XLIV Jornadas de Automática 2023

https://jautomatica.es

Optimización experimental con presupuesto finito combinando heurísticas Bayesianas en un POMDP

Pitarch, J.L.[a,*], Armesto, L.[b], Sala, A.[a], Montes, D.[c]

[a] *Instituto de Automática e Informática Industrial (ai2), Universitat Politècnica de València, Camino de Vera, s/n, 46022, Valencia, España.*
[b] *Instituto de Diseño y Fabricación (IDF), Universitat Politècnica de València, Camino de Vera, s/n, 46022, Valencia, España.*
[c] *Dpto. Ingeniería de Sistemas y Automática, Universidad de Valladolid, C/ Dr. Mergelina s/n, 47011, Valladolid, España.*

To cite this article: Pitarch, J.L., Armesto, L., Sala, A., Montes, D., 2023. Experimental optimisation with finite budget combining Bayesian heuristics in a POMDP. XLIV Jornadas de Automática, 447-452. https://doi.org/10.17979/spudc.9788497498609.447

Resumen

Mejorar la toma de decisiones a partir de los resultados observados tras la experimentación es una tarea habitual en muchas aplicaciones, tanto a nivel de investigación en laboratorio como a escala industrial. Sin embargo, realizar experimentos suele acarrear un coste no despreciable, por lo que una excesiva exploración es perjudicial. La optimización Bayesiana es una técnica muy utilizada en este contexto, debido a su bajo coste computacional y a que proporciona un buen balance entre explotación y exploración. No obstante, está técnica no tiene en cuenta explícitamente el coste real de realizar un experimento, ni si existe un presupuesto (o número de experimentos, tiempo, etc.) máximo. El problema de toma de decisiones bajo incertidumbre y presupuesto finito es un proceso de decisión de Markov parcialmente observable (POMDP, por sus siglas en inglés). Este trabajo aborda el problema de optimización experimental combinando reconocidas heurísticas Bayesianas en un enfoque POMDP resuelto mediante programación dinámica, donde un árbol de escenarios se construye partir del conocimiento del proceso/sistema disponible (con incertidumbre) en cada etapa. Dicho conocimiento se modela mediante un proceso Gaussiano que se actualiza con cada nueva observación. El algoritmo desarrollado ha sido testeado con éxito para optimizar las consignas de un reactor de tanque agitado que debe producir una cierta cantidad de lotes.

Palabras clave: Programación dinámica, Optimización de procesos, Procesos Gaussianos, Optimización bajo incertidumbre.

Experimental optimisation with finite budget combining Bayesian heuristics in a POMDP

Abstract

Improving decision making from the observed results after experimentation is a usual task in many applications, from the research lab scale to the industrial one. However, conducting experiments often takes a non-negligible cost. Consequently, an excessive exploration is harmful. Bayesian optimisation is a widely-used technique in this context, due to its low computational cost and because it provides good exploration-exploitation trade-offs. However, this technique does not explicitly account for the actual cost of the experiment, nor whether a limited budget (economic, number of experiments, time, etc.) exists. The problem of decision making under uncertainty and finite budget is a Partially-Observable Markov Decision Process (POMDP). This work addresses the experimental optimisation problem by combining well-known Bayesian heuristics in a POMDP framework solvable via dynamic programming, where a scenario tree is built from the available system/process knowledge (with uncertainty) at each stage. Such a knowledge is modelled as a Gaussian process which is updated with each new observation. The developed algorithm has been tested successfully to optimise the setpoints in a continuous stirred tank reactor that must produce a certain number of batches.

Keywords: Dynamic programming, Process optimisation, Gaussian processes, Optimisation under uncertainty.

*Autor para correspondencia: jlpitarch@isa.upv.es

1. Introducción

El aprendizaje a través de la experimentación-observación es una tarea a llevar a cabo en cualquier proceso de toma de decisiones donde no se dispone de un conocimiento perfecto del sistema y/o el entorno, es decir, no se dispone de un modelo de predicción. Existen diversas técnicas aplicables en este contexto: las indirectas, basadas en identificación de modelos y control adaptativo (Yip y Marlin, 2003; Rodríguez-Blanco et al., 2017); las directas, sin uso de modelos, basadas en la búsqueda y aprendizaje de la política de control óptima (Deisenroth et al., 2013).

