables uniformly distributed in [0,1]
    u = fliplr(cumprod(rand(1,N_res).^(1./(N_res:-1:1))));
    j=1;
    for i=1:N_res
        while (u(1,i)>cumDist(1,j))
            j=j+1;
        end
        N_kind(1,j)=N_kind(1,j)+1;
    end;

end;

%=== COPY RESAMPLED TRAJECTORIES =====

index=1;
for i=1:S
    if (N_kind(1,i)>0)
        for j=index:index+N_kind(1,i)-1
            outIndex(j) = inIndex(i);
        end;
    end;
    index = index+N_kind(1,i);
end

%-----

}

```

B.12. output_mean_atanh.m

```
%-Salida calculada como la media de las partículas y sus pesos
function [crack, alpha, foo] = output_mean_atanh(particles)
    crack = sum(particles.value(1,:).*particles.weights);
    alpha = sum(particles.value(2,:).*particles.weights);
    foo = sum(particles.value(3,:).*particles.weights);
end
```

B.13. D_atanh.m

```
function dy = D_atanh(x1,st)
    X = kron(x1,st.a) + st.b(:,ones(length(x1),1));
    dy = (-st.a(:,ones(length(x1),1)) ./ ...
        (X.^2 - ones(length(st.a),length(x1)))).*(X <= (1-1e-5)));}
```

B.14. funcPred.m

```
function pred = funcPred(particles,output,iter,lim)
    st.a = particles.value(1,:)' ;
    st.b = particles.value(2,:)' ;
    w =particles.weights;
    bar = zeros(length(st.a),lim-iter+1);
    bar(:,1) = output.f.est(iter);
    delta= D_atanh(iter : lim-1,st);
    for j = 1 : lim-iter
        bar(:,j+1) = bar(:,j) + delta(:,j);
    end
    pred = (w*bar(:,2:lim-iter+1))';
end}
```

B.15. estErr.m

```
function estE = estErr(output, obs, lim)
    estE = zeros(lim,1);
    for i = 1 : lim
        estE(i) = sqrt(sum((output.f.est(1:i)-obs(1:i)).^2)/i);
    end

end
```

```
}
```

B.16. predErr.m

```
function output = predErr(Y,output,lim,ws)

    obs = Y(1:lim);
    obs = obs(:,ones(lim,1));
    err = output.pred- obs;
    for i = 1 : lim-ws(1)

        range = confRange(output,i,lim);
        if (i<=(lim-ws(3)))
            aux = zeros(ws(3),1);
            for j = i+1 : i+ws(3)
                aux(j-i) = err(j,i) / range(j);
            end
            output.err.pred.short(i) = sqrt(sum(aux(1:ws(1)).^2));
            output.err.pred.med(i) = sqrt(sum(aux(1:ws(2)).^2));
            output.err.pred.long(i) = sqrt(sum(aux(1:ws(3)).^2));
        else
            if (i<=(lim-ws(2)))
                aux = zeros(ws(2),1);
                for j = i+1 : i+ws(2)
                    aux(j-i) = err(j,i) / range(j);
                end
                output.err.pred.short(i) = sqrt(sum(aux(1:ws(1)).^2));
                output.err.pred.med(i) = sqrt(sum(aux(1:ws(2)).^2));
            else
                aux = zeros(ws(1),1);
                for j = i+1 : i+ws(1)
                    aux(j-i) = err(j,i) / range(j);
                end
                output.err.pred.short(i) = sqrt(sum(aux(1:ws(1)).^2));
            end
        end

    end
end
}
```

B.17. confRange.m

```
function range = confRange(OUTPUT,iter,lim)
range = nan(1,lim);
st.a = OUTPUT.a.part(iter,:)' ;
st.b = OUTPUT.b.part(iter,:)' ;
%bar = zeros(length(st.a),lim-iter+1);
bar = zeros(length(st.a),lim-iter+1);
bar(:,1) = OUTPUT.f.est(iter)*ones(length(st.a),1);
delta= D_atanh(iter : lim-1,st);
for j = 1 : lim-iter
    bar(:,j+1) = bar(:,j) + delta(:,j);
end
range(iter:lim) = max(bar) - min(bar);
}
```

B.18. paramGen.m

```
function param = paramGen(args, var)
n = length(args);
param = args(ones(2*n+1,1),:);
j = 1;
for i = 1:n
    j = j+1;
    param(j,i) = param(1,i) * var(i);
    j = j+1;
    param(j,i) = param(1,i) / var(i);
end
}
```

B.19. currentStatePrognosis.m

```
function bar = currentStatePrognosis(OUTPUT,iter,lim)
range = nan(1,lim);
st.a = OUTPUT.a.part(iter,:)' ;
st.b = OUTPUT.b.part(iter,:)' ;
bar = zeros(length(st.a),lim-iter+1);
bar(:,1) = OUTPUT.f.est(iter)*ones(length(st.a),1);
delta= D_atanh(iter : lim-1,st);
for j = 1 : lim-iter
    bar(:,j+1) = bar(:,j) + delta(:,j);
end
}
```

```
end  
}
```