Resumen en castellano

Estudio sobre la ciberseguridad en sistemas de control distribuidos basados en MPC

Este trabajo se enmarca dentro del campo del control predictivo basado en modelo (MPC, por sus siglas en inglés), con especial enfásis en problemas de control distribuido (DMPC).

El proyecto está orientado a sistemas compuestos por múltiples subsistemas con dinámicas acopladas interactuando entre sí, y que a su vez están controlados en base al MPC. El objetivo será la evolución óptima del sistema global operando de manera descentralizada mediante la incorporación del algoritmo de control propuesto en [1]. De esta manera, el comienzo de este trabajo es el estudio e implementación de dicho algoritmo, el cual es analizado en primer lugar en condiciones estándares de funcionamiento.

Con ello, se procede a considerar la posible presencia de agentes maliciosos en el sistema dispuestos a inyectar información que pueda comprometer su evolución, lo que genera una importante brecha de seguridad. Es en esta última situación en la que el trabajo hace especial hincapié, de ahí el notable enfoque hacia problemas relacionados con ciberseguridad.

El estudio de los ataques que puede sufrir el algoritmo comienza con la presentación de distintas posibilidades con las que cuenta un agente malicioso durante el desarrollo del mismo para introducir información falsa, así como el mecanismo por el cual ésta es extendida por el sistema. Igualmente, se expone cómo un agente dispuesto a atacar puede optimizar algunas de dichas posibilidades para lograr un mayor grado de aprovechamiento.

Finalmente, se introduce brevemente la técnica denominada min-max con el fin de desarrollar un mecanismo para la reacción ante dichos ataques, de manera que se mitiguen los problemas derivados de éstos.

La exposición teórica de lo aquí comentado está sucedida por una serie de simulaciones que permitirán probar los resultados presentados analíticamente. A continuación se muestra un breve resumen de lo incluido en los distintos apartados, en el que se incluyen algunos de los resultados más importantes expuestos en ellos.

Capítulo 2: Modelo del sistema acoplado

En este capítulo se presenta el modelo dinámico de los sistemas acoplados que se van a utilizar a lo largo del trabajo. Lo importante de esta primera parte es la definición que aquí se muestra (1), ya que será uno de los puntos de partida para desarrollos posteriores. Como se puede observar, se va a considerar un modelo que considera los efectos de las interacciones entre subsistemas definiendo las últimas como acoplamientos en los estados y en las variables de control.

$$\begin{aligned} x_i(k+1) = A_{ii}x_i(k) + B_{ii}u_i(k) + \sum_{j=1, j\neq i}^{M} \left[A_{ij}x_j(k) + B_{ij}u_j(k)\right] \\ y_i(k) = C_i x_i(k) \\ i = 1, 2, ..., M. \end{aligned} \tag{1}$$

Se define como M el número de subsistemas en la planta, que a su vez se diferenciarán con los subíndices i, j o l.

Extensión del modelo para un horizonte de predicción N

La anterior presentación es continuada por la definición de la expresión asociada al modelo para los siguientes N tiempos de muestreo (por tanto, N denotará el horizonte de predicción). En esta parte el objetivo es llegar a una relación que partiendo de un determindado instante k nos proporcione una predicción de lo que ocurrirá en la planta desde el momento actual hasta el momento $k+N$. Se introduce aquí el uso de vectores que representan trayectorias, cuyo empleo será frecuente en el trabajo dadas las características del problema.

Mostramos abajo las definiciones más importantes de esta parte y la extensión del modelo buscada (2).

$$w_i(k) = \sum_{j=1, j\neq i}^{M} \left[A_{ij}x_j(k) + B_{ij}u_j(k)\right]$$

$$G_{xi} = \begin{bmatrix} A_{ii} \\ A_{ii}^2 \\ \vdots \\ A_{ii}^N \end{bmatrix}, \qquad G_{ui} = \begin{bmatrix} B_{ii} & & & \\ A_{ii}B_{ii} & B_{ii} & & \\ \vdots & & \ddots & \\ A_{ii}^{N-1}B_{ii} & \cdots & \cdots & B_{ii} \end{bmatrix}, \qquad G_{wi} = \begin{bmatrix} I & & & \\ A_{ii} & I & & \\ \vdots & & \ddots & \\ A_{ii}^{N-1} & \cdots & \cdots & I \end{bmatrix}$$

$$G_{wi}^{x_j} = \begin{bmatrix} I & & & \\ A_{ii} & I & & \\ \vdots & & \ddots & \\ A_{ii}^{N-1} & \cdots & \cdots & I \end{bmatrix} \begin{bmatrix} A_{ij} \\ A_{ij}^2 \\ \vdots \\ A_{ij}^N \end{bmatrix}, \qquad G_{wi}^{u_j} = \begin{bmatrix} I & & & \\ A_{ii} & I & & \\ \vdots & & \ddots & \\ A_{ii}^{N-1} & \cdots & \cdots & I \end{bmatrix} \begin{bmatrix} B_{ij} & & & \\ A_{ij}B_{ij} & B_{ij} & & \\ \vdots & & \ddots & \\ A_{ij}^{N-1}B_{ij} & \cdots & \cdots & B_{ij} \end{bmatrix}$$

$$\mathbf{x}_i(k) = \begin{bmatrix} x_i(k|k) \\ x_i(k+1|k) \\ \vdots \\ x_i(k+N-1|k) \end{bmatrix}, \qquad \mathbf{u}_i(k) = \begin{bmatrix} u_i(k|k) \\ u_i(k+1|k) \\ \vdots \\ u_i(k+N-1|k) \end{bmatrix}, \qquad \mathbf{w}_i(k) = \begin{bmatrix} w_i(k|k) \\ w_i(k+1|k) \\ \vdots \\ w_i(k+N-1|k) \end{bmatrix}$$

$$\begin{aligned} \mathbf{x}_i(k+1) &= G_{xi}x_i(k|k) + G_{ui}\mathbf{u}_i(k) + G_{wi}\mathbf{w}_i(k) \\ \mathbf{x}_i(k+1) &= G_{xi}x_i(k|k) + G_{ui}\mathbf{u}_i(k) + \sum_{j=1, j\neq i}^{M} \left[G_{wi}^{x_j}x_j(k|k) + G_{wi}^{u_j}\mathbf{u}_j(k) \right] \end{aligned} \tag{2}$$