En problemas donde realizar un excesivo numero de experimentos reales acarrea un coste importante, se utilizan técnicas indirectas que buscan el óptimo sobre un modelo subrogado, previo paso al experimento real. La optimización Bayesiana (OB) es la técnica más popular en este contexto (Calandra et al., 2016; del Rio Chanona et al., 2021), donde el conocimiento (incierto) de la función a optimizar se suele caracterizar de forma probabilística por un proceso Gaussiano (PG). A partir de este modelo probabilístico, la OB se basa en optimizar una cierta *función de adquisición* que no es más que una heurística o compromiso entre exploración y explotación.

Existen varias funciones de adquisición disponibles en la literatura (Frazier, 2018), así como funciones de covarianza semiparamétricas de diferente complejidad para generar el PG a partir de regresión con los datos recogidos (Wu y Movellan, 2012). No obstante, la OB está pensada para resolver un problema de optimización estático y no tiene explícitamente en cuenta el coste acumulado de realizar experimentos y que puede existir un límite a los mismos en su formulación.

Encontrar las decisiones que arrojen el mejor compromiso exploración-explotación bajo incertidumbre, cuando realizar un experimento acarrea un coste, pero a su vez puede mejorar el conocimiento del sistema, convierte el problema de optimizar una función estática en un problema dinámico. Este tipo de problemas se presentan en la literatura como procesos de decisión de Markov parcialmente observables (Spaan, 2012), y la OB aplicada a éstos es sólo una heurística miope de predicción a un paso (Astudillo et al., 2021). Un problema POMDP debe ser resuelto mediante programación dinámica (PD), explorando todas las opciones posibles derivadas del producto cartesiano entre acciones y observaciones. De esta forma puede valorar todas las potenciales consecuencias de tomar decisiones a varios pasos futuros (Busoniu et al., 2017; Armesto y Sala, 2022). Sin embargo, resolver estos problemas de forma óptima suele ser intratable computacionalmente y, en especial, cuando las variables de decisión pueden tomar infinitos valores en un espacio continuo.

Una idea interesante es combinar OB con un enfoque POMDP de forma que se obtengan algoritmos no miopes, i.e. capaces de estimar el coste a varios pasos futuros simulando (y actualizando) el PG con decisiones que vienen dadas por una función de adquisición de OB prefijada (Lam et al., 2016; Astudillo et al., 2021). Sin embargo, suponer una determinada heurística para calcular las futuras decisiones "virtuales" con el modelo PG se antoja claramente subóptimo.

En este trabajo se aborda la optimización experimental pura con un presupuesto finito como un problema OB-POMDP. Para ello, los autores proponemos una estrategia no miope y eficiente que evalúa de forma dinámica (i.e. en cada paso de predicción elige entre) varias funciones de adquisición de OB. Con esta elección inteligente a la hora de evaluar escenarios de futuras acciones, se reduce el conservadurismo de otras propuestas de la literatura sin pagar un inabordable precio en coste computacional (Armesto et al., 2023). Para demostrarlo, aplicamos el algoritmo desarrollado para resolver un problema de optimización experimental en un reactor químico de producción por lotes, donde cada experimento (lote producido) consume recursos y el coste total de producción se obtiene una vez se ha cumplido la producción de todos los lotes.

El resto del artículo se estructura como sigue. La siguiente sección resume los métodos utilizados y formaliza el problema a resolver. La Sección 3 presenta una síntesis del algoritmo OB-POMDP desarrollado. En la Sección 4 se presentan y discuten los resultados obtenidos en el caso de estudio del reactor químico. El artículo concluye con las ideas relevantes.

2. Métodos y problema a resolver

Formalmente, el objetivo de la OB es encontrar $x^* = \arg\min_{x\in\mathbb{X}} f(x)$, donde $\mathbb{X} \subset \mathbb{R}^u$ es un conjunto de decisiones válidas, sin asumir existencia de dinámica en $f: \mathbb{R}^u \to \mathbb{R}$. Para ello se optimiza alguna heurística con un modelo probabilístico de f, generalmente un PG de media μ y covarianza Σ, $f(x) \approx \mathcal{N}(\mu(x), \Sigma(x))$, generado a partir de una función de correlación o "suavidad" $\kappa(x_a, x_b)$ entre dos datos de entradas (x_a, x_b) cualesquiera, y un histórico de datos pasados entrada y salida. Dichos datos se suponen provenientes de medidas afectadas por ruido: $y = f(x) + w,\ w \approx \mathcal{N}(0, \lambda^2)$.

