

## ¿OPTIMIZACIÓN POR SIMULACIÓN U OPTIMIZACIÓN PARA LA SIMULACIÓN?

Ebert Brea

Universidad Central de Venezuela  
Facultad de Ingeniería, Escuela de Ingeniería Eléctrica.  
Apartado Postal 47456, Caracas 1041-A, Venezuela.  
E-mail: ebrea@elecrisc.ing.ucv.ve



### RESUMEN

La simulación de sistemas de eventos discretos ha ocupado, en los últimos años, un importante lugar en el análisis de sistemas y en la evaluación de tomas de decisiones. Por otra parte, la optimización también ha representado ser un útil herramienta para la búsqueda de soluciones que mejoren el desempeño de los sistemas estudiados. Sin embargo, estos dos puntos de vistas inicialmente han sido desarrollados separadamente, en donde ambos campos han establecido sus linderos. No obstante, el surgimiento de nuevas aplicaciones a sistemas cada vez más complejos han requerido la vinculación de lo que antes se estudiaban por separado. La motivación este trabajo es presentar los puntos de vistas de las respuestas a las preguntas: ¿Es la simulación la que ha ayudado a la optimización de los sistemas? o por el contrario, ¿Es la optimización la que ha asistido a la experimentación en simulación?

En principio podría decirse que las respuestas de estas interrogantes son afirmativas y es en este artículo donde serán presentadas la justificación de ambas aristas a través de ejemplos prácticos y casos de estudios por una parte, y de desarrollos de métodos y técnicas, en donde se podrá apreciar que ninguno de los campos domina y, más aún, ambos coexisten y hoy en día su vinculación "por" y "para" son de gran importancia en la resolución de problemas reales.

**Palabras clave:** Optimización, simulación, experimentación en simulación.

### 1 INTRODUCCIÓN

Cuando el término simulación es nombrado, se podría preguntar: ¿simulación de que tipo? En tal sentido, en este trabajo se refiere a la simulación de sistemas de eventos discretos (SED), donde las aleatoriedades del sistema son representadas por las diferentes variables

que definen el sistema. Sin embargo, los SED pueden ser divididos en dos grandes grupos, los estáticos y los dinámicos. Un ejemplo de estos últimos sistemas, es la simulación de una agencia bancaria, en donde la llegadas de clientes así como el tipo de transacción pueden estar representadas por variables aleatorias con alguna distribución dada y donde la dinámica del sistema está representada por los instantes en que ocurren los eventos (Banks et al., 2001) y (Pritker et al., 1989). En relación a los SED estáticos, los mismos representan sistemas en donde el instante de ocurrencia de los eventos no son medidos o representados. Como ejemplos, pueden ser los modelos de simulación por Monte Carlo, donde el instante en que ocurren los eventos no son representados en el modelo.

Desde el desarrollo de la simulación de SED, ésta ha servido como soporte en la toma de decisiones sobre algún sistema bajo estudio. Pritker et al. (1989) muestra como la simulación ha sido empleada en estas situaciones. No obstante, ha emergido un enfoque adicional en donde la simulación ha servido en la optimización de sistemas. Unos de las primeras publicaciones orientadas sobre el tema es Azadivar y Talavage (1980), donde los autores proponen una simple idea de optimización por comparación de los diferentes escenarios definidos.

La idea de la comparación de diferentes escenarios fue formalmente formulada por Ho et al. (1992), donde los autores proponen un método de comparación a objeto de determinar el mejor punto de diseño a través de la comparación estadística. El método podría decirse que es un método de fuerza bruta, en donde se realizan comparaciones de los diferentes puntos de diseños, para así determinar el mejor punto de diseño.

Esto ha generado diversos enfoques a fin de buscar un método eficiente de optimización, en donde hace pensar que la optimización está al servicio de la simulación. Entre los principales enfoques están: el análisis infinitesimal de perturbación (Ho y Cao, 1991) y el método de función de puntaje propuesto por Rubinstein y Shapiro

(1993). El trabajo desarrollado por Brea (1996) presenta un método de selección para determinar el métodos de sensibilidad a objeto de estimar el gradiente de una función de desempeño y así emplearla como estimador de gradiente para ser usada en la optimización.