Modelo centralizado asociado

Para finalizar este capítulo se introduce el modelo centralizado asociado al sistema global cuyos subsistemas están definidos con el modelo (1).

$$\underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix}}_{x(k+1)}(k+1) = \underbrace{\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1M} \\ A_{21} & A_{22} & \cdots & A_{2M} \\ \vdots & & \ddots & \\ A_{M1} & A_{M2} & \cdots & A_{MM} \end{bmatrix}}_{A_{cen}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix}}_{x(k)}(k) + \underbrace{\begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1M} \\ B_{21} & B_{22} & \cdots & B_{2M} \\ \vdots & & \ddots & \\ B_{M1} & B_{M2} & \cdots & B_{MM} \end{bmatrix}}_{B_{cen}} \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_M \end{bmatrix}}_{u(k)} \tag{3}$$

Capítulo 3: De MPC basado sólo en la comunicación entre agentes a MPC basado en la cooperación

MPC basado en comunicación

Se presenta aquí una pequeña descripción de las características de esta estrategia de control. Como definición importante tenemos la expresión (4), ya que será utilizada más tarde por los agentes para evaluar su situación local, lo cual adquiere gran utilidad en el estudio de los ataques.

$$\begin{aligned} &\phi_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{x}_{j\neq i}^{p-1}(k), \mathbf{u}_{j\neq i}^{p-1}(k); x_i(k|k)) = \\ &\sum_{n=0}^{N-1} x_i^T(k+n|k)Q_i(k+n|k)x_i(k+n|k) + u_i^T(k+n|k)R_i(k+n|k)u_i(k+n|k) \end{aligned} \tag{4}$$

MPC basado en cooperación

Haciendo uso de la función anterior, en esta parte se procede a introducir cierta cooperación entre los agentes en la implentación del algoritmo de control. Definimos aquí la función objetivo asociada a esta útima situación (5), donde λ_i son los pesos correspondientes a cada subsistema i.

$$\phi_{i,c}(\mathbf{u}_i(k), \mathbf{u}_{j\neq i}^{p-1}(k); x(k|k)) = \sum_{l=1}^{M} \lambda_l \phi_l(\mathbf{u}_i(k), \mathbf{u}_{j\neq i}^{p-1}(k); x_l(k|k)) \tag{5}$$

Igualmente, se incluye un desarrollo matemático que permite pasar de (5) a (6) cuando dicha función es utilizada como función objetivo en un problema de optimización en la variable $\mathbf{u}_i(k)$. No se muestra en este resumen las expresiones de los elementos en (6), estando éstas indicadas en el correspondiente apartado de la memoria.

$$\min_{\mathbf{u}_i} \qquad \frac{1}{2}\mathbf{u}_i(k)^T \mathscr{H}_i \mathbf{u}_i(k) + \left(r_i(\mathbf{u}_{j\neq i}^{p-1}(k)) + q_i(x(k|k)) \right)^T \mathbf{u}_i(k) \tag{6}$$

Capítulo 4: Diseño del controlador

En este capítulo se expone la manera elegida para implementar el caso de cooperación haciéndo uso del algoritmo base para este proyecto, el cual se presenta en [1].

Modelo centralizado

Partimos del modelo centralizado de la planta, que igualmente es extendido para trabajar con un horizonte de predicción N. Definimos aquí el problema de optimización para el caso en el que se controla desde una perspectiva centralizada (7).

$$\min_{\mathbf{u}(k)} \phi(\mathbf{x}(k), \mathbf{u}(k); x(k|k)) = \frac{1}{2}\mathbf{u}(k)^T H \mathbf{u}(k) + F^T \mathbf{u}(k) \tag{7}$$

$$H = G_u^T \widehat{Q} G_u + \widehat{R}$$

$$F = G_u^T \widehat{Q} G_x x(k|k)$$

$$\mathbf{x}(k+1) = \begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N|k) \end{bmatrix}, G_x = \begin{bmatrix} A_{cen} \\ A_{cen}^2 \\ \vdots \\ A_{cen}^N \end{bmatrix}, G_u = \begin{bmatrix} B_{cen} & & & \\ A_{cen}B_{cen} & B_{cen} & & \\ \vdots & & \ddots & \\ A_{cen}^{N-1}B_{cen} & \cdots & \cdots & B_{cen} \end{bmatrix}, \mathbf{u}(k) = \begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N-1|k) \end{bmatrix}$$

$$\widehat{Q} = \begin{bmatrix} Q(k+1) & & & \\ & Q(k+2) & & \\ & & \ddots & \\ & & & Q(k+N) \end{bmatrix}, \quad \widehat{R} = \begin{bmatrix} R(k) & & & \\ & R(k+1) & & \\ & & \ddots & \\ & & & R(k+N-1) \end{bmatrix}$$

$$Q(k) = \begin{bmatrix} Q_1(k) & & & \\ & Q_2(k) & & \\ & & \ddots & \\ & & & Q_M(k) \end{bmatrix}, \quad R(k) = \begin{bmatrix} R_1(k) & & & \\ & R_2(k) & & \\ & & \ddots & \\ & & & R_M(k) \end{bmatrix}$$

