

Optimización en Tiempo Real utilizando la Metodología de Adaptación de Modificadores

T. Rodríguez-Blanco ^{a,*}, D. Sarabia ^b, C. de Prada ^a

^a Departamento de Ingeniería de Sistemas y Automática, Universidad de Valladolid, C/Doctor Mergelina s/n, 47011, Valladolid, España.

^b Departamento de Ingeniería Electromecánica, Universidad de Burgos, Avda. Cantabria s/n, 09006, Burgos España.

Resumen

La gestión óptima de las plantas de proceso normalmente se lleva a cabo en una capa de optimización en tiempo real (Real Time Optimization, RTO) que actúa sobre la capa de control y que toma decisiones considerando objetivos económicos en base a un modelo del proceso, normalmente estacionario. Sin embargo, dicha operación óptima no está garantizada debido a la presencia de incertidumbre entre el modelo usado para la toma de decisiones y el proceso real. Con la idea de conducir el proceso a su punto de operación óptimo usando un modelo que se sabe incierto o erróneo, surge la metodología de adaptación de modificadores (Modifier Adaptation o MA). En dicha metodología, el problema de optimización económica de la capa RTO es modificado mediante unos términos correctores, conocidos como modificadores, estimados a partir de medidas de la planta, con el objetivo de conducir el proceso a su punto de operación óptimo. El presente artículo hace una revisión de las técnicas desarrolladas hasta el momento dentro de la metodología MA analizando sus características y modos de implementación.

Palabras Clave:

Optimización en tiempo real, incertidumbre, adaptación de modificadores

Real-Time optimization using the Modifier Adaptation methodology

Abstract

Optimal process operation is carried out by a Real-Time Optimization (RTO) layer which operates above the control layer and takes decisions based on steady-state plant models by considering economic objectives. However, this optimal operation is not guaranteed due to the presence of plant-model mismatch. To bring the process to the optimum operating point, the economic optimization problem solved in the RTO layer is changed following the Modifier Adaptation methodology (MA). This methodology changes the economic optimization problem solved in the RTO layer by adding some corrector terms or modifiers estimated from plant measurements to bring the process to the real optimum. This article presents a review of the different MA techniques developed until now and analyzing their features and the way to implement them.

Keywords:

Real-Time optimization, uncertainty, modifier-adaptation.

1. Introducción

La gestión de sistemas de gran escala es uno de los temas que ha atraído más interés en los últimos años debido a la importancia práctica que tiene en el buen funcionamiento de servicios y procesos básicos de nuestra sociedad. Nos referimos a infraestructuras como redes de agua, gas, tráfico aéreo, empresas de producción, etc. donde deben tomarse decisiones que satisfagan una serie de requisitos de operación complejos y restricciones en el valor de muchas variables y, además, ser óptimas desde un punto de vista de eficiencia, económico,

ambiental, etc. Esta toma de decisiones se aborda utilizando modelos de gran dimensión y métodos de optimización.

En la industria de procesos es habitual la existencia de una estructura jerárquica de control como la de la Figura 1 (Tatjewski, 2008). En particular, RTO consiste en una capa de optimización que opera sobre la capa de control y que toma decisiones en una escala de horas en base a un modelo del proceso real considerando explícitamente objetivos económicos (Marlin y Hrymak, 1997). El punto óptimo obtenido por la capa RTO es enviado a los controladores del nivel inferior que incluyen el control básico y normalmente controladores avanzados como el control predictivo basado en modelos (MPC).

*Autor para la correspondencia: tania.rodriguez@autom.uva.es

Sin embargo, la operación óptima del proceso no está garantizada ya que las decisiones se toman en base a modelos, los cuales nunca son perfectos ya que presentan tanto incertidumbre estructural, ya que son una aproximación del proceso, como paramétrica, debido a los parámetros inciertos. Por ello, el óptimo calculado con el modelo de la capa RTO puede no corresponder al punto óptimo de operación del proceso real o incluso ser un punto no factible. Además, las diferentes capas de la estructura de control usan modelos distintos, mientras que la capa RTO está basada en modelos estacionarios no lineales, la capa MPC utiliza modelos dinámicos lineales, de modo que podría existir una falta de coherencia entre los mismos que afecte al resultado final (Engell, 2006).

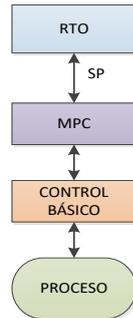


Figura 1: Estructura jerárquica de control típica.

En su artículo, Engell propone diversas formas de tratar la problemática debida a la interacción entre las diferentes capas de la estructura de control, como son aumentar la frecuencia de ejecución del RTO, para que la escala de tiempo de las distintas capas sea similar, o integrar optimización estacionaria en la capa MPC (a través de una capa intermedia de programación lineal LP) de modo que se utilice el mismo tipo de modelos en las distintas capas. Sin embargo, en este artículo se describirá la metodología MA que es capaz de tratar con este tipo de incertidumbre, además de con la incertidumbre planta-modelo, de una forma más sencilla corrigiendo únicamente el modelo estacionario de la capa RTO. Como ya se ha mencionado, normalmente, la capa RTO utiliza modelos estacionarios del proceso para la toma de decisiones, formulando el problema de optimización como sigue:

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) \\ \text{s.t.} \quad & g_i(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) \leq 0 \quad i = 1, \dots, n_g \\ & \mathbf{h}(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) = 0 \\ & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \end{aligned} \quad (1)$$

donde ϕ es la función de coste a minimizar, g_i representa las restricciones del modelo ($i = 1, \dots, n_g$), \mathbf{h} representa el conjunto de restricciones de igualdad impuestas por el propio modelo, \mathbf{y} el vector de variables de salida, $\boldsymbol{\beta} \in \mathcal{R}^{n_\beta \times 1}$ el vector de parámetros inciertos y $\mathbf{u} \in \mathcal{R}^{n_u \times 1}$ el vector columna que contiene las variables de decisión cuyo valor está limitado entre un valor inferior \mathbf{u}^L y otro superior \mathbf{u}^U . Estas variables de decisión pueden representar las referencias de una capa de control avanzado (ver Figura 1) o si ésta no existe, las consignas a los controladores básicos, normalmente PIDs. ϕ puede representar diferentes objetivos, por ejemplo, el coste de un proceso, eficiencia o emisiones de contaminantes. g_i suelen indicar límites de operación o especificaciones del producto obtenido.

Existen básicamente tres grupos de métodos para el tratamiento de la incertidumbre en la capa RTO dependiendo del uso que se haga de las medidas tomadas del proceso. El primero de ellos y más intuitivo consiste en usar las medidas de la planta para actualizar los parámetros del modelo y a continuación

utilizar este modelo corregido para llevar a cabo la optimización económica (Chen y Joseph, 1987). Esta formulación, conocida como el enfoque en “dos etapas”, se resuelve de forma iterativa hasta que converge a una solución supuestamente óptima, es decir, cada vez que se alcanza un estado estacionario del proceso se resuelven ambos problemas (de estimación y de optimización económica). Sin embargo, la presencia de incertidumbre estructural modelo-proceso hace que la metodología anterior no sea capaz de alcanzar el óptimo real del proceso. A partir de esta dificultad surge una variante del método en “dos etapas”, el segundo método, que utiliza las medidas del proceso para estimar los gradientes de la función de coste respecto a las variables de decisión y calcular unos modificadores de la función de coste del problema de optimización para forzar que las condiciones necesarias de optimalidad (NCO) del problema modificado coincidan con las del proceso real. De este modo se puede afirmar que el algoritmo converge a un punto de operación que satisface las NCO del proceso. El método descrito es conocido como ISOPE (Integrated System Optimization and Parameter Estimation) (Roberts, 1979) y es la base de la metodología MA. La técnica de MA utiliza las medidas del proceso para estimar los gradientes y calcular términos correctores de la función de coste y las restricciones del problema de optimización sin tener que actualizar los parámetros del modelo. La gran ventaja de MA es su probada capacidad para converger a un punto que cumpla las NCO del proceso a pesar de la presencia de incertidumbre estructural. Finalmente, el último grupo de métodos incorpora las medidas del proceso para actualizar directamente las entradas del proceso sustituyendo el problema de optimización por un problema de control por realimentación que pretende satisfacer las condiciones de optimalidad del proceso. Algunos ejemplos son Extremum-Seeking Control (ESC) (Krstic y Wang, 2000), Neighboring-Extremal Control (NEC) (François *et al.*, 2005) o Self-Optimizing Control (SOC) (Skogestad, 2000).

MA ha ido evolucionando desde sus inicios en los años 70, surgiendo numerosas variantes. Su implementación en varios casos de estudio, como el reactor Otto-Williams (Roberts, 1979), (Marchetti *et al.*, 2010) o una columna de destilación despropanizadora (Rodríguez-Blanco, *et al.*, 2015), ha demostrado que es una gran herramienta para optimizar la operación de procesos en presencia de un elevado grado de incertidumbre modelo-planta. Por ello, este artículo hace una revisión de las principales técnicas desarrolladas explicando sus características y el modo de implementación.

El artículo está organizado como se muestra a continuación. La sección 2 muestra las principales variantes de la metodología MA. La sección 3 presenta enfoques alternativos para solucionar los problemas que surgen en sistemas de gran dimensión, es decir, en presencia de un número elevado de entradas o restricciones del proceso. A continuación, en la sección 4, se plantea la dificultad de tener que esperar al estado estacionario del proceso para actualizar los modificadores y como poder evitarlo. La sección 5 presenta las principales aplicaciones en las que se ha implementado la metodología MA. Para finalizar, la sección 6 describe las principales conclusiones del artículo y presenta posibles temas de trabajo futuro.

2. Evolución de la metodología de adaptación de modificadores

2.1. Integrated System Optimization and Parameter Estimation (ISOPE)

Para poder tener en cuenta las discrepancias modelo-proceso surgió una variante del método de dos etapas denominada

ISOPE. La primera etapa consiste en resolver un problema de estimación de parámetros para que las salidas del proceso y del modelo coincidan. La segunda etapa resuelve un problema de optimización económica en el que se incluye un parámetro adicional en la función de coste que hace que el gradiente del modelo se iguale al del proceso con el objetivo de que las condiciones de optimalidad de primer orden (KKT) del problema (1) descritas por (2), coincidan con las del proceso real. Este modificador se calcula como la diferencia entre el gradiente del modelo y el del proceso real.

$$\begin{aligned}
 &g_i \leq 0 \quad \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \\
 &\mu_i g_i = 0 \quad \zeta^{U^T}(\mathbf{u} - \mathbf{u}^U) = 0 \quad \zeta^{L^T}(\mathbf{u}^L - \mathbf{u}) = 0 \\
 &\mu_i \geq 0 \quad \zeta^{U^T} \geq 0 \quad \zeta^{L^T} \geq 0 \\
 &\frac{\partial L}{\partial \mathbf{u}} = \frac{\partial \phi}{\partial \mathbf{u}} + \sum_{i=1}^{n_g} \mu_i^T \frac{\partial g_i}{\partial \mathbf{u}} + \zeta^U - \zeta^L = 0
 \end{aligned} \tag{2}$$

donde L es el Lagrangiano y μ , ζ son multiplicadores de Lagrange.