Si la media del PG incluye una parte determinista con un parámetro explícito θ a estimar (i.e., la media constante o *bias* del PG), entonces estamos hablando del método de regresión conocido como *ordinary krigging* en la literatura, muy utilizado en geología matemática y ciencias de sistemas de la Tierra (Cressie, 1990). Conceptualmente, todo se reduce a hacer regresión del PG con una función de correlación

$$\tilde{\kappa}(x_a, x_b) \coloneqq \kappa(x_a, x_b) + \Sigma_\theta + \lambda I; \tag{1}$$

donde $\kappa(x_1, x_2)$ es cualquier función de correlación Gaussiana típica y Σ_θ es la covarianza a priori del parámetro a estimar (Wu y Movellan, 2012). Entonces, dado un histórico de n datos de entrada $X \coloneqq [x_1, x_2, x_3, \dots, x_n]$ y salida $Y \coloneqq [y(x_1), y(x_2), \dots, y(x_n)]$, la mejor predicción mínimo-cuadrática de $y(x)$ es:

$$\hat{y}(x) = \tilde{\kappa}(x, X)\tilde{\kappa}(X, X)^{-1} Y \tag{2}$$

2.1. Enfoque de programación dinámica

Dado un conocimiento o estimación de f, que en nuestro contexto POMDP será el PG construido a partir de un histórico de datos, denotado a partir de ahora por $\beta(Y, X)$, el enfoque de la programación dinámica es encontrar una política de decisión $x = \pi(X, Y)$, $\pi: \mathbb{X} \times \mathbb{R} \to \mathbb{R}^u$, que minimice cierto coste o *función de valor* sobre una secuencia de N experimentos:

$$V_\pi(X_0, Y_0) \coloneqq \mathbb{E}_\beta \left\{ \sum_{k=1}^{N} \gamma^k\, r\left(y\left(\pi(X_k, Y_k)\right)\right) \right\} \tag{3}$$

En (3), X_k e Y_k es histórico de datos hasta el experimento k, $r: \mathbb{R} \to \mathbb{R}$ es el coste inmediato, y $0 < \gamma \leq 1$ es el llamado factor de descuento. La función de valor V^* de una política de decisión óptima π^* debe verificar la ecuación de Bellman:

$$V^*(X,Y) = \min_x \mathbb{E}_{\beta,x}\{r(y) + \gamma\, V^*([y;Y],[x;X])\} \quad (4)$$

De esta manera, la política óptima es aquella que cumple:

$$\pi^*(X,Y) = \arg\min_x \mathbb{E}_{\beta,x}\{r(y) + \gamma\, V^*([y;Y],[x;X])\} \quad (5)$$

Para que resolver (5) sea tratable computacionalmente, en este trabajo se asume que $N \ll \infty$, y el espacio de decisiones $x \in \mathcal{X}$ se restringe a unas pocas opciones dadas por las diferentes heurísticas/funciones de adquisición de la OB.

2.2. Planteamiento del problema

Dado el modelo probabilístico (2) de $f(x)$ y el problema de programación dinámica (5), el objetivo es plantear un POMDP de horizonte finito para encontrar la secuencia de decisiones x_k que optimizan un índice de coste sobre observaciones reales

$$J \coloneqq \gamma^N J_N(Y_N) + \sum_{k=1}^{N-1} \gamma^k\, r(Y_k)\,; \quad (6)$$

existiendo varias opciones tanto para el coste inmediato r como para el terminal J_N (Armesto et al., 2023).

La idea es que, una vez se mida la respuesta y_k del sistema $f(x_k)$ en cada experimento k, el modelo probabilístico $\beta(Y_k, X_k)$ debe actualizarse y se repite el procedimiento hasta que se consuma el presupuesto asignado para experimentación. En la siguiente sección se presenta la solución propuesta.