Cambio de variable

El cambio de variable aplicado es la idea clave para pasar del caso centralizado a un problema distribuido y afrontar el control de manera descentralizada. Dicho cambio es presentado en (8), donde las matrices $\mathbf{M}_i$ están formadas por ceros y unos de modo que permiten el desacoplo buscado.

$$\mathbf{u}(k) = \sum_{i=1}^{M} \mathbf{M}_i \mathbf{u}_i(k) \tag{8}$$

Introduciendo (8) en la función objetivo del caso centralizado definimos los problemas a resolver por cada agente i según el algoritmo de referencia. Con ello, se tiene que dichos problemas están determinados por (9).

$$\frac{1}{2}\mathbf{u}_i(k)^T \left(\mathbf{M}_i^T H \mathbf{M}_i\right) \mathbf{u}_i(k) + \left(\sum_{j \neq i}^{M} \mathbf{u}_j^{p-1}(k)^T \mathbf{M}_j^T H \mathbf{M}_i + F^T \mathbf{M}_i \right)^T \mathbf{u}_i(k) \tag{9}$$

$$H_i = \mathbf{M}_i H \mathbf{M}_i$$

$$F_i^p(k) = \mathbf{M}_i^T H \sum_{j \neq i}^{M} \mathbf{M}_j \mathbf{u}_j^{p-1}(k) + \mathbf{M}_i^T F$$

Restricciones

El principal resultado de esta sección es el mostrado en (11). La finalidad es pasar de las restricciones dadas sobre los estados y entradas para cada instante de tiempo a una inecuación que nos represente lo mismo pero sobre la trayectoria de acciones de control a optimizar por los agentes. Para ello además se hace uso de la inecuación equivalente para el caso centralizado (10).

$$\underbrace{\begin{bmatrix} \widehat{A}_x G_u \\ \widehat{A}_u \end{bmatrix}}_{AU} \mathbf{u}(k) \leq \underbrace{\begin{bmatrix} \widehat{b}_x - \widehat{A}_x G_x x(k|k) \\ \widehat{b}_u \end{bmatrix}}_{bU} \tag{10}$$

$$\underbrace{\begin{bmatrix} \widehat{A}_x G_u \mathbf{M}_i \\ \widehat{A}_u \mathbf{M}_i \end{bmatrix}}_{AU_{dec,i}} \mathbf{u}_i(k) \leq \underbrace{\begin{bmatrix} \widehat{b}_x - \widehat{A}_x G_x x(k|k) - \widehat{A}_x G_u(\sum_{j\neq i}^{M} \mathbf{M}_j \mathbf{u}_j(k)) \\ \widehat{b}_u - \widehat{A}_u(\sum_{j\neq i}^{M} \mathbf{M}_j \mathbf{u}_j(k)) \end{bmatrix}}_{bU_{dec,i}} \tag{11}$$

Algoritmo

Se muestra aquí paso a paso en qué consiste el algoritmo de distribuido en cuestión. Partiendo de los parámetros necesarios para su implementación, para cada instante de tiempo k, se puede resumir brevemente en:

1. Determinar H,F,AU,bU con $x(k|k)$.
2. Entrar en un bucle *while* con $p = 0$. Dicho bucle está condicionado por $dist_i < \epsilon$ para todos los subsistemas i o $p > p_{max}$.
3. Definir $AU_{dec,i}$ y $bU_{dec,i}$.
4. Minimización de (9) en la variable $\mathbf{u}_i(k)$ por parte de cada agente para definir cada $\mathbf{u}^p_{i,opt}(k)$.
5. Calcular $\mathbf{u}^p_i(k)$ y predecir las trayectorias de estados asociadas. Esto se hará introduciendo cierta inercia en el algoritmo.
6. Guardar las trayectorias para su uso en una posible próxima iteración (esta información será compartida entre agentes)
7. Calcular cada $dist_i$ e incrementar p.
8. Posibles casos:

   -Si $p > p_{max}$ o $dist_i < \epsilon$ para todo i, entonces aplicamos las primeras entradas de $\mathbf{u}^p_i(k)$.

   -Si $p \leq p_{max}$ o $dist_i \geq \epsilon$ para algún i, entonces volvemos al segundo paso.

Capítulo 5: Estabilidad del algoritmo

Funciones cuadráticas

Se introducen aquí brevemente algunas características de las formas cuadráticas que son utilizadas y se demuestra la convexidad de una expresión genérica.

Visión general de las funciones objetivos empleadas

Se añaden algunos comentarios sobre las funciones objetivo utilizadas y se aplican los resultados anteriores, de manera que se reitera la suposición de que trabajamos con funciones convexas y definidas positivas.

El algoritmo durante las negociaciones y su desarrollo en el tiempo

En esta parte se aplica la definición matemática de convexidad para demostrar que la inclusión de la inercia en el algoritmo no afecta al descenso del coste por iteración, de manera que el principal resultado que se aporta es el expuesto en (12).

$$\phi^p_{i,c}(\mathbf{u}^p_i(k), \mathbf{u}^p_{j\neq i}(k), x(k|k)) \leq \phi^{p-1}_{i,c}(\mathbf{u}^{p-1}_i(k), \mathbf{u}^{p-1}_{j\neq i}(k), x(k|k)) \tag{12}$$

Asimismo, se analiza qué ocurre cuando se pasa de un cierto k a $k+1$ y se muestra que, haciendo uso del resultado anterior, se llega a una caída o la estabilidad del valor del coste a lo largo del tiempo.