A continuación se muestra la metodología a seguir para llevar a cabo su implementación. Por simplicidad se considera un problema de optimización económica sin restricciones, aunque ISOPE también ha sido desarrollado e implementado en procesos con restricciones (Brdys et.al, 1986):

Algoritmo 1: ISOPE (Roberts, 1979)

Paso 1: Alcanzado un estado estacionario del proceso, se plantea la ejecución k de la capa RTO, con las entradas \mathbf{u}_{k-1} siendo aplicadas a la planta y midiendo las variables de salida $\mathbf{y}_p(\mathbf{u}_{k-1})$, se resuelve un problema de estimación de parámetros para actualizar los parámetros inciertos β_k , cuyos límites inferior y superior vienen dados por β^L y β^U respectivamente. Este problema (3) es resuelto bajo la condición de que las salidas calculadas con el modelo $\mathbf{y}(\mathbf{u}_{k-1}, \beta_k)$ coincidan con las medidas del proceso real $\mathbf{y}(\mathbf{u}_{k-1}, \beta_k) = \mathbf{y}_p(\mathbf{u}_{k-1})$, obteniéndose la solución óptima para los parámetros inciertos β_k^* .

$$\begin{aligned}
 &\min_{\beta} (\mathbf{y}_P - \mathbf{y})^T (\mathbf{y}_P - \mathbf{y}) \\
 &s.t \quad \mathbf{h}(\mathbf{u}_{k-1}, \mathbf{y}(\mathbf{u}_{k-1}, \beta)) = 0 \\
 &\quad \beta^L \leq \beta \leq \beta^U
 \end{aligned} \tag{3}$$

Paso 2: Suponiendo que el gradiente de la salida de la planta $\frac{\partial \mathbf{y}_P}{\partial \mathbf{u}}(\mathbf{u}_{k-1})$ está disponible, se calcula el modificador $\lambda_k \in \mathfrak{R}^{n_u}$ con información del estado estacionario del proceso (4):

$$\lambda_k^T = \frac{\partial \phi}{\partial \mathbf{y}}(\mathbf{u}_{k-1}, \mathbf{y}(\mathbf{u}_{k-1}, \beta_k^*)) \left[\frac{\partial \mathbf{y}_P}{\partial \mathbf{u}}(\mathbf{u}_{k-1}) - \frac{\partial \mathbf{y}}{\partial \mathbf{u}}(\mathbf{u}_{k-1}, \beta_k^*) \right] \tag{4}$$

Paso 3: Una vez que el modelo ha sido actualizado y el modificador está calculado se resuelve el problema de optimización económica modificado (5) para obtener las nuevas entradas \mathbf{u}^*_k . En este problema ϕ_M es la función de coste modificada. La solución de (5) se aplicará al proceso hasta la siguiente iteración del RTO que tendrá lugar cuando el proceso alcance un nuevo estado estacionario.

$$\begin{aligned}
 &\min_{\mathbf{u}} \phi_M = \phi(\mathbf{u}, \mathbf{y}(\mathbf{u}, \beta_k^*)) + \lambda_k^T \mathbf{u} \\
 &s.t \quad \mathbf{h}(\mathbf{u}, \mathbf{y}(\mathbf{u}, \beta_k^*)) = 0 \\
 &\quad \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U
 \end{aligned} \tag{5}$$

De este modo el gradiente de la función de coste modificada coincidirá con el gradiente de la planta en \mathbf{u}_k y la condición de cualificación del modelo presente en la literatura del método

ISOPE (Brdys y Roberts, 1987) se cumplirá, es decir, que en el óptimo las condiciones KKT del problema modificado sean iguales a las del proceso.

Paso 4(opcional): Si se quiere suavizar los cambios y mejorar la convergencia a un punto de operación que satisfaga las NCO de la planta es posible filtrar las entradas obtenidas en (5) mediante un filtro exponencial de primer orden (6), donde \mathbf{K} es un vector constante, que varía entre 0 y 1 y aplicar las entradas filtradas al proceso.

$$\mathbf{u}_k^* = \mathbf{u}_{k-1} + \mathbf{K}(\mathbf{u}_k^* - \mathbf{u}_{k-1}) \tag{6}$$

Paso 5: Comprobar si se cumple el criterio de convergencia, por ejemplo $|\mathbf{u}_k^* - \mathbf{u}_{k-1}^*| \leq \text{tol}$, es decir, que no haya cambios en las variables de decisión por encima de una tolerancia dada, y parar el algoritmo. Si esta condición no se satisface, $k = k+1$ y volver al paso 1 cuando se alcance el próximo estado estacionario del proceso.

Por tanto con esta metodología cada ejecución k de la capa RTO se corresponde con un estacionario del proceso.

2.2. Metodología de adaptación de modificadores (Modifier Adaptation methodology)

En el año 2002, Tatjewski demostró que la convergencia al óptimo del método ISOPE no depende de la estimación de parámetros inciertos sino de la igualdad entre las salidas del proceso y del modelo en cada ejecución del RTO (Tatjewski, 2002). Por esta razón, introdujo un nuevo modificador $\mathbf{b}_k \in \mathfrak{R}^{n_y}$ que tiene en cuenta la diferencia entre las salidas del proceso y del modelo asegurando que bajo convergencia estos valores coincidirán. El problema de optimización resultante que será resuelto por la capa RTO se muestra en (7), donde ϕ_M es la función de coste modificada, \mathbf{u}^*_{k-1} es la entrada aplicada en el estado estacionario anterior, es decir, la solución óptima del anterior RTO, \mathbf{h} representa el conjunto de restricciones de igualdad impuestas por el propio modelo y el subíndice P indica que tanto el gradiente de la función de coste como las salidas son evaluadas a partir de medidas del proceso.

$$\begin{aligned}
 &\min_{\mathbf{u}} \phi_M = \phi(\mathbf{u}, \mathbf{y} + \mathbf{b}_k, \beta) + \lambda_k^T \mathbf{u} \\
 &s.t \quad \mathbf{h}(\mathbf{u}, \mathbf{y} + \mathbf{b}_k, \beta) = 0 \\
 &\quad \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U
 \end{aligned} \tag{7}$$

donde

$$\lambda_k^T = \frac{\partial \phi_P}{\partial \mathbf{u}} \Big|_{\mathbf{u}^*_{k-1}} - \frac{\partial \phi}{\partial \mathbf{u}} \Big|_{\mathbf{u}^*_{k-1}} \quad \mathbf{b}_k = \mathbf{y}_P(\mathbf{u}^*_{k-1}) - \mathbf{y}(\mathbf{u}^*_{k-1}) \tag{8}$$

Siguiendo esta idea, Gao y Engell definieron nuevos modificadores para las restricciones del proceso (Gao y Engell, 2005) formulando el problema de optimización (9). La principal contribución de este trabajo es el modo en el que se corrigen las restricciones del proceso. Además del bias corrector ε_k definido previamente (Forbes y Marlin, 1994) que resulta de la diferencia entre el valor de la restricción evaluada con medidas del proceso $g_{P,i}$ y evaluada con el modelo g_i , se incluyen términos correctores del gradiente de las restricciones. Estos modificadores

$\gamma_k \in \mathfrak{R}^{n_u \times n_g}$ se calculan como la diferencia entre los gradientes de las restricciones estimados con el modelo y los del proceso real (10). De este modo, bajo convergencia, se asegura que se alcanza un punto de operación que satisface las NCO del proceso real. En este caso $g_{M,i}$ son las restricciones modificadas, $g_{P,i}$ es el valor de las restricciones evaluadas con medidas del proceso y g_i las restricciones originales del modelo.

$$\begin{aligned}
 \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \mathbf{y} + \mathbf{b}_k, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}^*_{k-1}) \\
 \text{s.t} \quad & \mathbf{h}(\mathbf{u}, \mathbf{y} + \mathbf{b}_k, \boldsymbol{\beta}) = 0 \\
 & g_{M,i} = g_i(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) + \boldsymbol{\gamma}_{k,i}^T (\mathbf{u} - \mathbf{u}^*_{k-1}) + \varepsilon_{i,k} \leq 0 \\
 & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U
 \end{aligned} \tag{9}$$

donde

$$\boldsymbol{\lambda}_k^T = \left. \frac{\partial \phi_P}{\partial \mathbf{u}} \right|_{\mathbf{u}^*_{k-1}} - \left. \frac{\partial \phi}{\partial \mathbf{u}} \right|_{\mathbf{u}^*_{k-1}} \quad \boldsymbol{\gamma}_{k,i}^T = \left. \frac{\partial g_{P,i}}{\partial \mathbf{u}} \right|_{\mathbf{u}^*_{k-1}} - \left. \frac{\partial g_i}{\partial \mathbf{u}} \right|_{\mathbf{u}^*_{k-1}} \tag{10}$$

$$\varepsilon_{i,k} = g_{P,i}(\mathbf{u}^*_{k-1}) - g_i(\mathbf{u}^*_{k-1}) \quad \mathbf{b}_k = \mathbf{y}_P(\mathbf{u}^*_{k-1}) - \mathbf{y}(\mathbf{u}^*_{k-1})$$

A partir de igualar las condiciones KKT del proceso y las del modelo surgió la metodología de adaptación de modificadores o Modifier Adaptation, presentada por Chachuat en el año 2009 (Chachuat *et al.*, 2009) y formalizada por Marchetti (Marchetti *et al.*, 2009). A partir del problema (9) y (10) los autores eliminaron el modificador \mathbf{b}_k justificando que con el resto de los modificadores, bajo convergencia, las condiciones KKT del problema modificado coincidirán con las del proceso. El problema (11) se resuelve iterativamente implementando las entradas óptimas al proceso \mathbf{u}^*_k y actualizando los modificadores una vez que se ha alcanzado el estado estacionario como muestra la Figura 2:

$$\begin{aligned}
 \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) + \boldsymbol{\lambda}_k^T (\mathbf{u} - \mathbf{u}^*_{k-1}) \\
 \text{s.t} \quad & g_{M,i} = g_i(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) + \boldsymbol{\gamma}_k^T (\mathbf{u} - \mathbf{u}^*_{k-1}) + \varepsilon_{i,k} \leq 0 \\
 & \mathbf{h}(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) = 0 \\
 & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \\
 & i = 1, \dots, n_g
 \end{aligned} \tag{11}$$

donde los modificadores se calculan como se indicó en (10). Por otro lado, en esta formulación, el número de modificadores necesarios (n_{mod}) viene dado por (12) donde se observa que depende tanto del número de variables de decisión (n_u) como del número de restricciones (n_g):

$$n_{mod} = n_g + n_u(n_g + 1) \tag{12}$$

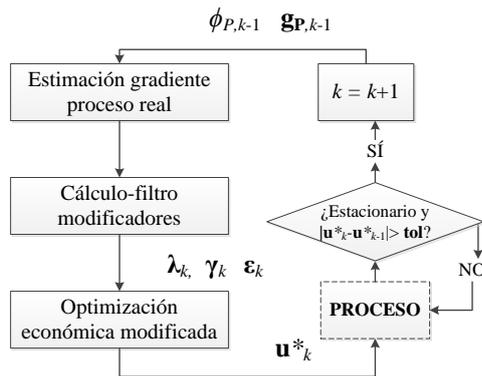


Figura 2: Formulación general metodología MA

Para que un modelo del proceso sea adecuado para su uso en la capa RTO se debe satisfacer el criterio de adecuación del modelo. Este criterio afirma que \mathbf{u}_p^* , es decir, el óptimo del proceso, debe satisfacer las NCO de primer y segundo orden para el problema de optimización modificado (11), con los modificadores evaluados en \mathbf{u}_p^* . Como MA se encarga de hacer que las condiciones KKT de primer orden coincidan con las de la planta, sólo quedan por ajustar las NCO de segundo orden. Por ello, el Hessiano reducido del Lagrangiano del problema (11) debe ser positivo definido en \mathbf{u}_p^* .