3. Estrategia OB-POMDP

El objetivo del algoritmo desarrollado es decidir cuál será la siguiente decisión x_k, $k: 1, \dots, N$, a experimentar dada una historia de experimentos anteriores $\{X_{k-1}, Y_{k-1}\}$ y su PG asociado. La combinación de un modelo probabilístico y la programación dinámica requiere considerar un árbol de escenarios que se construye a partir de un conocimiento o modelo inicial $\beta_0 \coloneqq \beta(Y_0, X_0)$. Cada escenario evalúa un conjunto de decisiones $\tilde{x}_l \in \mathcal{X}$ y observaciones "virtuales" $\tilde{y}_l \in \mathcal{Y}$ para $l: 1, \dots, N-k+1$, i.e., las decisiones que quedan por tomar sobre el presupuesto/horizonte total N. De acuerdo a (5), la decisión de primera etapa seleccionada, x_1, será aquella que obtenga el menor coste esperado. Ésta será por tanto la candidata a testar experimentalmente, y su respuesta medida y_1 servirá para actualizar el conocimiento sobre la función real f, i.e, el PG.

3.1 Árbol de escenarios

El árbol anteriormente descrito se construye con dos tipos de nodos/bifurcaciones: nodos asociados a decisiones y nodos para promediar observaciones (asociados a cada decisión $\tilde{x}_j$). Si un nodo, denotado como $\mathcal{N}$, es de decisión, entonces estima la función de valor $V_{\mathcal{N}}$ del escenario óptimo de entre los generados por la rama del árbol que comprende a él y todos sus hijos mediante propagación inversa. Los $\mathcal{N}'$ hijos de un nodo decisión serán nodos observación, generados por ramificación con el conjunto de decisiones candidatas $\tilde{x}_i \in \mathcal{X}$. Dichas decisiones son calculadas mediante optimización

$$\tilde{x}_{\mathcal{N},i} \coloneqq \arg\min_x F_i(\beta_{\mathcal{N}}, x) \quad (7)$$

siendo F_i funciones de adquisición típicas de la OB:

- Valor esperado (*Expected Value*)
- Probabilidad de mejorar (*Probability of Improvement)*
- Mejora esperada (*Expected Improvement)*
- Límite de confianza inferior (*Lowest Confidence Bound)*

Cada nodo de decisión se asocia a un modelo probabilístico de f (i.e. un PG denotado por $\beta_{\mathcal{N}}$) y un conjunto de acciones $(x_{\mathcal{N},i})$, y devuelve la función de valor $V_{\mathcal{N}}$ asociada a la acción $x^*_{\mathcal{N}}$ que minimiza el coste esperado hasta profundidad N.

Un nodo de observación estima igualmente la función de valor $Q_{\mathcal{N}}$ del escenario óptimo de entre los generados por la rama del árbol que comprende a él y todos sus hijos, pero a partir del conocimiento $\beta_{\mathcal{N}}$ de su nodo padre y una decisión $x_{\mathcal{N}}$ candidata dada. Los hijos de un nodo de observación serán nodos de decisión, generados por ramificación de posibles observaciones de f con la incertidumbre existente en $\beta_{\mathcal{N}}$. Para limitar la complejidad del árbol, en lugar de muestrear el PG extensivamente para aproximar la función de valor, utilizamos un conjunto de M valores y_j con sus pesos asociados α_j dados por la cuadratura de Gauss-Hermite (Abramowitz y Stegun, 1972). El valor esperado de $Q_{\mathcal{N}}$ se aproxima entonces con los estimados asociados a cada y_j, análogo a la idea que subyace en el desarrollo del filtro *unscented* de Kalman.