Capítulo 6: Ataques al esquema de DMPC

Objetivo del atacante

Antes de adentrarnos en la presentación de distintas formas de atacar al algoritmo, definimos aquí cual es la función de coste que valora el bienestar de cada subsistema desde un punto de vista local y cuya reducción será el fin por el que actuar de manera irregular. Dicho coste será aproximado por (4).

Falsa referencia

Un agente malicioso a que hace uso de esta posibilidad busca una reducción de su coste local mediante la alteración de $x_{a,ref}$ presente en la función a optimizar según el algoritmo. De esta manera, el problema que resolverá pasará a ser

$$\min_{\mathbf{u}_a(k)} \ \frac{1}{2}\mathbf{u}_a(k)^T(\mathbf{M}_a H \mathbf{M}_a)\mathbf{u}_a(k) + \left(\sum_{j\neq a}^{M} \mathbf{u}_j^{p-1}(k)^T \mathbf{M}_j^T H \mathbf{M}_a + F^T(k, x^f_{a,ref})\mathbf{M}_a\right)\mathbf{u}_a(k)$$

En primer lugar se expone el caso en el que dicha referencia es un valor dado que se mantiene constante a lo largo del desarrollo del algoritmo, así como el mecanismo por el cual la información falsa alcanza el resto de subsistemas alterando así la evolución global.

Búsqueda de una referencia óptima

Este caso se introduce para logar la mayor optimalidad posible de este tipo de ataque. Se basa en que la referencia falsa entra en juego como variable de manera que el atacante optimiza tanto su acción de control como el valor que debería tomar dicha referencia falsa para lograr la mayor reducción posible de su coste.

$$x^f_{a,ref}{}^{*} = \arg\min_{x^f_{a,ref}} \frac{1}{2}\mathbf{u}_a^T(k)\left(G_{ua}^T\widehat{Q}_a G_{ua} + \widehat{R}_a\right)\mathbf{u}_a(k) + \left((x_a(k|k) - x^f_{a,ref})^T G_{xa}^T \widehat{Q}_a G_{ua} + \mathbf{w}_a^{p-1}(k)^T G_{wa}^T \widehat{Q}_a G_{ua}\right)^T \mathbf{u}_a(k)$$

Caso particular. Problema sin restricciones.

Este apartado es simplemente una resolución analítica de la búsqueda de la referencia óptima. Para ello se asume que no hay restricciones impuestas lo cual permite encontrar los óptimos mediante derivaciones de las funciones dadas. Por su poca utilidad práctica no se muestra el resultado en este resumen, estando el procedimiento y expresión final en la memoria.

Falsos pesos

Este tipo de ataque se basa en la modificación de los valores originales que toman los λ_i en la función objetivo, de manera que el problema de optimización que se resuelve cause soluciones a favor del agente a.

En un principio se considera el caso en el que el atacante aumenta el valor de λ_a, lo que causa que su acción de control propuesta esté inclinada hacia el mismo.

Caso particular. Agente egoísta

Se expone aquí un caso particular del anterior en el cual no sólo el peso λ_a es alterado manteniendo el resto constate, sino que el agente malicioso anula de su problema de optimización totalmente el efecto del resto de la planta. De esta manera, a actúa sin ninguna intención de participar en el objetivo común de lograr una mejor situación para el sistema global. Este ataque implica que

$$\lambda_j = 0, \qquad \forall\, j \neq a$$
$$\lambda_a = 1$$

Falsas restricciones

Con el mismo fin que los casos anteriores, se busca que a través de un cambio en las restricciones originalmente impuestas se logre una mejora del coste local asociado al atacante.

Representación bidimensional

La finalidad de este capítulo es presentar de manera visual el fin y consecuencias de atacar. Esto se realiza empleando un horizonte de predicción $N = 1$ de manera que se puede plasmar en el plano u_1/u_2 lo que ocurre durante el procesos de negociación. Se incluye aquí como representación lo obtenido para ataques basados en falsa referencia. El resto de casos quedan expuestos en la memoria.

Indicadores de rendimiento para la evaluación de resultados

En este apartado se definen algunos ratios que permiten asociar al comportamiento mostrado gráficamente una serie de números que aportan información acerca del comportamiento del sistema, y por tanto, acerca del control que se realiza sobre el mismo. Entre estos cabe destacar el Precio de la Anarquía o el Precio de la Corrupción, siendo este último definido en este proyecto para una evaluación del daño realizado al introducir información falsa.