Este algoritmo tiene la ventaja de que no es necesario conocer el conjunto de restricciones activas a priori (algo que no ocurre cuando se aplican métodos como ESC o NEC), añadiendo tantos modificadores $\boldsymbol{\gamma}_k$ como restricciones de operación del proceso a costa de incrementar el tiempo de convergencia del método. Por otro lado el método consigue alcanzar un punto que satisfice las NCO incluso en presencia de incertidumbre estructural. El principal inconveniente es que aunque el método converja al óptimo de la planta puede hacerlo siguiendo un camino no factible, ya que no existe garantía de factibilidad en las ejecuciones intermedias, de ahí la importancia del uso de filtros en los modificadores, como el mostrado en (13), para suavizar los cambios y seguir un camino factible hacia el óptimo.

$$\begin{aligned}
 \boldsymbol{\lambda}_k &= (\mathbf{I} - \mathbf{K}_\lambda) \boldsymbol{\lambda}_{k-1} + \mathbf{K}_\lambda \boldsymbol{\lambda}_k^T \\
 \boldsymbol{\gamma}_k &= (\mathbf{I} - \mathbf{K}_\gamma) \boldsymbol{\gamma}_{k-1} + \mathbf{K}_\gamma \boldsymbol{\gamma}_k^T \\
 \boldsymbol{\varepsilon}_k &= (\mathbf{I} - \mathbf{K}_\varepsilon) \boldsymbol{\varepsilon}_{k-1} + \mathbf{K}_\varepsilon \boldsymbol{\varepsilon}_k^T
 \end{aligned} \tag{13}$$

donde \mathbf{K}_λ , \mathbf{K}_γ y \mathbf{K}_ε son matrices diagonales con autovalores en el intervalo (0,1]. En general, MA puede implementarse siguiendo los siguientes pasos:

Algoritmo 2: Modifier Adaptation (Marchetti *et al.*, 2009)

- Paso 1:** Inicializar $k = 0$ y los modificadores $\boldsymbol{\lambda}_0$, $\boldsymbol{\gamma}_0$ y $\boldsymbol{\varepsilon}_0$.
- Paso 2:** Una vez que el proceso haya alcanzado el estado estacionario, medir la función de coste del proceso $\phi_{P,k-1}$ y de las restricciones $\mathbf{g}_{P,k-1}$. Calcular el modificador de orden cero $\varepsilon_{i,k}$ según la expresión (10) para cada restricción.
- Paso 3:** Estimar los gradientes del proceso para el cálculo de los modificadores de primer orden. Este paso será diferente en cada metodología, ya que cada una considera un método de estimación distinto, como se verá a lo largo del artículo.
- Paso 4 (opcional):** Filtrar el valor de los modificadores dados por la capa superior de optimización usando la expresión (13).
- Paso 5:** Calcular la nueva variable de decisión \mathbf{u}^*_k que será aplicada al proceso hasta el siguiente estado estacionario resolviendo el problema de optimización modificado (11).
- Paso 6:** Comprobar si se satisface el criterio de convergencia, por ejemplo $|\mathbf{u}^*_k - \mathbf{u}^*_{k-1}| \leq \text{tol}$, es decir, que no haya cambios en las variables de decisión por encima de una tolerancia dada, y parar el algoritmo si es así. Pueden considerarse otros criterios de parada como que no haya cambios en la función de coste mayores que una tolerancia $|\phi_{P,k}^* - \phi_{P,k-1}^*| \leq \text{tol}$. Si esta condición no se satisface, $k = k+1$ y volver al paso 2 cuando se alcance el próximo estado estacionario del proceso.

Como se observa en la Figura 2 esta metodología requiere el cálculo de gradientes del modelo y gradientes experimentales. Mientras que hay numerosas herramientas para calcular los gradientes del modelo con exactitud, los gradientes de la planta son más difíciles de estimar. Debido a la necesidad de disponer de los gradientes del proceso varios métodos han surgido para poder estimarlos adecuadamente, paso 3, que se combinan con la metodología general MA del Algoritmo 2. El método más intuitivo de estimación de gradientes de la planta consistiría en dar un salto en cada variable manipulada, esperar al estacionario del proceso y por diferencias finitas obtener cada gradiente. Sin embargo, esto requiere un elevado número de experimentos intermedios para estimar todos los gradientes (n_u experimentos) antes de cada ejecución del RTO, lo que hace este tipo de estimación inviable en la práctica.

2.3. Dual Modifier Adaptation (DMA)

El método de adaptación de modificadores dual o Dual Modifier Adaptation (DMA) propuesto inicialmente por Brdys y

Tatjewski (M. Brdys & Tatjewski, 1994; M. Brdys & Tatjewski, 2005) y llamado Dual ISOPE es una versión de MA en la que a través de añadir una restricción extra (restricción dual) se garantiza que las acciones de control pasadas aplicadas al proceso han generado una excitación suficiente en el proceso para estimar los gradientes experimentales adecuadamente. Estos gradientes se estiman incluyendo información de puntos de operación pasados, generados por RTO previas, usando la definición de derivada direccional. Suponiendo que existe información de $n_u + 1$ soluciones $\mathbf{u}_k, \mathbf{u}_{k-1}, \dots, \mathbf{u}_{k-n_u}$ de iteraciones de RTO previas, el vector de diferencias \mathbf{s}_{ki} con respecto a iteraciones previas se define como (14):

$$\mathbf{s}_{ki} = \mathbf{u}_{k-i} - \mathbf{u}_k \quad \forall i = 1 \dots n_u \quad (14)$$

Siendo los vectores \mathbf{s}_{ki} linealmente independientes se puede construir una matriz cuadrada no singular $\mathbf{S}_k \in \mathcal{R}^{n_u \times n_u}$ formada por los vectores de diferencias de las variables de decisión con respecto a n_u períodos anteriores.

$$\mathbf{S}_k = \begin{bmatrix} \mathbf{s}_{k1} & \dots & \mathbf{s}_{kn_u} \end{bmatrix}^T \quad (15)$$

Se obtienen los gradientes del proceso a partir de la definición de derivada direccional como muestra (16) donde la variable z puede representar tanto la función de coste del proceso ϕ_p como las restricciones $g_{p,i}$ evaluadas en puntos de operación anteriores:

$$\frac{\partial z}{\partial \mathbf{u}} \approx \frac{dz}{d\mathbf{u}} = (\mathbf{S}_k)^{-1} \begin{bmatrix} z(\mathbf{u}_{k-1}) - z(\mathbf{u}_k) \\ \vdots \\ z(\mathbf{u}_{k-n_u}) - z(\mathbf{u}_k) \end{bmatrix} \quad (16)$$

Para asegurar que los gradientes se obtienen con exactitud, una nueva restricción es añadida al problema de optimización ($\delta^{-1}(\mathbf{S}_k) \geq \delta_L$), donde δ representa el número de condición de la matriz \mathbf{S}_k definida en (15) y δ_L el parámetro que indica el mínimo grado de excitación del proceso (Gao et al., 2015a). Ésta restricción representa la característica dual del método: mientras el resto de la optimización intenta converger al óptimo del modelo modificado (primal objective), la restricción dual garantiza que en la siguiente iteración del RTO el sistema tendrá suficiente energía para estimar el gradiente del proceso adecuadamente (objetivo dual). La consecuencia de añadir esta restricción es que la región factible se reduce, lo que podría implicar una pérdida de optimalidad en la solución del RTO con respecto al problema original modificado (11). El problema de optimización modificado viene dado por (17). Por simplicidad, las ecuaciones de igualdad $\mathbf{h}(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) = 0$ han sido omitidas de la formulación ya que su efecto se incluye en las salidas $\mathbf{y} = \mathbf{h}(\mathbf{u}, \boldsymbol{\beta})$:

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) + \lambda_k^T (\mathbf{u} - \mathbf{u}_{k-1}^*) \\ \text{s.t.} \quad & g_{M,i} = g_i(\mathbf{u}, \mathbf{y}, \boldsymbol{\beta}) + \gamma_k (\mathbf{u} - \mathbf{u}_{k-1}^*) + \varepsilon_{i,k} \leq 0 \\ & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \\ & i = 1, \dots, n_g \\ & \delta(\mathbf{S}_k) \geq \delta_L \end{aligned} \quad (17)$$

La implementación de DMA se lleva a cabo siguiendo el algoritmo 2 sustituyendo los pasos 1, 3 y 5 por los siguientes:

Algoritmo 3: Dual Modifier Adaptation (M. Brdys & Tatjewski, 1994)

Paso 1: Ejecutar $n_u + 1$ estados estacionarios previos para disponer de la información necesaria para inicializar el cálculo de los gradientes de la planta y por lo tanto el valor inicial de los modificadores λ_0, γ_0 y ε_0 . Inicializar $k = 0$.

Paso 3: Estimar los gradientes del proceso a partir de (16).

Paso 5: Seleccionar el parámetro de sintonía δ_L y calcular la nueva variable de decisión \mathbf{u}^*_k que será aplicada al proceso hasta

el siguiente estado estacionario resolviendo el problema de optimización modificado (17)

La principal ventaja de este algoritmo es que los gradientes del proceso se calculan a partir de información de RTO previas por lo que no requiere excitación extra como por ejemplo en el caso de estimar los gradientes por diferencias finitas. Sin embargo, la evolución del algoritmo es muy sensible al parámetro δ_L cuyo valor no es fácil de conocer a priori. Dicho parámetro de sintonía debe seleccionarse cuidadosamente ya que un valor alto implica una mayor excitación del proceso pudiendo provocar que el proceso siga un camino no factible hacia el óptimo pero, en cambio, un valor pequeño, puede provocar que los gradientes no se calculen correctamente.

2.4. Iterative gradient-modification optimization (IGMO)

Uno de los mayores retos en la práctica es la estimación de gradientes experimentales a partir de medidas con ruido. IGMO explora la suavidad inherente del mapeo de la planta para permitir una optimización fiable y eficiente incluso si los datos presentan ruido. Este método calcula los gradientes de la planta a partir de datos recogidos en puntos de operación anteriores del mismo modo que lo hace DMA (16). En este caso, cuando la restricción dual (17) no se satisface se calcula una perturbación adicional \mathbf{u}^{addi} maximizando la inversa del número de condición \mathbf{S}_k^a sujeto a las restricciones del proceso modificadas.

$$\max_{\mathbf{u}^{addi}} \quad \kappa^{-1}(\mathbf{S}_k^a) \quad (18)$$

$$\text{s.t.} \quad g_{M,i}(\mathbf{u}^{addi}) = g_i(\mathbf{u}^{addi}) + \gamma_k (\mathbf{u}^{addi} - \mathbf{u}_{k-1}) + \varepsilon_{i,k} \leq 0$$

donde

$$\mathbf{S}_k^a = \begin{bmatrix} (\mathbf{u}_k - \mathbf{u}^{addi}) \dots (\mathbf{u}_k - \mathbf{u}_{k-n_u+1}) \end{bmatrix}^T \quad (19)$$

Para proveer una convergencia fiable en presencia de ruido, IGMO calcula los gradientes a partir de aproximaciones cuadráticas. Cómo elegir el conjunto de datos para la regresión U_k y la evaluación del espacio de búsqueda B_k se discuten en detalle en (Gao et al., 2015a). $\rho_{m,k}$ y $\rho_{q,k}$ son las exactitudes de predicción de los mapeos basados en el modelo y las funciones cuadráticas (ver (Gao et al., 2015a) para conocer cómo se calculan estos valores). ϕ_q y $g_{q,i}$ son la función objetivo y restricciones de la aproximación cuadrática. La implementación de este método se lleva a cabo siguiendo el algoritmo 4.