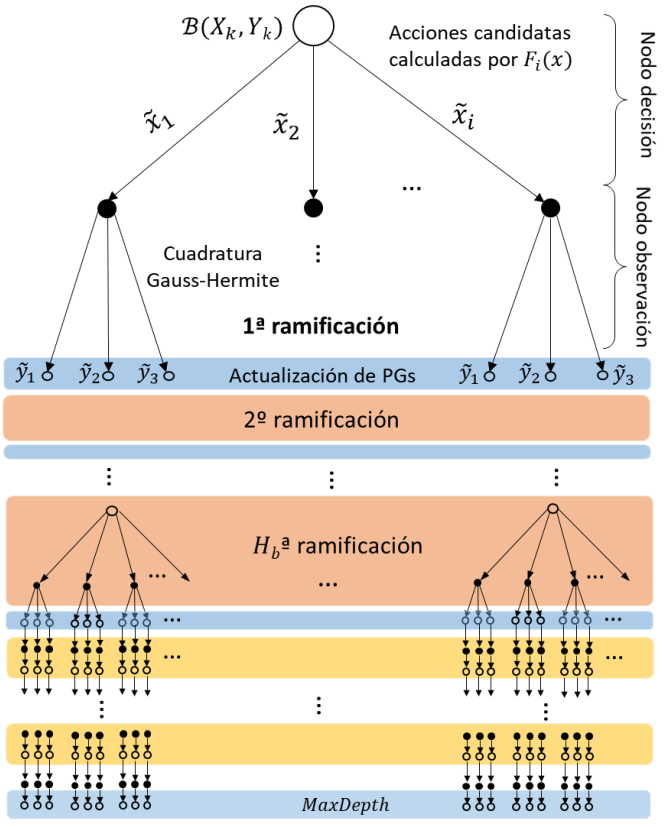

Figura 1: Esquema del árbol de escenarios generado por la estrategia de exploración propuesta.

$$Q_{\mathcal{N}} \coloneqq \sum_{j=1}^{M} \alpha_j \left(r(y_j, Y_{\mathcal{N}}) + \gamma V_{\mathcal{N}_j} \right); V_{\mathcal{N}} \coloneqq \min_{\mathcal{N}'} Q_{\mathcal{N}'} \qquad (8)$$

3.2 Algoritmo propuesto

El pseudocódigo del algoritmo que calcula una decisión x dado el conocimiento probabilístico actual del sistema o proceso f, la ejecuta experimentalmente, actualiza dicho conocimiento con la observación realizada, y repite el procedimiento hasta que el presupuesto de N decisiones se ha agotado, viene resumido en el Algoritmo 1.

Algoritmo 1. OB-POMDP

Requiere: $F_i(\beta, x), X_0, Y_0, N, J(\cdot), H_b$

1: Crear $\beta(Y_0, X_0)$ % Crea el PG semiparamétrico inicial
2: **Para** $k = 1$ **hasta** N **hacer:**
3: $b = \text{RootNode}(\beta, F_i)$ % Crea el nodo raíz
4: $MaxDepth \leftarrow N - k + 1$ % Reducción de horizonte
5: $b.\text{branch}(H_b, MaxDepth)$ % Ramificación recursiva
6: $[V, x] \leftarrow b.\text{ValueFunction}(J)$
% Calcula la función de valor por propagación inversa
7: Aplicar x, medir y % Realizar experimento real
8: $\beta \leftarrow \beta \oplus \{y, x\}$ % Actualizar el PG semiparamétrico

El algoritmo comienza creando un PG con la información disponible a priori del sistema. Con este conocimiento inicial y la lista de funciones de adquisición de OB, la clase ROOTNODE crea el nodo de decisión raíz. Los nodos creados tienen implementados dos métodos: 1) BRANCH, para construir el árbol de escenarios de forma recursiva, creando internamente otros nodos del tipo que corresponda por ramificación; y 2) VALUEFUNCTION, para evaluar cada escenario en simulación y calcular la función de valor asociada a cada nodo por propagación inversa.

De hecho, dado que las heurísticas Bayesianas F_i son razonablemente buenas y las observaciones $\tilde{y}$ del árbol de escenarios son "virtuales" (i.e. dadas por la cuadratura de Gauss-Hermite según la incertidumbre del PG y de los sensores), no se obtiene ventaja significativa de ramificar durante todo el horizonte N (Armesto et al., 2023). En consecuencia, el método BRANCH sólo ramifica como máximo hasta una profundidad H_b dada. De ahí hasta N se aplica la F_i de nodo anterior y la observación más probable según el PG $\beta_{\mathcal{N}}$ asociado a cada nodo $\mathcal{N}$.

Presentar de forma clara y entendible los detalles de implementación excedería el límite de páginas disponible y, como consideramos que no es la contribución principal porque dependen en buena parte del lenguaje de programación utilizado, decidimos omitirlos en este artículo.

Además de las funciones de adquisición F_i, horizontes, y de disponer de al menos un par de datos $\{y_0, x_0\}$ para inicializar el primer PG, hará falta disponer de un conjunto de hiperparámetros correspondientes al PG (longitud de escala, varianza estacionaria, ruido de medida, etc.).

4. Aplicación a un proceso de producción por lotes

En problemas de optimización con presupuesto finito, el Algoritmo 1 ha demostrado obtener mejor desempeño en media que las opciones de OB estándar y otras propuestas multi-etapa POMDP de la literatura (Armesto et al., 2023). Sin embargo, el banco de test realizado para dicha evaluación fue teórico: las funciones $f(x)$ desconocidas a optimizar eran PGs aleatoriamente generados con los mismos hiper-parámetros de los que dispone el algoritmo. Evidentemente, estas no son las condiciones de un caso real. En esta sección lo aplicamos a un proceso químico de producción por lotes.