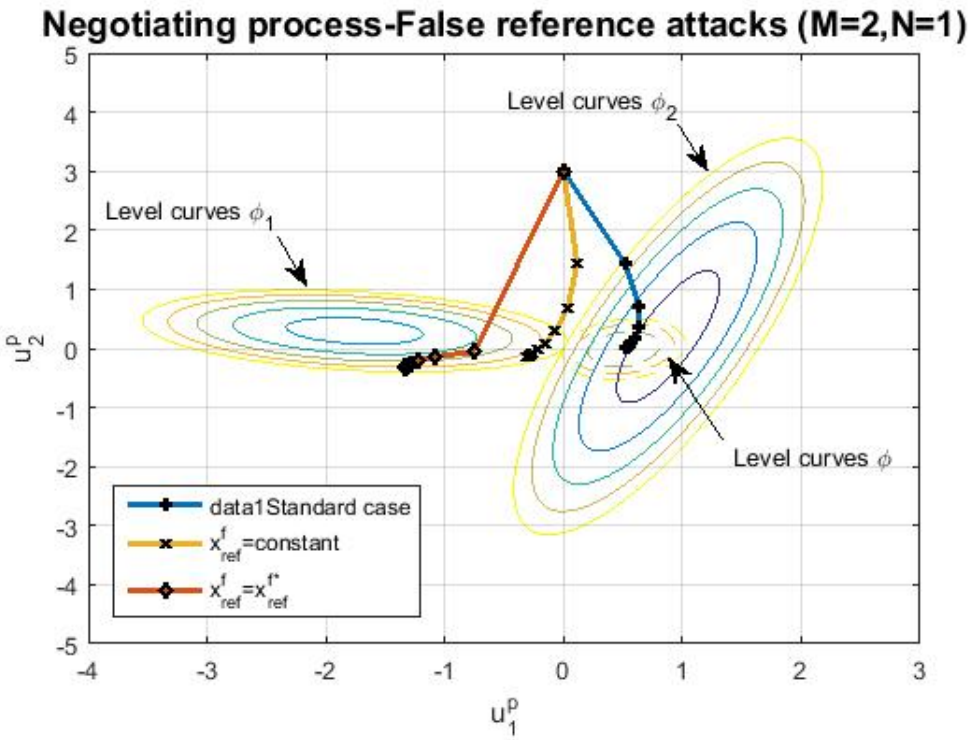

Figura 1: Trajectories in $u_1^p(k)$ / $u_2^p(k)$ for false reference attacks

La estrategia min-max

Para cerrar la presentación de posibles formas de amenazar la ciberseguridad del algoritmo, se expone una estrategia de defensa basada en min-max. La aplicación en este trabajo del min-max supone una propuesta muy conservadora, ya que sólo se reacciona ante ataques ante situaciones límite, lo cual conlleva la posibilidad de que un agente honesto permita que otro se aproveche de él mientras dicho límite no se alcance. En caso contrario, el agente honesto se saldrá del proceso de cooperación para resolver su propia situación local.

Capítulo 7: Ejemplo 1 - Dos dobles integradores con entradas aclopadas

El primer ejemplo presentado en este trabajo está formado por dos dobles integradores con dinámicas acopladas. Se considera en primer lugar el caso estándar de funcionamiento para una posterior introdución de los ataques. A modo de resumen se van a incluir aquí las gráficas correspondientes al coste acumulado y los cambios referentes a éste que se observan debido a los agentes maliciosos. Los parámetros empleados, así como el resto de figuras pueden verse en la memoria.

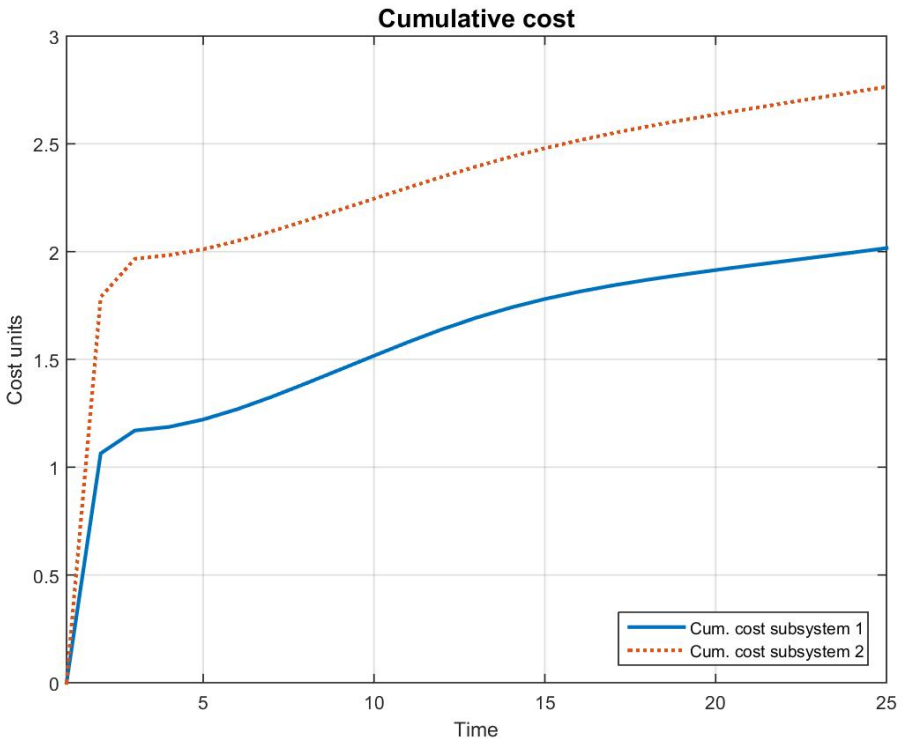

Figura 2: Costes acumulados para los subsistemas 1 y 2 (caso estándar)