Algoritmo 4: Iterative gradient-modification optimization (Gao et al., 2015a)

Paso 1: Elegir un punto inicial \mathbf{u}_0 y aplicar al proceso ese punto y $\mathbf{u}_0 + c \mathbf{e}_i$, siendo c un parámetro de paso adecuado y \mathbf{e}_i ($i=1, \dots, n_u$) vectores unitarios ortogonales. Calcular los gradientes en \mathbf{u}_0 mediante diferencias finitas y aplicar IGMO hasta que se generen $(n_u + 1)(n_u + 2)/2$ set points. Obtener el conjunto de datos U_k (ver (Gao et al., 2015a)). Inicializar $\rho_{m,k}$ y $\rho_{q,k}$ en 0.

Paso 2: Calcular las funciones cuadráticas ϕ_q y $g_{q,i}$ y determinar el espacio de búsqueda B_k .

Paso 3: Extraer los gradientes de las funciones cuadráticas. Adaptar el problema de optimización y calcular $\hat{\mathbf{u}}_k$ como sigue:

a) Si $\rho_{m,k} \leq \rho_{q,k}$ resolver el problema de optimización modificado (17) sujeto a la restricción $\hat{\mathbf{u}}_k \in B_k$.

b) Si no es así, realizar una optimización basada en la aproximación cuadrática como muestra (20).

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_{q,k}(\mathbf{u}) \\ \text{s.t.} \quad & g_{q,i,k}(\mathbf{u}) \leq 0 \end{aligned} \quad (20)$$

$$\hat{\mathbf{u}}_k \in B_k$$

Paso 4: Evaluar el proceso en $\hat{\mathbf{u}}_k$ para obtener ϕ_q y $g_{q,i}$. La siguiente iteración se define en función de las siguientes condiciones:

- a) Iteración exitosa. Si $\hat{\phi}_{P,k} < \phi_{P,k}$, definir $\hat{\mathbf{u}}_{k+1} = \hat{\mathbf{u}}_k$ y definir el siguiente conjunto de regresión U_{k+1} . Actualizar los índices de calidad $\rho_{m,k+1}$ y $\rho_{q,k+1}$. Comprobar el criterio de terminación, si no se satisface, fijar $k = k+1$ y volver al paso 2.
- b) Iteración no exitosa. Si $\hat{\phi}_{P,k} \geq \phi_{P,k}$ actualizar el conjunto de datos incluyendo $\hat{\mathbf{u}}_k$. Volver al paso 2.

2.5. *Nested Modifier Adaptation (NMA)*

Aunque las técnicas MA basadas en el cálculo de gradientes garanticen que un punto de operación que satisface las NCO del proceso se alcanza independientemente del método utilizado para estimar dichos gradientes, lo cierto es que en aplicaciones reales, la estimación de dichos gradientes puede ser un problema, bien porque sean caros de obtener, por ejemplo en procesos interconectados donde las perturbaciones necesarias para su obtención pueden afectar a otras unidades, o bien porque su estimación no sea lo suficientemente exacta, como ocurre en la presencia de ruido. Para evitar dicho cálculo surgió una formulación llamada Nested Modifier Adaptation (NMA) (Navia et al., 2015). Éste método usa una arquitectura anidada con un algoritmo de optimización no basado en gradientes, como el Nelder- Mead, para actualizar los modificadores, iterando con ellos sobre el problema de optimización modificado hasta que se alcanza un punto que cumple las NCO del proceso. De este modo, la estimación del gradiente del proceso real se sustituye por otro método que tiene en cuenta la minimización de la función Lagrangiana medida directamente de la planta. Los pasos a seguir en la implementación de NMA son los descritos en el algoritmo 2 sustituyendo el paso 3 de estimación de gradientes por el que se describe a continuación. Esta implementación se presenta gráficamente en la Figura 3.

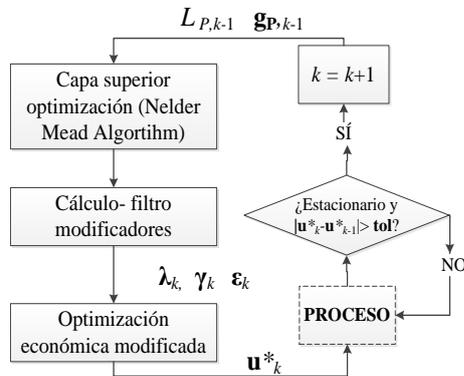


Figura 3: Esquema de implementación NMA

Algoritmo 5. Nested Modifier Adaptation (Navia et al., 2015)

Añadir a paso 1: Ejecutar n_n+1 estados estacionarios para calcular los gradientes de la planta por diferencias finitas y por lo tanto el valor inicial de los modificadores λ_0 , γ_0 y ϵ_0 . A continuación, el algoritmo Nelder-Mead requiere $n_{mod}+1$ estados estacionarios para construir el simplex inicial cuyos vértices corresponden a diferentes valores de los modificadores obtenidos a partir de los iniciales λ_0 , γ_0 y ϵ_0 (cambiando uno de los modificadores en cada vértice, ya sea sumando un término o multiplicándolo por un factor).

Paso 3: La capa superior de optimización sin restricciones actualiza el valor de los modificadores de primer orden λ_k y γ_k con el objetivo de minimizar la función Lagrangiana medida del proceso resolviendo (21).

$$\min_{\lambda_k, \gamma_k} L_{P,k-1} = \phi_{P,k-1} + \sum_{i=1}^{n_g} \mu_{k-1,i} g_{P,i,k-1} \tag{21}$$

La principal ventaja de NMA es que no requiere el cálculo de gradientes experimentales que puede resultar bastante complicado en la práctica, y además es menos sensible al ruido que otros métodos como DMA precisamente porque no requiere el cálculo de gradientes. Sin embargo, este algoritmo necesita una buena estimación inicial de los modificadores, si no, el camino hacia el óptimo podría no ser muy directo y seguir una trayectoria no factible. Además, el uso del algoritmo Nelder Mead en la capa superior de optimización presenta ciertos inconvenientes como son la elección de muchos parámetros de sintonía del propio algoritmo que pueden afectar a la convergencia hacia el óptimo y que cada una de sus iteraciones puede requerir varios estados estacionarios. Finalmente, cuando el número de modificadores es elevado debido a la presencia de numerosas variables de decisión o restricciones operacionales, la convergencia hacia el óptimo será muy lenta puesto que el número de variables de decisión de la capa superior de optimización aumenta considerablemente y su convergencia hacia el óptimo se ralentiza. Para resolver este problema una formulación del método NMA se presenta en el apartado 3.2.

3. MA en problemas de gran dimensión

Una de las desventajas principales de las metodologías descritas anteriormente está relacionada con la dimensión del problema, ya que el número de modificadores necesarios depende tanto del número de variables de decisión como del número de restricciones, como muestra (12).

La estimación de gradientes experimentales es la tarea más complicada a la hora de aplicar la metodología MA. Este difícil trabajo experimental aumenta con el número de variables de decisión, llegando a ser prácticamente imposible la implementación de MA en procesos con muchas entradas. Por ejemplo, en el caso del método DMA un elevado número de variables de decisión implica un mayor número de estados estacionarios previos para inicializar el algoritmo, lo que ralentiza la convergencia del método. Además, todos los cambios aplicados en las entradas del proceso en períodos anteriores deben ser linealmente independientes, es decir, deben explorarse todas las direcciones en el espacio de entradas de la planta por lo que DMA no puede conducir el proceso directamente hacia el óptimo sino que convergerá lentamente excitando todas las entradas del proceso. Para tratar este problema surgió una nueva metodología llamada MA Direccional (D-MA) que obtiene las derivadas del proceso solo en ciertas direcciones privilegiadas. Por otro lado, para el caso en el que exista un número elevado de restricciones una alternativa que se describe más adelante es usar la metodología NMA, con la que no es necesario calcular los gradientes experimentales, añadiendo únicamente un modificador por cada entrada del proceso en la función de coste, independientemente del número de restricciones. Dichos modificadores vendrán dados por la diferencia entre el gradiente del Lagrangiano del proceso y del modelo.

3.1. *Directional Modifier Adaptation (D-MA)*

La estimación de gradientes constituye el principal cuello de botella en la implementación de MA ya que el número de experimentos necesarios para estimarlos aumenta linealmente con el número de entradas del proceso, llegando a no poder aplicar MA en procesos con muchas entradas. D- MA pretende superar esta limitación estimando los gradientes solo en $n_r < n_u$ direcciones privilegiadas. De esta manera, la convergencia hacia el punto de operación que satisfaga las NCO de la planta se

acelera ya que se requieren menos experimentos para estimar los gradientes del proceso en cada iteración del RTO. D-MA define $\mathbf{U}_r = [\delta\mathbf{u}_1 \dots \delta\mathbf{u}_r]$ como una matriz ($n_u \times n_r$), cuyas columnas contienen las n_r direcciones privilegiadas en el espacio de entradas en las cuales se evaluará la derivada direccional de la función de coste y de las restricciones en el punto de operación \mathbf{u}_k (Costello et al., 2016). La derivada direccional puede considerarse un gradiente parcial que contiene información de cómo una función varía (localmente) en ciertas direcciones del espacio de entradas de la función y viene definida por (22):

$$\nabla_{\mathbf{U}_r} j_P := \left. \frac{\partial j_P(\mathbf{u}_k + \mathbf{U}_r \mathbf{r})}{\partial \mathbf{r}} \right|_{\mathbf{r}=\mathbf{0}} \quad j_P \in \left\{ \phi_P, g_{P,1}, g_{P,2}, \dots, g_{P,n_g} \right\} \quad (22)$$

donde $r \in \mathbb{R}^{n_r}$. Las aproximaciones de los gradientes del proceso vienen dados por (23):

$$\begin{aligned} \widehat{\nabla} \phi_P(\mathbf{u}_k) &= \frac{\partial \phi}{\partial \mathbf{u}}(\mathbf{u}_k)(\mathbf{I}_{n_u} - \mathbf{U}_r \mathbf{U}_r^+) + \nabla_{\mathbf{U}_r} \phi_P \mathbf{U}_r^+ \\ \widehat{\nabla} g_{P,i}(\mathbf{u}_k) &= \frac{\partial g_i}{\partial \mathbf{u}}(\mathbf{u}_k)(\mathbf{I}_{n_u} - \mathbf{U}_r \mathbf{U}_r^+) + \nabla_{\mathbf{U}_r} g_{P,i} \mathbf{U}_r^+ \end{aligned} \quad (23)$$

$i = 1 \dots n_g$

donde el superíndice $(\cdot)^+$ representa la pseudo- inversa de Moore- Penrose, y \mathbf{I}_{n_u} es la matriz identidad de dimensión n_u . En D-MA, los gradientes de la función de coste y de las restricciones del proceso necesarios para estimar los modificadores en (10) son sustituidos por las estimaciones hechas con (23) para cada \mathbf{u}_k .