4.1. Reactor de Otto-Williams

El conocido modelo propuesto por Williams y Otto en 1960 es un reactor de tanque continuamente agitado (CSTR por sus siglas en inglés) donde existen dos entradas de material reactivo: F_A que no es controlable pero sí conocido, y F_B que es variable de decisión. La temperatura del reactor T_R es la otra variable de decisión que se puede manipular para obtener una determinada calidad del producto a la salida al final del lote.

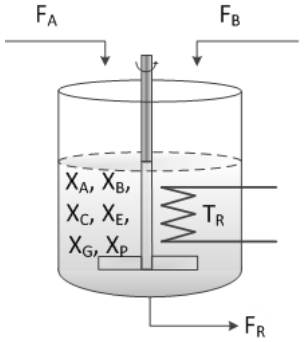

Figura 2: Diagrama simplificado del reactor de Otto-Williams.

En el interior del reactor ocurren tres reacciones químicas en paralelo, existiendo en total 6 componentes: 4 productos (C, E, G y P) y las partes de A y B que quedan por reaccionar.

$$\begin{aligned} A + B &\overset{k_1}{\rightarrow} C; \;\; k_1 = 1.66 \cdot 10^6 \cdot e^{-\frac{6666.7}{273+T_R}} \\ B + C &\overset{k_2}{\rightarrow} P + E; \;\; k_2 = 7.2177 \cdot 10^8 \cdot e^{-\frac{8333.3}{273+T_R}} \\ C + P &\overset{k_3}{\rightarrow} G; \;\; k_3 = 2.6745 \cdot 10^{12} \cdot e^{-\frac{11111}{273+T_R}} \end{aligned} \qquad (9)$$

Tabla 1: Precios, parámetros del proceso y límites de operación.

Param.	Valor	Unidad	Param.	Valor	Unidad
C_A	76.23	€/kg	F_B^{UP}	3	kg/s
C_B	114.34	€/kg	F_B^{LO}	6	kg/s
P_P	1143.38	€/kg	T_R^{UP}	100	ºC
P_E	25.92	€/kg	T_R^{LO}	70	ºC
F_A	1.8275	kg/s	V_R	2105	kg

El objetivo de la optimización estacionaria será encontrar la combinación de las dos variables de decisión (F_B, T_R), dentro de unos límites de operación, que obtenga el máximo beneficio económico acumulado al producir $N = 6$ lotes de los productos presentes en el caudal de salida F_R:

$$J \coloneqq \sum_{k=1}^{N} \gamma^k \left(\left(X_{P,k} \mathrm{P_P} + X_{E,k} \mathrm{P_E} \right) F_{R,k} - F_A \mathrm{C_A} - F_{B,k} \mathrm{C_B} \right) \quad (10)$$

En (10), $\mathrm{P_i}$ son los precios de venta de los productos y $\mathrm{C_j}$ los costes de los reactivos. Se decidió un factor de descuento $\gamma = 0.98$ para favorecer la producción de lotes de buena calidad ya desde los primeros intentos. Nótese que el beneficio (10) está definido directamente sobre caudales, por lo que el coste total necesitaría obtenerse multiplicando por la duración del lote (no incluido en la fórmula porque no influye para nada en el resultado de la optimización).

4.2. Configuración del algoritmo y resultados

El Algoritmo 1 ha sido configurado de la siguiente manera. Los hiperparámetros de la función de correlación (1) son:

$$\kappa(x_a, x_b) = M \cdot e^{-\frac{\|x_b - x_a\|^2}{2\sigma}};\ M = 25, \sigma = 0.1$$
$$\Sigma_\theta = 10/\sigma;\ \lambda = 0.25$$

Utilizamos una función de base radial estándar para la parte no paramétrica. Los valores mostrados han sido elegidos de tal forma que estiman una desviación estándar en J de 5€ alrededor de su valor medio, y una correlación de 0.13 al desplazarse un 10% sobre el rango de operación[1]. Estos valores se podrían estimar teniendo en cuenta el beneficio económico registrado de producir un lote similar en el pasado (dato histórico) y de las hojas de características de los analizadores de concentración y caudalímetros.