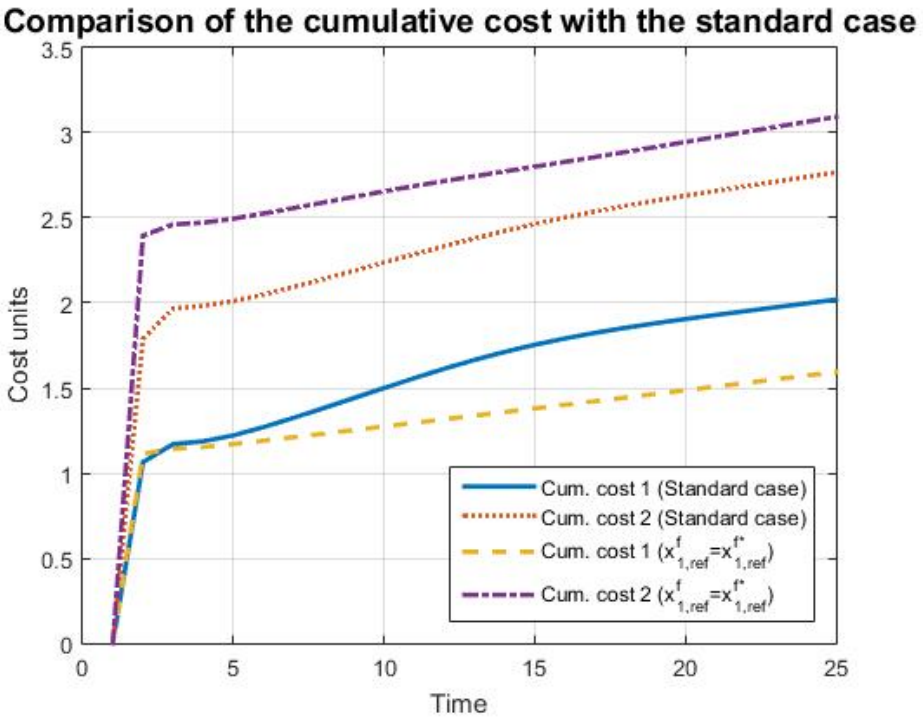

Figura 3: Comparación de los costes acumulados con el caso estándar para un ataque con $x^{f*}_{1,ref}$

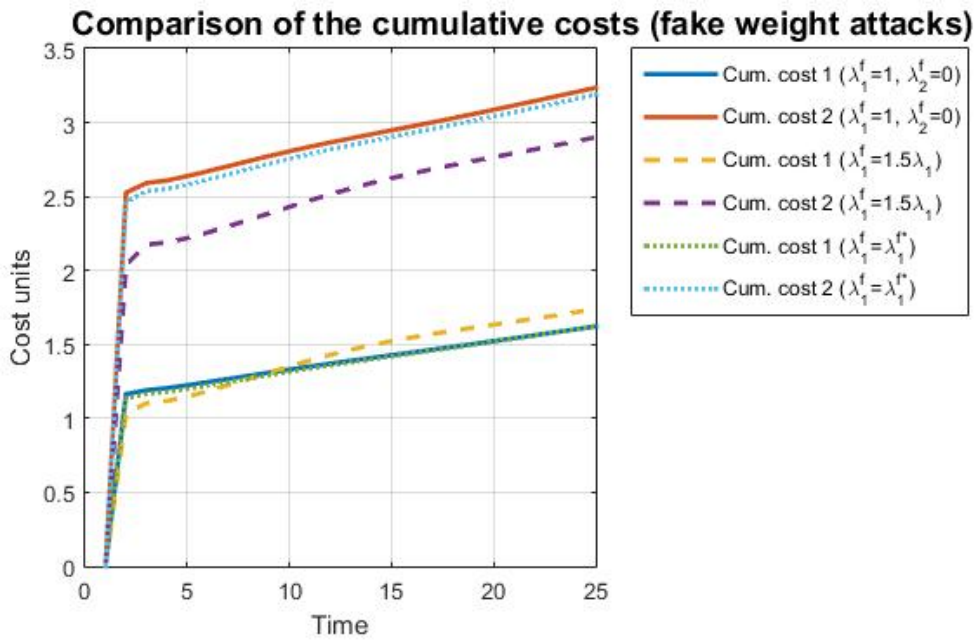

Figura 4: Comparación de los costes acumulados obtenidos con ataques basados en pesos falsos

Capítulo 8: Ejemplo 2 - Una planta de cuatro tanques

Se va a proceder de manera similar al anterior ejemplo, de manera que mostramos sólo las evoluciones de los costes acumulados a los que hemos llegado, como resumen del impacto de cada ataque aplicado. La definición de los parámetros utilizados están especificadas en el correspondiente capítulo de la memoria. En este ejemplo se incluye igualmente los resultados de haber implementado la estrategia min-max expuesta de manera teórica.

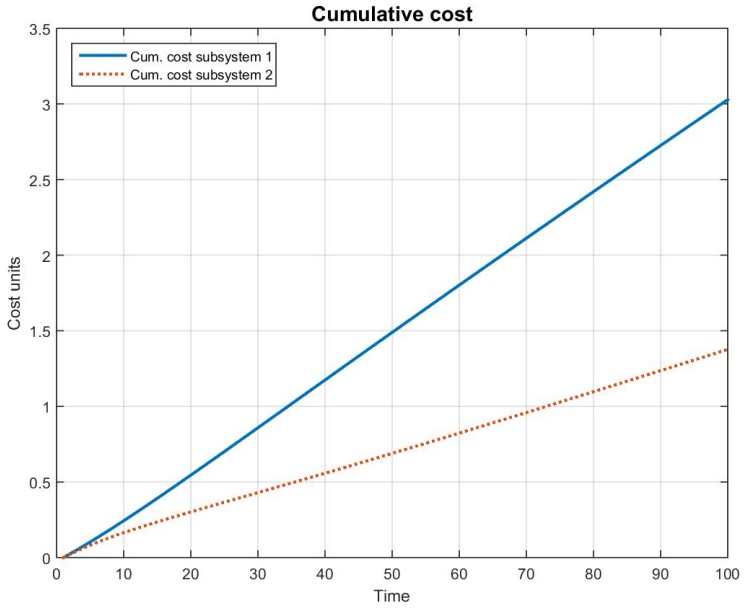

Figura 5: Costes acumulados para los subsistemas 1 y 2 (caso estándar)