Costello describió cómo seleccionar las direcciones privilegiadas para el caso en que exista incertidumbre paramétrica (Costello et al., 2016). Dicha selección se lleva a cabo realizando un análisis de sensibilidad del gradiente de la función Lagrangiana con respecto a los parámetros inciertos del modelo β . La idea principal es que si las variaciones de los parámetros afectan significativamente al gradiente de la función Lagrangiana solo en unas pocas direcciones del espacio de entradas, será suficiente con estimar los gradientes de la planta en estas direcciones.

D-MA es fácil de implementar y tiene importantes propiedades teóricas que hacen que converja rápidamente y que garantiza que se alcance el óptimo de la planta.

El algoritmo D-MA es descrito a continuación. La metodología a seguir es la misma que el algoritmo 2 añadiendo nuevas variables a inicializar y reemplazando el cálculo de gradientes experimentales por el de las derivadas direccionales:

Algoritmo 6. Directional Modifier Adaptation (Costello et al., 2016)

Añadir a paso 1: Elegir una matriz de direcciones “privilegiadas” en el espacio de entradas \mathbf{U}_r , en las cuales estimar los gradientes del proceso. Para saber cómo se obtiene esta matriz consultar Costello et al., 2016. Por lo tanto, para inicializar el algoritmo es necesario recoger la información de n_r puntos estacionarios previos.

Paso 3: Estimar la derivada direccional de la función de coste de la planta $\nabla_{\mathbf{U}_r} \phi_P(\mathbf{u}_k)$ y de las restricciones $\nabla_{\mathbf{U}_r} g_{P,i}(\mathbf{u}_k)$ a partir de (19) y a continuación calcular los gradientes del proceso $\widehat{\nabla} \phi_P(\mathbf{u}_k)$ y $\widehat{\nabla} g_{P,i}(\mathbf{u}_k)$ usando (23). Estas derivadas deben estimarse usando medidas recogidas en al menos n_r puntos de operación sucesivos próximos a \mathbf{u}_k .

La principal ventaja de este método es que si el número de direcciones privilegiadas seleccionadas es mucho menor que el número de variables de decisión, la tarea de estimación de

gradientes se ve simplificada drásticamente, sin embargo, existe el problema de seleccionar dichas direcciones correctamente.

3.2. Nested Modifier Adaptation corrigiendo los gradientes del Lagrangiano

Aunque el método NMA no requiere el cálculo de gradientes experimentales, la dimensión del problema, con respecto al número de entradas y restricciones del proceso, también afecta a la convergencia hacia el óptimo ya que cuanto mayor sea el número de modificadores necesarios, mayor será el número de variables de decisión de la capa superior de optimización lo que puede implicar una lenta convergencia hacia el óptimo. Para resolver este problema en presencia de restricciones operacionales, puede implementarse una técnica de MA alternativa la cual consiste en corregir el gradiente de la función Lagrangiana. Este enfoque fue presentado para el método ISOPE en presencia de restricciones de proceso (Brdys et al., 1986). Una formulación equivalente se aplica en el contexto de MA describiendo el problema de optimización modificado según (24), añadiendo los modificadores que corrigen el gradiente del Lagrangiano en la función de coste.

$$\begin{aligned} \min_{\mathbf{u}} \quad & \phi_M = \phi(\mathbf{u}, \mathbf{y}, \beta) + \lambda_k^{L^T} (\mathbf{u} - \mathbf{u}_{k-1}^*) \\ \text{s.t.} \quad & g_{M,i} = g_i(\mathbf{u}, \mathbf{y}, \beta) + \varepsilon_{i,k} \leq 0 \\ & \mathbf{u}^L \leq \mathbf{u} \leq \mathbf{u}^U \\ & i = 1, \dots, n_g \end{aligned} \quad (24)$$

Este enfoque reduce el número de modificadores necesarios como muestra (25) en comparación con (12).

$$n_{mod} = n_g + n_u \quad (25)$$

En el problema modificado (24) $\varepsilon_{i,k}$ son los modificadores de orden cero en las restricciones, $\lambda_k^{L^T}$ es el modificador del gradiente del Lagrangiano que representa la diferencia entre los gradientes de la función Lagrangiana del proceso L_P y del modelo L como muestra la ecuación (26) donde μ_k es el vector de los valores óptimos de los multiplicadores de Lagrange del problema de optimización modificado (24).

$$\lambda_k^{L^T} = \frac{\partial L_P}{\partial \mathbf{u}}(\mathbf{u}_{k-1}, \mu_{k-1}) - \frac{\partial L}{\partial \mathbf{u}}(\mathbf{u}_{k-1}, \mu_{k-1}) \quad (26)$$

$$L_P = \phi_P + \sum_{i=1}^{n_g} \mu_{k,i} g_{P,i} \quad L = \phi + \sum_{i=1}^{n_g} \mu_{k,i} g_i$$

El subíndice “P” indica que la variable ha sido evaluada a partir de medidas del proceso. La técnica descrita tiene la ventaja de que solo requiere un modificador de primer orden por cada entrada del proceso reduciendo el número de variables de decisión de la capa superior de optimización presente en NMA, acelerando la convergencia hacia un punto que satisface las NCO de la planta real. Su principal desventaja es que el hecho de no añadir correcciones de primer orden en las restricciones del proceso, necesarias para satisfacer la igualdad de las condiciones de optimalidad de primer orden del proceso y del modelo, puede provocar mayores violaciones de las restricciones antes de converger. Sin embargo, este algoritmo convergerá a un punto de operación óptimo en el que se satisfagan las condiciones KKT

$$\left(\frac{\partial L_P}{\partial \mathbf{u}} = \frac{\partial L_M}{\partial \mathbf{u}} = 0 \right) \text{ siendo } L_M \text{ la función Lagrangiana del}$$

problema de optimización modificado (24).

La implementación del método se realiza siguiendo los siguientes pasos que se representan gráficamente en la Figura 4:

Algoritmo 7. NMA corrigiendo el gradiente del Lagrangiano (Rodríguez-Blanco et al., 2017b)

Paso 1: Inicializar $k = 0$, los valores iniciales de los modificadores λ^L_{ϕ} , ϵ_{ϕ} , de la variable de entrada \mathbf{u}^*_0 y del multiplicador de Lagrange μ^*_0 .

Paso 2: Dados \mathbf{u}^*_{k-1} y μ^*_{k-1} se aplica \mathbf{u}^*_{k-1} al proceso y se espera a que alcance el estado estacionario. En este instante medir la función de coste del proceso $\phi_{P,k-1}$ y el valor de las restricciones $g_{P,k-1}$. Calcular el modificador de orden cero $\epsilon_{i,k}$ según la expresión (10) y estimar el Lagrangiano del proceso vía (27).

$$L_{P,k-1}(\mathbf{u}^*_{k-1}, \mu^*_{k-1}) = \phi_P(\mathbf{u}^*_{k-1}) + \sum_{i=1}^{n_g} \mu_{k-1,i} g_{P,i}(\mathbf{u}^*_{k-1}) \quad (27)$$

Paso 3: La capa superior de optimización sin restricciones actualiza el valor de los modificadores de primer orden λ^L_k , con el objetivo de minimizar el Lagrangiano del proceso $L_{P,k-1}(\mathbf{u}^*_{k-1}, \mu^*_{k-1})$ resolviendo (28).

$$\min_{\lambda^L_k} L_{P,k-1} \quad (28)$$

Paso 4 (opcional): Filtrar el valor de los modificadores dados por la capa superior de optimización usando la expresión (13).

Paso 5: Resolver el problema de optimización modificado (11) para obtener la nueva variable de decisión \mathbf{u}^*_k que será aplicada al proceso hasta el siguiente estado estacionario y el valor de μ^*_k .

Paso 6: Comprobar si se cumple el criterio de convergencia, por ejemplo $|\mathbf{u}^*_k - \mathbf{u}^*_{k-1}| \leq \text{tol}$, es decir, que no haya cambios en las variables de decisión por encima de una tolerancia dada, y parar el algoritmo si es así. Si esta condición no se satisface, establecer $k = k+1$ y volver al paso 2 cuando se alcance el próximo estado estacionario del proceso.

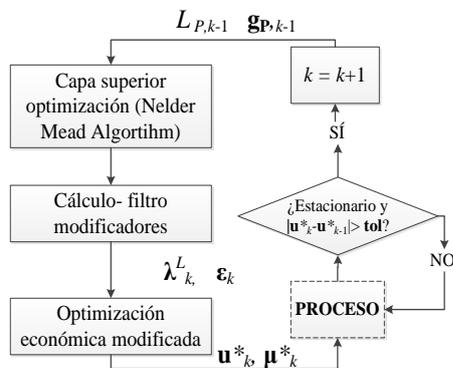


Figura 4: Esquema de implementación NMA corrigiendo el gradiente del Lagrangiano

4. MA basado en medidas del transitorio

Los diferentes métodos presentados anteriormente requieren esperar al estado estacionario del proceso para realizar la actualización de los modificadores. En muchas aplicaciones reales, por ejemplo, la operación de columnas de destilación, dicho estado estacionario puede alcanzarse después de varias horas lo que supone que la convergencia del método sea muy lenta y el punto óptimo de operación se alcance después de varios días de operación. Durante este largo período de tiempo las condiciones de operación o las discrepancias modelo-proceso pueden haber cambiado por lo que el método no convergerá al óptimo real. Éste problema hace que la

implementación de esta metodología en procesos reales no sea eficaz. Por ello, a continuación se presentan algunos enfoques de la metodología MA basadas en medidas transitorias del proceso de modo que la velocidad de convergencia al óptimo se vea acelerada. Al igual que las técnicas de MA tradicionales, esta clase de métodos no garantiza un camino factible durante la convergencia al óptimo, solo cuando se alcanza este punto todas las restricciones del proceso se satisfacen. Sin embargo, al alcanzar el óptimo del proceso más rápido se reduce el tiempo en el que las restricciones operacionales pueden ser violadas.

La idea de usar medidas durante el transitorio para estimar las variables requeridas para la optimización en estado estacionario fue seguida por Zhang y Roberts en 1990 (Zhang and Roberts, 1990), quienes combinaron la técnica ISOPE con una identificación de un modelo dinámico lineal para obtener los gradientes del proceso para la optimización en estado estacionario de procesos no lineales con restricciones y dinámicas lentas.

4.1. MA estimando los gradientes del proceso durante el transitorio basándose en NEC

En el año 2014, François y Bonvin propusieron un enfoque que usa medidas del transitorio para calcular los gradientes de la planta basándose en el método “neighbouring extremal (NEC)” que usa información del transitorio para la optimización en estado estacionario. La principal diferencia es que la actualización de la variable de entrada al proceso no se obtiene calculando una ley de control, sino, resolviendo un problema de optimización modificado.

NEC intenta mantener la optimalidad del proceso en presencia de perturbaciones implementando una ley de control que fuerza la satisfacción de las NCO. El método confía en aproximaciones lineales en torno al óptimo del modelo, principalmente las variaciones de primer orden de las NCO, para estimar la variación de la incertidumbre paramétrica $\delta\beta$ a partir de los cambios en las variables de salida δy y de entrada δu . Por esta razón, una condición que debe satisfacerse es $n_y \geq n_\beta$, es decir, que haya al menos tantas salidas medidas y como parámetros inciertos β .

Se considera el problema de optimización estático presentado en (1) y se tiene en cuenta que el modelo incluye la ecuación para las variables de salida (29):

$$\mathbf{y} = \mathbf{H}(\mathbf{u}, \beta) \quad (29)$$

Siendo las variables \mathbf{u} e \mathbf{y} el conjunto de entradas y salidas en estado estacionario.