Como es usual para dar prioridad a la estimación del parámetro de *bias*, su varianza ha sido fijada a un valor grande, en este caso proporcional a la suavidad esperada del PG.

Además de las 4 funciones de adquisición de OB estándar mencionadas en la Sección 3.1, hemos añadido a la lista dos heurísticas con un comportamiento más exploratorio:

- Límite de confianza inferior con 3 veces la desviación estándar (el usual utiliza 2σ).
- Heurística de exploración local. Se elige entre los vértices de un hipercubo de tamaño prefijado (p.ej., la mitad del rango de las entradas) centrado en la última x testeada. El vértice seleccionado será el que se encuentre a mayor distancia del centroide formado por los datos pasados X.

El algoritmo se iniciaría desde el mejor punto de operación que se conoce por el histórico de la planta, que normalmente no debería estar muy alejado del óptimo real. En este caso $x_0 = \{F_B = 4.85, T_R = 85\}$, cuyo beneficio es $y_0 = 178.85$ €/lote. Sin embargo, el óptimo del proceso se encuentra en $x^* = \{F_B = 4.79, T_R = 89.8\}$, dando un beneficio (sin considerar ruido de medida) de $y^* = 191.23$ €/lote.

Tabla 2: Pérdida de beneficio acumulada respecto a producir lotes óptimos.

EI	PI	LCB	EV	OB-POMDP
33.9€	71.84€	40.03€	71.79€	**16.4€**

Se ha ejecutado el Algoritmo 1 con un horizonte de ramificación $H_b = 2$, y se ha comparado su desempeño con el que consigue la OB estándar con las 4 opciones funciones de adquisición listadas en la Sección 3.1. Los resultados se resumen en la Tabla 2.

Como se puede observar, el Algoritmo 1 fue capaz de obtener un mayor beneficio acumulado después de producir 6 lotes con el reactor (menor pérdida de beneficio respecto a haber conocido el óptimo de antemano). La heurística fija de mejora esperada (EI) ha resultado ser la que mejor desempeño consigue en OB estándar, incurriendo no obstante en el doble de pérdida de beneficio respecto al desempeño del Algoritmo 1. En la Figura 3 se muestran los puntos de consigna decididos para realizar cada uno de los 6 lotes de producción, y el PG (conocimiento estimado acerca del reactor) que queda tras los experimentos.

La secuencia de decisiones tomadas por el algoritmo se corresponde con las siguientes funciones de adquisición:

1. Heurística de exploración local.
2. Mejora esperada (EI).
3. Probabilidad de mejorar (PI).
4. Valor esperado (EV).
5. Valor esperado (EV).
6. Valor esperado (EV).

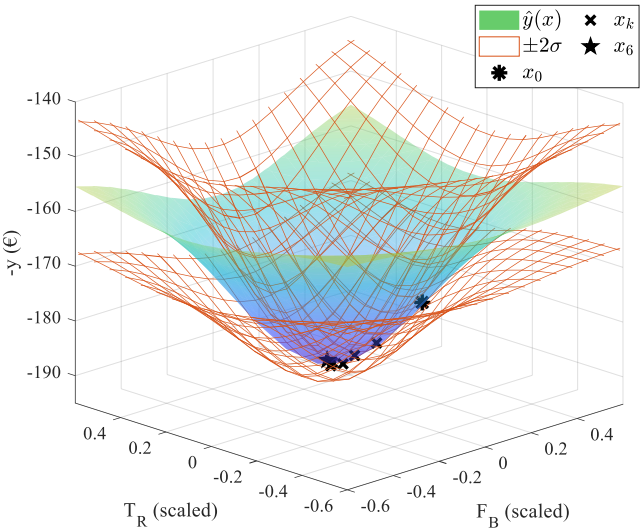

Figura 3: Ejecución de los 6 lotes por el algoritmo OB-POMDP y modelo probabilístico final construido con todas las observaciones.

5. Conclusiones

La toma óptima de decisiones en casos donde existe un presupuesto límite para experimentación requiere plantear el problema como POMDP para tratar correctamente el dilema exploración-explotación. El algoritmo OB-POMDP planteado ha demostrado que poder decidir la función de adquisición más prometedora a lo largo del horizonte de predicción mejora a las propias heurísticas miopes en OB estándar.