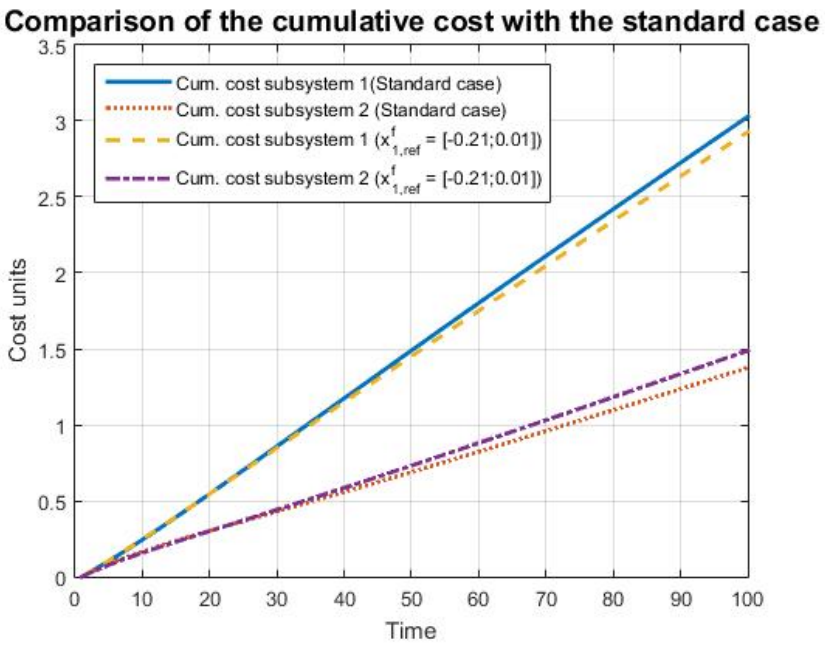

Figura 6: Comparación de los costes acumulados con caso estándar para un ataque basado en $x^f_{1,ref} = [-0{,}21; 0{,}01]$

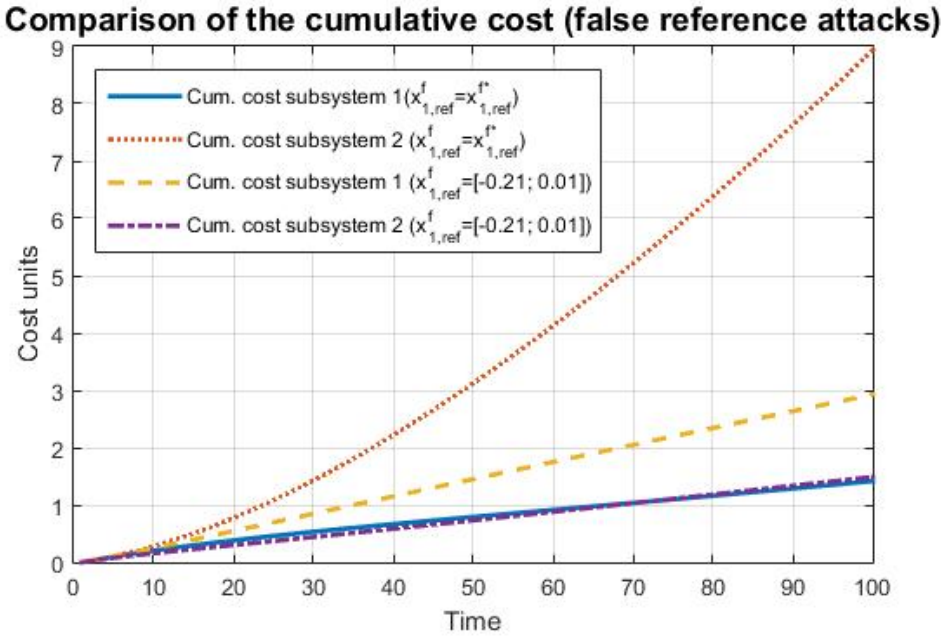

Figura 7: Comparación de los costes acumulados para ataques basados en falsas referencias

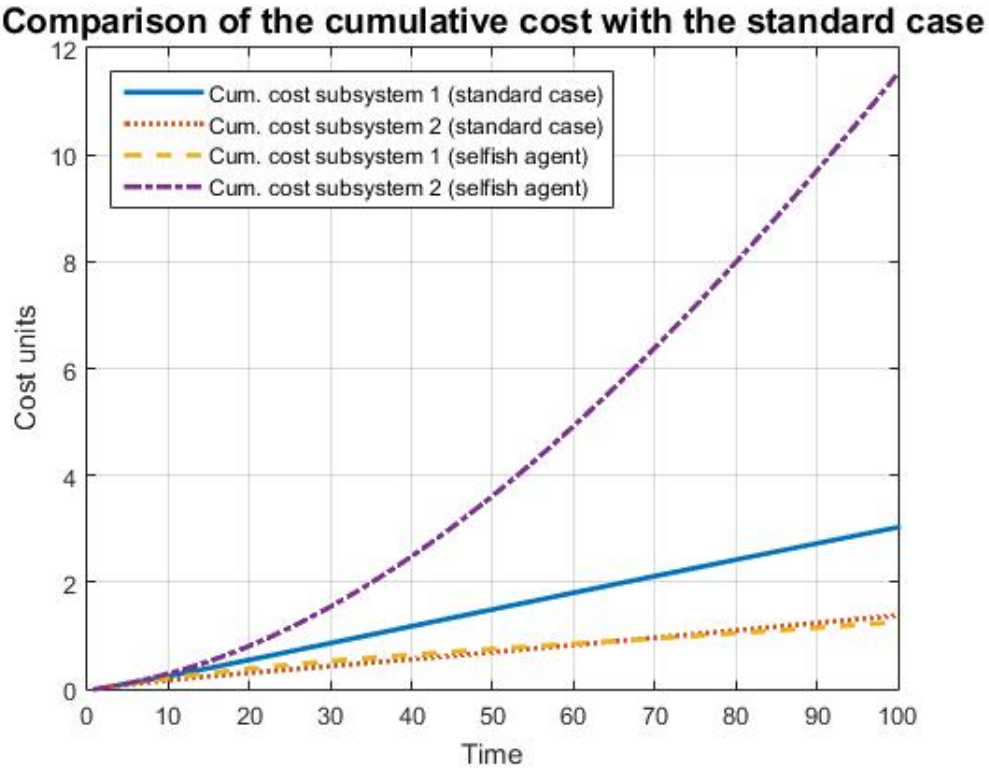

Figura 8: Comparación de los costes acumulados con el caso estándar para un ataque basado en agente egoísta