A partir del análisis de las variaciones de primer orden de las NCO y considerando las variaciones paramétricas $\delta\beta$ en torno al valor nominal de los parámetros β_{nom} , se puede obtener una aproximación de primer orden del gradiente de la función de coste $\delta\eta$ como se muestra en (30):

$$\delta\eta = \delta \left(\frac{\partial \phi}{\partial \mathbf{u}} \right) = \mathbf{A} \delta \mathbf{u} + \mathbf{B} \delta \beta \quad (30)$$

Obteniendo el valor de \mathbf{A} y \mathbf{B} a partir de (31):

$$\mathbf{A} = \frac{\partial^2 \phi}{\partial \mathbf{u}^2} \quad \mathbf{B} = \frac{\partial^2 \phi}{\partial \mathbf{u} \partial \beta} \quad (31)$$

Las variables de salida pueden linealizarse con respecto a \mathbf{u} y β :

$$\delta \mathbf{y} = \mathbf{Q} \delta \mathbf{u} + \mathbf{P} \delta \beta \quad (32)$$

Calculando \mathbf{Q} y \mathbf{P} a partir de (33):

$$\mathbf{Q} = \frac{\partial \mathbf{y}}{\partial \mathbf{u}} \quad \mathbf{P} = \frac{\partial \mathbf{y}}{\partial \beta} \quad (33)$$

Asumiendo que $n_y \geq n_\beta$ y usando (32) puede obtenerse $\delta\mathbf{y}$ y $\delta\mathbf{u}$ como sigue:

$$\delta\mathbf{p} = \mathbf{D}(\delta\mathbf{y} - \mathbf{Q}\delta\mathbf{u}) \quad (34)$$

donde \mathbf{D} es la pseudoinversa de \mathbf{P} .

La ecuación (30) provee una aproximación de primer orden de la variación del gradiente de la función de coste que puede ser estimada a partir de $\delta\mathbf{y}$ y $\delta\mathbf{u}$ usando (34) para eliminar $\delta\mathbf{p}$:

$$\delta\boldsymbol{\eta} = \mathbf{G}_y \delta\mathbf{y} + \mathbf{G}_u \delta\mathbf{u} \quad (35)$$

$$\mathbf{G}_y = \mathbf{B}\mathbf{D} = \frac{\partial^2 \phi}{\partial \mathbf{u} \partial \mathbf{p}} \left(\frac{\partial \mathbf{y}}{\partial \mathbf{p}} \right)^+ \quad (36)$$

$$\mathbf{G}_u = \mathbf{A} - \mathbf{B}\mathbf{D}\mathbf{Q} = \frac{\partial^2 \phi}{\partial \mathbf{u}^2} - \frac{\partial^2 \phi}{\partial \mathbf{u} \partial \mathbf{p}} \left(\frac{\partial \mathbf{y}}{\partial \mathbf{p}} \right)^+ \frac{\partial \mathbf{y}}{\partial \mathbf{u}} \quad (37)$$

Teniendo en cuenta que, en el óptimo, el gradiente de la función de coste y su variación serán 0 ($\boldsymbol{\eta} = \delta\boldsymbol{\eta} = 0$), ya que se ha considerado una aproximación de primer orden (30), los modificadores $\boldsymbol{\lambda}$ del problema RTO vienen dados por (38), donde $\boldsymbol{\eta}_p$ y $\boldsymbol{\eta}$ son gradientes de la función de coste del proceso y del modelo respectivamente, \mathbf{y}_p e \mathbf{y} son el vector de medidas de las variables de salida del proceso y del modelo, \mathbf{u}_p y \mathbf{u} el vector de entradas aplicadas al proceso y al modelo, y finalmente \mathbf{y}_{nom} el vector de salidas del modelo aplicando la solución nominal \mathbf{u}_{nom} (la solución del RTO sin modificadores).

$$\boldsymbol{\lambda} = \boldsymbol{\eta}_p - \boldsymbol{\eta} = \mathbf{G}_y \delta\mathbf{y}_p + \mathbf{G}_u \delta\mathbf{u}_p - (\mathbf{G}_y \delta\mathbf{y} + \mathbf{G}_u \delta\mathbf{u}) = \mathbf{G}_y \cdot (\delta\mathbf{y}_p - \delta\mathbf{y}) + \mathbf{G}_u \cdot (\delta\mathbf{u}_p - \delta\mathbf{u}) \quad (38)$$

donde:

$$\begin{aligned} \delta\mathbf{y}_p &= \mathbf{y}_p - \mathbf{y}_{nom} \\ \delta\mathbf{y} &= \mathbf{y} - \mathbf{y}_{nom} \\ \delta\mathbf{u}_p &= \mathbf{u}_p - \mathbf{u}_{nom} \\ \delta\mathbf{u} &= \mathbf{u} - \mathbf{u}_{nom} \end{aligned} \quad (39)$$

Se sigue el mismo procedimiento de estimación para calcular los modificadores de restricciones $\boldsymbol{\gamma}$, formulando (35) como la estimación de los gradientes del proceso y calculando \mathbf{G}_y y \mathbf{G}_u en función del valor de la restricción en vez de la función de coste.

La principal ventaja de este método es que el gradiente del proceso se obtiene usando únicamente el modelo fuera de línea a partir de las expresiones para el cálculo de \mathbf{G}_y y \mathbf{G}_u , y tomando medidas del proceso para calcular $\delta\mathbf{y}_p$. Sin embargo, este método presenta ciertas limitaciones:

- El método funciona correctamente solo si existe incertidumbre paramétrica o poca incertidumbre estructural.
- El usuario debe conocer cuáles son los parámetros inciertos, aunque su valor exacto sea desconocido.
- El número de salidas medidas debe ser igual o mayor al número de parámetros inciertos ($n_y \geq n_\beta$).

Algoritmo 8. MA estimando gradientes a partir de NEC (François y Bonvin, 2014)

Paso 1: Identificar los parámetros inciertos del modelo $\boldsymbol{\beta}$ y escoger tantas variables de salida \mathbf{y}_p medidas como parámetros, teniendo en cuenta que éstas deben verse afectadas ante cambios en el valor de $\boldsymbol{\beta}$.

Paso 2: Inicializar $k = 0$, los valores iniciales de los modificadores $\boldsymbol{\lambda}_0$, $\boldsymbol{\varepsilon}_0$, y de la variable de entrada \mathbf{u}^*_0 .

Paso 3: Medir la variación de las variables de salida del proceso $\delta\mathbf{y}_p$ y de las entradas $\delta\mathbf{u}_p$ entre dos RTOs ejecutadas durante el transitorio y calcular \mathbf{G}_y y \mathbf{G}_u a partir del modelo estacionario del RTO según (36) y (37).

Paso 4: Calcular los modificadores según la ecuación (38).

Paso 5: Resolver el problema de optimización modificado (11) para obtener la nueva variable de decisión \mathbf{u}^*_k que será aplicada al proceso hasta el siguiente instante de muestreo, que tendrá lugar durante el transitorio, sin necesidad de esperar al estado estacionario.

Paso 6: Comprobar si se cumple el criterio de convergencia, $|\mathbf{u}^*_k - \mathbf{u}^*_{k-1}| \leq \mathbf{tol}$ y parar el algoritmo. Si esta condición no se satisface, fijar $k = k+1$ y volver al paso 3.

Es importante resaltar que ahora cada ejecución k del RTO no se realiza al alcanzar el estado estacionario del proceso sino durante un transitorio, por tanto cada ejecución k se realizará fijando un periodo fijo de muestreo de la capa RTO.

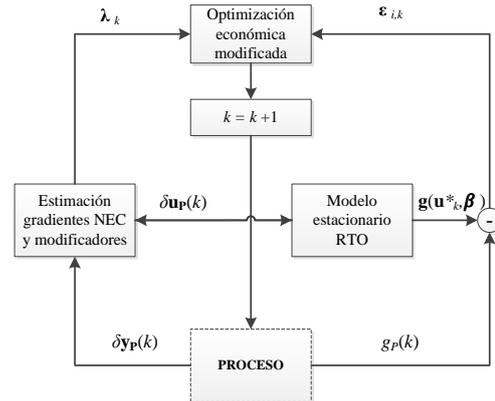


Figura 5: Esquema de implementación MA estimando gradientes del proceso vía NEC.

4.2. MA basado en la estimación directa de los gradientes del proceso usando RELS

Un método alternativo basado también en medidas durante el transitorio consiste en que los gradientes del proceso pueden ser estimados a partir de datos de entrada-salida durante el transitorio usando técnicas de estimación adaptativas. Una vez estimados, se calcula el valor de los correspondientes modificadores. De esta manera se consigue mayor rapidez de convergencia al óptimo estacionario de la planta que en los métodos MA tradicionales.

El primer paso es suponer una aproximación para la variación de la función de coste del proceso, en este caso, se ha elegido un polinomio de Taylor cuadrático que depende de la entrada aplicada en el instante de tiempo actual y las aplicadas en los dos instantes previos. Se pueden usar aproximaciones más sencillas, por ejemplo, de primer orden. Sin embargo esta aproximación sería realista solo para sistemas lineales que no son muy comunes en la práctica.

$$\Delta\hat{\phi}_k = \boldsymbol{\varphi}_k^T \hat{\boldsymbol{\theta}}_k = \frac{\partial \phi}{\partial \mathbf{u}_k} \Delta \mathbf{u}_k^T + \frac{\partial \phi}{\partial \mathbf{u}_{k-1}} \Delta \mathbf{u}_{k-1}^T + \frac{\partial^2 \phi}{\partial \mathbf{u}_k^2} 1/2 (\Delta \mathbf{u}_k \Delta \mathbf{u}_k^T)^T \quad (40)$$

$$+ \frac{\partial^2 \phi}{\partial \mathbf{u}_k \partial \mathbf{u}_{k-1}} (\Delta \mathbf{u}_k \Delta \mathbf{u}_{k-1})^T + \frac{\partial^2 \phi}{\partial \mathbf{u}_{k-1}^2} 1/2 (\Delta \mathbf{u}_{k-1} \Delta \mathbf{u}_{k-1}^T)^T \quad (41)$$

$$\boldsymbol{\varphi}_k^T = \left[\Delta \mathbf{u}_k^T \quad \Delta \mathbf{u}_{k-1}^T \quad 1/2 (\Delta \mathbf{u}_k \Delta \mathbf{u}_k^T)^T \quad (\Delta \mathbf{u}_k \Delta \mathbf{u}_{k-1}^T)^T \quad 1/2 (\Delta \mathbf{u}_{k-1} \Delta \mathbf{u}_{k-1}^T)^T \right]$$

$$\hat{\boldsymbol{\theta}}_k^T = \left[\frac{\partial \phi}{\partial \mathbf{u}_k} \quad \frac{\partial \phi}{\partial \mathbf{u}_{k-1}} \quad \frac{\partial^2 \phi}{\partial \mathbf{u}_k^2} \quad \frac{\partial^2 \phi}{\partial \mathbf{u}_k \partial \mathbf{u}_{k-1}} \quad \frac{\partial^2 \phi}{\partial \mathbf{u}_{k-1}^2} \right] \quad (42)$$

donde $\hat{\boldsymbol{\theta}}_k$ es el vector de parámetros estimados que contiene los gradientes de proceso requeridos para calcular los modificadores, es decir, los gradientes del proceso con respecto a las variables de decisión \mathbf{u}_k .