El algoritmo planteado para aproximar el problema POMDP es tratable computacionalmente y ha demostrado su eficacia en un problema con cierta similitud a la realidad de la industria química y de procesos.

Nótese que, aunque todos los algoritmos han partido del mismo conocimiento inicial acerca del proceso (datos x_0, y_o) y sin acceso a un modelo del reactor (sólo utilizado como caja negra en simulación actuando de "planta experimental"), los

[1] Las entradas han sido escaladas a media 0 y varianza 1.

desempeños finales registrados sólo son comparables hasta cierta tolerancia debido al ruido de medida, que limita la repetitividad de los experimentos.

Agradecimientos

Esta investigación forma parte de los proyectos LOCPU (PID2020-116585GB-I00) y a-CIDiT (PID2021-123654OB-C31) financiados por MCIN/AEI/10.13039/501100011033.

El primer autor agradece el soporte del MIU a través del Plan de Recuperación, Transformación y Resiliencia financiado por la Unión Europea – NextGenerationEU.

Referencias

Abramowitz, M., Stegun, I.A., 1972. Handbook of Mathematical Functions, 10th printing with corrections, Dover Publications, ISBN: 978-0-486-61272-0. [Equation 25.4.46]

Armesto, L., Pitarch, J.L., Sala, A., 2023. Acquisition function choice in Bayesian optimization via partially observable Markov decision process, in: Ishii, H. (Ed.), IFAC World Congress 2023, In Press.

Armesto, L., Sala, A., 2022. Volume-weighted Bellman error method for adaptive meshing in approximate dynamic programming. Revista Iberoamericana de Automática e Informática industrial, 19(1), 37–47. DOI: 10.4995/riai.2021.15698

Astudillo, R., Jiang, D., Balandat, M., Bakshy, E., Frazier, P., 2021. Multi-step budgeted Bayesian optimization with unknown evaluation costs. In: Advances in Neural Information Processing Systems, 34, 20197-20209.

Busoniu, L., Babuska, R., De Schutter, B., Ernst, D., 2017. Reinforcement learning and dynamic programming using function approximators. CRC press. DOI: 10.1201/9781439821091

Calandra, R., Seyfarth, A., Peters, J., Deisenroth, M.P., 2016. Bayesian optimization for learning gaits under uncertainty. Annals of Mathematics and Artificial Intelligence 76, 5–23. DOI: 10.1007/s10472-015-9463-9

Cressie, N., 1990. The origins of kriging. Mathematical Geolology 22, 239–252. DOI:10.1007/BF00889887

Deisenroth, M.P., Neumann, G., Peters, J., 2013. A survey on policy search for robotics. Foundations and Trends® in Robotics 2, 1–142. DOI:10.1561/2300000021

del Rio Chanona, E.A., Petsagkourakis, P., Bradford, E., Graciano, J.E.A., Chachuat, B., 2021. Real-time optimization meets Bayesian optimization and derivative-free optimization: A tale of modifier adaptation. Computers & Chemical Engineering 147, 107249. DOI: 10.1016/j.compchemeng.2021.107249

Frazier, P.I., 2018. Bayesian optimization, in: Recent advances in optimization and modeling of contemporary problems. Informs, 255–278. DOI: 10.1287/educ.2018.0188

Lam, R., Willcox, K., Wolpert, D.H., 2016. Bayesian optimization with a finite budget: An approximate dynamic programming approach. In: Advances in Neural Information Processing Systems 29, 883-891.

Rodríguez-Blanco, T., Sarabia, D., Pitarch, J.L., de Prada, C., 2017. Modifier Adaptation methodology based on transient and static measurements for RTO to cope with structural uncertainty. Computers & Chemical Engineering 106, 480–500. DOI: 10.1016/j.compchemeng.2017.07.001

Spaan, M.T.J., 2012. Partially observable Markov decision processes, In: Wiering, M., van Otterlo, M. (eds) Reinforcement Learning. Springer, pp. 387–414. DOI: 10.1007/978-3-642-27645-3_12

Wu, T., Movellan, J., 2012. Semi-parametric Gaussian process for robot system identification, IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 725-731. DOI: 10.1109/IROS.2012.6385977

Yip, W.S., Marlin, T.E., 2003. Designing plant experiments for realtime optimization systems. Control Engineering Practice 11, 837–845. Process Dynamics and Control. DOI: 10.1016/S0967-0661(02)00213-7