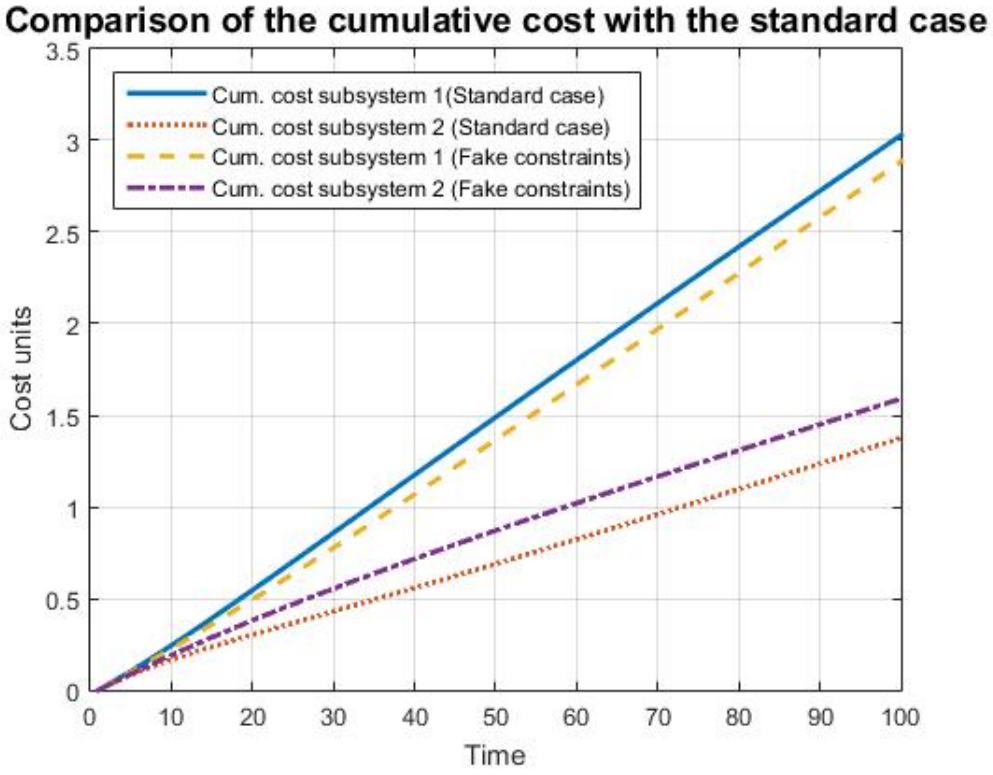

Figura 9: Comparación de los costes acumulados con el caso estándar para un ataque basado en restricciones falsas

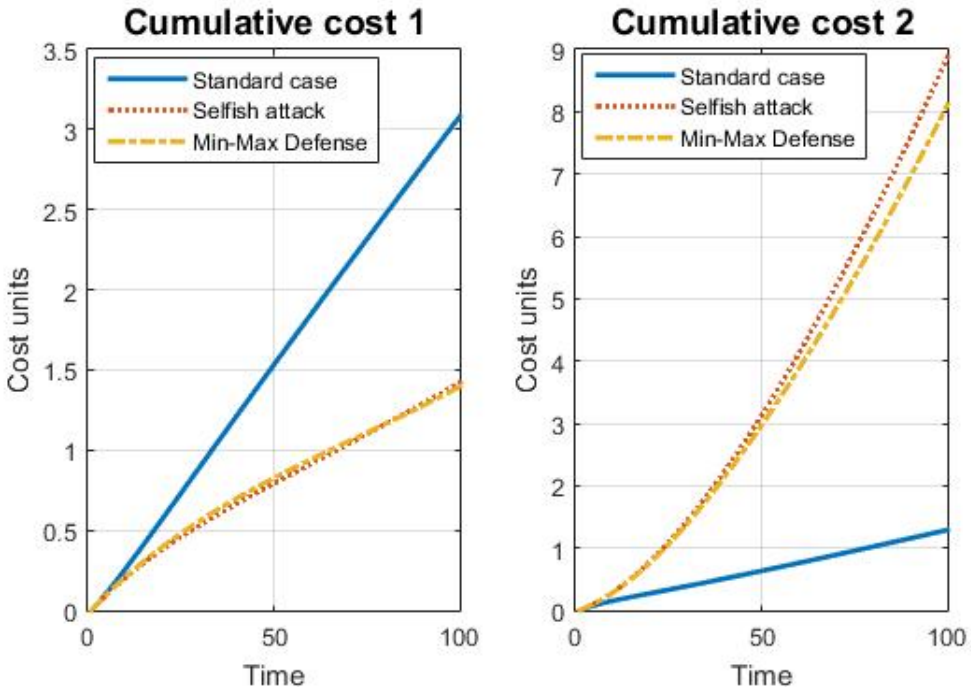

Figura 10: Resultado de introducir la estrategia min-max de defensa