Los gradientes contenidos en θ_k son estimados empleando el algoritmo de mínimos cuadrados recursivo extendido (RELS) con factor de olvido α . El concepto de olvido significa que los datos más antiguos son descartados gradualmente en favor de la información más reciente (Vahidi et al., 2005).

Se basa en la diferencia entre la entrada actual u_k y la aplicada en las iteraciones previas y en la diferencia entre el cambio en la función de coste medido $\Delta\phi_k$ y el predicho $\Delta\hat{\phi}_k$ (Guay, 2014). El método de estimación de parámetros procede como sigue:

$$\Sigma_0 = \frac{1}{\alpha} \mathbf{I} \tag{43}$$

$$\Delta u_k = u_k - u_{k-1} \tag{44}$$

$$\Delta u_{k-1} = u_{k-1} - u_{k-2} \tag{45}$$

$$\Delta\hat{\phi}_k = \varphi_k^T \hat{\theta}_{k-1} \tag{46}$$

$$e_k = \Delta\phi_k - \Delta\hat{\phi}_k \tag{47}$$

$$(\Sigma_k)^{-1} = \frac{1}{\alpha} (\Sigma_{k-1})^{-1} - \tag{48}$$

$$\frac{1}{\alpha^2} (\Sigma_{k-1})^{-1} \varphi_k \left(1 + \frac{1}{\alpha} \varphi_k^T (\Sigma_{k-1})^{-1} \varphi_k \right)^{-1} \varphi_k^T (\Sigma_{k-1})^{-1}$$

$$\hat{\theta}_k = \hat{\theta}_{k-1} + \frac{1}{\alpha} (\Sigma_{k-1})^{-1} \varphi_k \left(1 + \frac{1}{\alpha} \varphi_k^T (\Sigma_{k-1})^{-1} \varphi_k \right)^{-1} (e_k) \tag{49}$$

donde Σ es la matriz de covarianza del error estimado cuyo valor inicial es Σ_0 , e_k es el error en la predicción de la variación de la función de coste y u_k es la solución del problema RTO modificado para cada período de muestreo durante el transitorio k . Por otro lado, $\Delta\phi_k = \phi_k - \phi_{k-1}$ es la diferencia entre la función de coste del proceso actual y la función de coste medida en el período de muestreo anterior.

Una ventaja de esta técnica comparada con la descrita en la sección 4.1 es que no es necesario calcular segundas derivadas para estimar los gradientes de la planta. Además, esta formulación no requiere ninguna suposición sobre el tipo de incertidumbre del modelo, si es paramétrica o estructural, ni tampoco saber cuál es el parámetro responsable de dicha incertidumbre. Por ello, este método puede aplicarse en presencia de los dos tipos de incertidumbre sin necesidad de identificar cuáles son los parámetros inciertos.

La implementación de este método se lleva a cabo siguiendo las etapas que se describen a continuación. Una representación gráfica de la técnica descrita se muestra en la Figura 6.

Algoritmo 9. MA estimando gradientes a partir de RELS (Rodríguez-Blanco et al., 2017a)

Paso 1: Establecer $k = 0$ e inicializar el valor del vector de parámetros inciertos $\hat{\theta}_0$, el valor del error de estimación e_0 y la matriz de covarianza Σ_0 cuyo valor inicial suele ser tomado como (44). Para inicializar el algoritmo se deben aplicar anteriormente dos valores de las variables de decisión, u_{k-1} y u_{k-2} , para realizar la primera estimación del gradiente del proceso. Independientemente del número de variables de decisión del problema siempre se necesitarán tan solo dos puntos de operación previos para poder inicializar la estimación del gradiente.

Paso 2: En el instante de muestreo k , durante el transitorio, se mide la función de coste del proceso $\phi_{p,k-1}$ y las restricciones $g_{p,k-1}$. A partir de estas medidas y las tomadas en el instante de muestreo anterior $k-1$, se calculan los términos de variación $\Delta\phi_{p,k}$ y $\Delta g_{p,k}$ y el vector de datos es formado por (50):

$$\begin{aligned} \Delta\phi_{p,k} &= \phi_{p,k-1} - \phi_{p,k-2} \\ \Delta g_{p,k} &= g_{p,k-1} - g_{p,k-2} \end{aligned} \tag{50}$$

$$\Delta u_k = u_k - u_{k-1}$$

$$\Delta u_{k-1} = u_{k-1} - u_{k-2}$$

Paso 3: El error de estimación de la variación de la función de coste del proceso $e_{\phi,k}$ y de las restricciones $e_{g,k}$ se calcula según (51):

$$e_{\phi,k} = \Delta\phi_{p,k} - \varphi_{k-1}^T \hat{\theta}_{\phi,k-1} \tag{51}$$

$$e_{g,k} = \Delta g_{p,k} - \varphi_{k-1}^T \hat{\theta}_{g,k-1}$$

Paso 4: Estimar utilizando el algoritmo RELS descrito por (44)-(50), el gradiente de la función de coste y de las restricciones del proceso, $\left. \frac{\partial\phi_p}{\partial u} \right|_{u_{p,k-1}}$ y $\left. \frac{\partial g_p}{\partial u} \right|_{u_{p,k-1}}$ contenidos en el vector

$\hat{\theta}_{\phi,k}$ y $\hat{\theta}_{g,k}$ respectivamente.

Paso 5: Con los gradientes obtenidos en el paso anterior estimar los modificadores λ_k , γ_k y ϵ_k aplicando las expresiones (10).

Paso 6: Resolver el problema de optimización modificado (11) para obtener la nueva variable de decisión u^*_k que será aplicada al proceso hasta el siguiente instante de muestreo.

Paso 7: Comprobar si se cumple el criterio de convergencia, por ejemplo $|u^*_k - u^*_{k-1}| \leq \text{tol}$, es decir, que no haya cambios en las variables de decisión por encima de una tolerancia dada, y parar el algoritmo si es así. Si esta condición no se satisface, fijar $k = k+1$ y volver al paso 2.

Igual que en algoritmo 8 cada ejecución k del RTO no requiere que el proceso haya alcanzado el estado estacionario sino que se ejecutará cada período de muestreo, fijado de antemano, durante el transitorio.

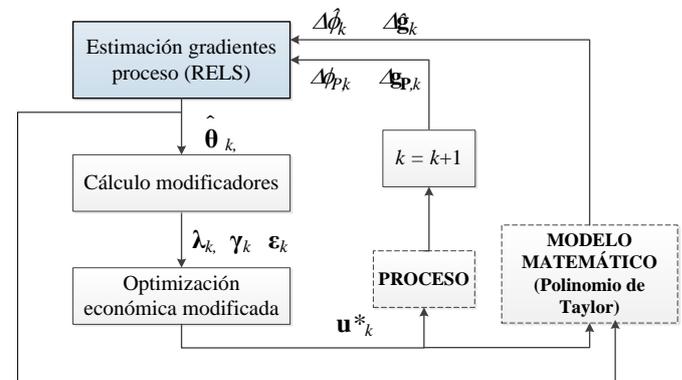
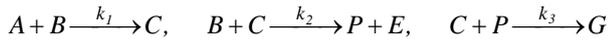


Figura 6: Esquema de implementación MA basado en la estimación de los gradientes del proceso durante el transitorio

5. Aplicaciones

Las diferentes técnicas descritas anteriormente dentro de la metodología MA han sido implementadas de forma exitosa en numerosas aplicaciones. Uno de los problemas más tratados en este campo ha sido la operación óptima del reactor Otto Williams, en presencia de incertidumbre paramétrica y estructural entre el modelo RTO y el proceso. Ésta incertidumbre surge de una simplificación en el modelo dinámico debida a que la presencia de uno de los componentes es muy baja por lo que se decide no tenerlo en cuenta en el modelo estacionario, lo que conlleva un importante cambio en la cinética de las reacciones que ocurren en el proceso. En la planta real ocurren las tres reacciones irreversibles que se muestran:



donde k_1, k_2 y k_3 son las constantes cinéticas de la reacción. En cambio, el modelo RTO sólo considera dos reacciones:



El objetivo de la optimización es maximizar el beneficio en el estacionario manipulando el flujo de reactivo F_B y la temperatura del reactor T_R :

$$\phi_P = F_R (X_P P_P + X_E P_E) - F_A X_A C_A - F_B X_B C_B$$

Sobre este ejemplo se han aplicado los métodos basados en información del estacionario ISOPE (Roberts, 1979), DMA (Marchetti et al., 2010), NMA (Navia et al., 2013) e IGMO (Gao et al., 2015a) logrando siempre alcanzar el punto óptimo de operación. La Tabla 1 muestra el número de estados estacionarios requeridos por cada método fijando un mismo criterio de parada $\|\mathbf{u}_k^* - \mathbf{u}_{k-1}^*\| \leq 0.01$. Cada estacionario corresponde aproximadamente a una hora de operación. Estos resultados y una explicación detallada del rendimiento de cada técnica se presentan en (Gao et al., 2015b).

Tabla 1: Número de estados estacionarios necesarios.

	ISOPE	DMA	NMA	IGMO
Número de estacionarios	10	14	25	8

La metodología basada en la estimación de gradientes durante el transitorio usando RELS (Rodríguez-Blanco et al., 2017a) también ha sido implementada reduciendo 4 veces el tiempo requerido para alcanzar el óptimo en comparación con el uso de DMA ($\delta^L = 0.01$) (Rodríguez-Blanco et al., 2016), aunque el número de soluciones RTO obtenidas es mayor en el caso del método basado en el transitorio, ya que se ejecuta con una frecuencia 6 veces mayor que en el caso estacionario.

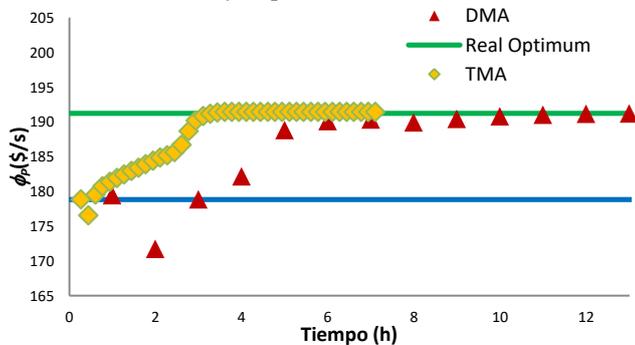


Figura 7: Evolución del beneficio obtenido usando DMA y el método transitorio estimando gradientes a partir de RELS (TMA).

Otra técnica que supuso un ahorro de tiempo importante con respecto a la implementación de los métodos estacionarios tradicionales fue NMA corrigiendo los gradientes del Lagrangiano (Rodríguez-Blanco et al., 2017b) reduciendo en 3 veces tanto el número de iteraciones requeridas como el tiempo necesario para converger al óptimo en comparación con el uso del método NMA convencional. Para realizar este ejemplo se añadieron dos restricciones operacionales en la composición del reactivo A y el subproducto G en el proceso para poder observar las ventajas que presenta la nueva formulación NMA.

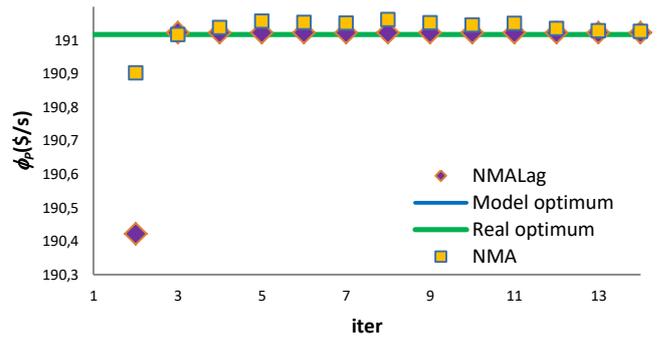


Figura 8: Evolución del beneficio obtenido usando NMA y NMA con el modificador del Lagrangiano en la función de coste (NMA Lag).

Fijando una tolerancia del 0.5% con respecto al valor óptimo de la función de coste se obtienen los resultados mostrados en la Tabla 2. NMA requiere 46 estados estacionarios, los 3 primeros para estimar los gradientes para el cálculo de los modificadores iniciales, 7 para construir el primer simplex que corresponde al número de modificadores de primer orden más 1 y por último 36 soluciones RTO. La formulación alternativa de NMA solo necesita 11 estados estacionarios, 3 para la estimación inicial de los modificadores, 3 para construir el simplex y finalmente 5 soluciones RTO para converger.

Tabla 2: Comparación NMA y NMA Lag.

	NMA	NMA Lag
Número de modificadores	8	3
Modificadores de primer orden	6	2
Iteraciones	14	4
Puntos estacionarios	46	11

Una aplicación más realista ha sido utilizada por (Rodríguez-Blanco et al., 2015), considerando la operación de una columna de destilación despropanizadora que cuenta con una capa de control predictivo. En este caso se presentan dos tipos de incertidumbre, la debida al uso de un modelo erróneo o inexacto en la capa RTO y la originada por la presencia de distintos tipos de modelos en la estructura de control (RTO usa un modelo estacionario no lineal, MPC basado en modelos dinámicos lineales). En este caso, el modelo dinámico utilizado para representar el proceso de forma muy detallada cuenta con 2152 variables (1976 explícitas, es decir, que se despejan explícitamente de una ecuación, 129 en derivadas, 40 algebraicas o implícitas y 7 condiciones de contorno, es decir, número de grados de libertad del modelo) y 2145 ecuaciones (129 ecuaciones diferenciales y 2023 algebraicas), sin embargo, la capa RTO utiliza un modelo estacionario simplificado con tan solo 32 ecuaciones algebraicas y 39 variables (29 explícitas, 3 algebraicas y 7 condiciones de contorno) lo que supone una gran incertidumbre estructural modelo-proceso. Sobre esta aplicación se han implementado las técnicas DMA y NMA (Rodríguez-Blanco et al., 2015) observando que las técnicas estacionarias consiguen alcanzar el óptimo pero después de varios días de operación, concretamente en 60 horas, lo que implica 10 estados estacionarios. Para acelerar la convergencia al punto óptimo de operación se decidió aplicar los métodos basados en el transitorio, como el basado en NEC (Rodríguez-Blanco et al., 2016) sólo aplicable cuando existe incertidumbre paramétrica, y el basado en RELS (Rodríguez-Blanco et al., 2017c) válido en presencia de incertidumbre estructural y paramétrica, alcanzando el óptimo del proceso en un solo estado estacionario en ambos casos, aproximadamente en tan solo 8 horas de operación.

El método D-MA fue implementado sobre un ejemplo experimental que consistía en hallar la trayectoria óptima de una cometa (Costello *et al.*, 2016), este problema contaba con 40 variables de decisión lo que hace inviable la implementación de los métodos MA tradicionales. En presencia de incertidumbre estructural y paramétrica, D-MA es capaz de encontrar la trayectoria óptima estimando los gradientes en dos únicas direcciones ($n_r = 2$) en tan solo 10 iteraciones, la comparación del rendimiento del algoritmo se ha llevado a cabo considerando un mayor número de direcciones ($n_r = 4$) observándose que la convergencia es mucho más lenta en este caso, ya que se estiman gradientes en más direcciones de las necesarias.

Entre las aplicaciones experimentales destacan la implementación en una celda de combustible (Bunin *et al.*, 2012) y en una columna de flotación utilizada para la recuperación de cobre (Navia *et al.*, 2016) donde los autores demuestran que el método de dos etapas no es suficiente para alcanzar el óptimo real y que DMA y NMA sí que son capaces aunque alguna iteración intermedia no sea factible. En este último ejemplo, el método MA estimando los gradientes experimentales a partir de RELS, también ha sido implementando reduciendo aproximadamente el número de estacionarios requeridos en 10, en comparación con usar DMA.

6. Conclusiones

En este artículo se ha presentado una revisión de las diferentes técnicas desarrolladas dentro de la metodología MA, las cuales suponen una gran herramienta para tratar con las discrepancias modelo-proceso en RTO y conducir los procesos hasta un punto de operación que satisfice las NCO del proceso.

En el artículo también se describen dos técnicas desarrolladas recientemente por los autores que implican importantes ventajas en la implementación de MA. Una de ellas, tratando la problemática de la presencia de numerosas restricciones operacionales, la nueva formulación de NMA corrigiendo el gradiente del Lagrangiano, y otra acelerando la convergencia de MA ejecutando RTOs durante el transitorio estimando los modificadores a partir de información dinámica utilizando técnicas de estimación adaptativas como RELS.

Sin embargo aún existen problemas abiertos como, garantizar la factibilidad de la trayectoria hacia el óptimo, sin que esto afecte a la velocidad de convergencia del método o usar MA en presencia de variables enteras donde el cálculo de gradientes con respecto a variables de decisión enteras no puede ser aplicado tal como se ha descrito en este artículo.

Agradecimientos

Este trabajo ha sido realizado gracias al proyecto DPI2015-70975-P del MINECO del Gobierno de España bajo la beca FPI BES-2013-062737.

Referencias

Brdys, M., Chen, S., Roberts, P.D. 1986. An extension to the modified two-step algorithm for steady-state system optimization and parameter estimation. *International Journal of Systems Science*, 17:8, 1229 – 1243.

Brdys, M., Roberts, P.D. 1987. Convergence and optimality of modified two-step algorithm for integrated system optimisation and parameter estimation. *Int. Journal of Systems Science*, 18(7), 1305-1322.

Brdys, M., Tatjewski, P. 1994. An algorithm for steady-state optimizing dual control of uncertain plants. 1st IFAC Workshop on new trends in design of control systems, 249-254. Smolenice, Slovakia.

Brdys, M., Tatjewski, P. 2005. Iterative algorithms for multilayer optimizing control. Imperial College Press, London UK.

Bunin, G. A., Wuillemin, Z., François, G., Nakajo, A., Tsikonis, L., & Bonvin, D. 2012. Experimental real-time optimization of a solid oxide fuel cell stack via constraint adaptation. *Energy*, 39, 54-62.

Chen, C.Y., Joseph, B., 1987. On-line optimization using a two-phase approach: An application study. *Industrial & Engineering Chemistry Research*, 26, 1924-1930.

Chachuat, B., Srinivasan, B., & Bonvin, D. 2009. Adaptation strategies for real-time optimization. *Computers & Chemical Engineering*, 33, 1557-1567.

Costello, S., François, G., Bonvin, D., 2016. A directional modifier adaptation algorithm for real-time optimization. *J. Process Control*, 39, 64-76.

Engell, S., 2007. Feedback control for optimal process operation, *J. Process Control*, 17, 203-219.

François, G., Srinivasan, B., Bonvin, D. 2005. Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *Journal of Process Control*, 15(6),701-712.

François, G., Bonvin, D., 2014. Use of transient measurements for the Optimization of Steady-State Performance via Modifier Adaptation. *Industrial & Engineering Chemistry Research*, 53 (13), 5148-5159.

Forbes, J.F., Marlin, T.E. 1994. Model accuracy for economic optimizing controllers: the bias update case. *Industrial & Engineering Chemistry Research*, 33, 1919-1929.

Gao, W., Engell, S., 2005. Iterative set-point optimization of batch chromatography. *Computers & Chemical Engineering*, 29, 1401-1409.

Gao, W., Wenzel, S., Engell, S., 2015a. A reliable modifier-adaptation strategy for real-time optimization. *Computers & Chemical Engineering*, 91, 318-328.

Gao, W., Wenzel, S., & Engell, S., 2015b. Comparison of Modifier Adaptation Schemes in Real-Time Optimization. In *ADCHEM 2015* (Vol. 48, 182-187). Whistler, Canada.: IFAC.

Guay, M., 2014. A time-varying extremum-seeking control approach for discrete-time systems. *Journal of Process Control* 24, 98-112.

Krstic, M., Wang, H. 2000. Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36, 595-601.

Marchetti, A., Chachuat, B., & Bonvin, D., 2009. Modifier-Adaptation Methodology for Real-Time Optimization. *Industrial & Engineering Chemistry Research*, 48, 6022-6033.

Marchetti, A., Chachuat, B., Bonvin, D., 2010. A dual modifier-adaptation approach for real-time optimization. *Journal of Process Control*, 20, 1027-1037.

Marlin, T., Hrymak, E. A. N., 1997. Real-time operations optimization of continuous processes. *AIChE Symposium Series*, 93, 156-164.

Navia, D., Gutiérrez, G., de Prada, C. 2013. Nested Modifier-Adaptation Methodology for RTO in the Otto Williams Reactor. In 10th International Symposium on Dynamics and Control Process Systems (DYCOPS 2013); IFAC: Mumbai, India.

Navia, D., Briceño, L., Gutiérrez, G., de Prada, C., 2015. Modifier-adaptation methodology for real-time optimization reformulated as a nested optimization problem. *Ind. Eng. Chem. Res* 2015; 54,12054-71.

Navia, D., Villegas, D., Cornejo, I., & de Prada, C., 2016. Real-time optimization for a laboratory-scale flotation column. *Computers & Chemical Engineering*, 86, 62-74.

Roberts, P.D., 1979. An algorithm for steady-state system optimization and Parameter-Estimation. *International Journal of Systems Science*, 10, 719-734.

Rodríguez-Blanco, T., Sarabia, D., Navia, D., de Prada, C., 2015. Modifier-adaptation methodology for RTO applied to distillation columns. *ADCHEM 2015*, Whistler, British Columbia, Canada.

Rodríguez-Blanco, T., Sarabia, D., de Prada, C., 2016. Modifier- adaptation methodology for RTO applied to distillation columns using a simplified steady-state model. *MSC 2016*, Buenos Aires, Argentina.

Rodríguez-Blanco, T., Sarabia, D., de Prada, C., 2017a. Modifier- adaptation approach using RELS to compute process gradients. *FOCAPO-CPC 2017*. Tucson, Arizona.

Rodríguez-Blanco, T., Sarabia, D., de Prada, C., 2017b. Nested Modifier Adaptation for RTO correcting the Lagrangian gradients applied to the Otto Williams reactor. *ESCAPE 27*. Barcelona, España.

Rodríguez-Blanco, T., Sarabia, D., de Prada, C., 2017c. Modifier Adaptation methodology based on transient and static measurements for RTO to cope with structural uncertainty. *Computers & Chemical Engineering*, 106, 480-500.

Skogestad, S., 2000. Self- optimizing control: The missing link between steady-state optimization and control. *Computers and Chemical Engineering*, 24, 569-575.

Tatjewski, P., 2008. Advanced control and on-line process optimization in multilayer structures. *Annual Reviews in Control*, 32, 71-85.

Vahidi, A., Stefanopoulou, A., Peng, H., 2005. Recursive least squares with forgetting for online estimation of vehicle mass and road grade: Theory and experiments. *Vehicle System Dynamics*, 2005: 43, 31-55.