Adicional a esto, Fu (2002, 1994) ha presentado una revisión de los métodos más relevantes en la optimización de sistemas por simulación, y en particular, el método de optimización por respuesta de superficie, el cual es ampliamente discutido en (Myers y Montgomery, 2002). Este método consiste en determinar la respuesta de superficie empleando los esquemas de diseños de experimentos y calcular el punto estacionario a fin de establecer el óptimo del sistema. Sin embargo, Fu (1994) presenta un método de dos fases, en donde a través de un hiperplano se establece la mejor dirección que ubica la zona estacionaria y luego por diseño de experimentos se estima el modelo de segundo orden para así estimar el punto estacionario.

De lo discutido hasta este punto se ha podido apreciar que la optimización de sistemas ha sido determinada por medio de la simulación. Sin embargo, un claro ejemplo de como la optimización ha permitido mejorar la experimentación en simulación puede ser apreciada por el trabajo presentado por Chen et al. (2000), el cual será estudiado en este escrito.

Otros enfoques como el desarrollado por Shi y Ólafsson (2000b), donde una pedagógica explicación es presentada por Brea y Cheng (2003b) a un problema de optimización de cuatro colas.

Un reciente trabajo presentado por Brea (2004) muestra la potencialidad del enfoque del método de Nelder y Mead (1965) cuando es empleado a problemas con restricciones lineales y las funciones objetivos están sometidas a ruido. Este enfoque brinda un enfoque de la optimización por simulación, debido a las aleatoriedades que está sometida las funciones de desempeños de los modelos de simulación de SED.

Este artículo está estructurado como sigue: En la Sección 2 es presentado una breve descripción del método de Nelder-Mead Linealmente Restringido, en el cual se muestra su potencialidad como método de optimización de modelos de simulación. Adicionalmente, es presentado un método de optimización aleatorio para su empleo en optimización por modelos de simulación. Un método de optimización para su empleo en la realización de experimentaciones óptimas en modelos modelos de simulación es mostrado en la Sección 3. Finalmente, son presentadas las conclusiones de esta trabajo en la Sección 4.

## 2 OPTIMIZACIÓN POR SIMULACIÓN

La optimización de sistemas a través de la simulación ha sido en los últimos años un punto de vista que ha tenido

un gran desarrollo. Sin embargo, el desarrollo de métodos más eficientes ha seguido ocupando un importante lugar hoy en día.

Desde esta perspectiva, pueden ser identificado tres enfoques. Estos enfoques son: la optimización por métodos Newtonianos empleando lo que hoy en día se conoce como análisis de sensibilidad por Análisis de Perturbación Infinitesimal Ho y Cao (1991) y Glasserman (1991), y el basado en la estimación del gradiente y el hessiano es el desarrollado por Rubinstein y Shapiro (1993), llamado "Score Function". La optimización de sistema por métodos directos tales como los basados en el método de Nelder y Mead (1965), patrón de búsqueda, el cual está resumido en el tratado de Kolda et al. (2003), entre otros, ha constituido ser una poderosa herramienta de optimización. Un tercer enfoque está basado en la optimización por métodos aleatorios y uno de los más recientemente desarrollado es el presentado por Shi y Ólafsson (2000b), llamado Particiones Jerarquizadas. En esta sección serán brevemente presentados el método de optimización desarrollado por Brea (2004) y el método de Particiones Jerarquizadas presentado por Shi y Ólafsson (2000b).

### 2.1 Nelder-Mead bajo Restricciones Lineales

Sea  $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$  a función objetivo no lineal y continua, donde  $\mathbf{x} = [x_1, x_2, \dots, x_d]'$  es un vector columna de componentes  $x_1, \dots, x_d$  en el espacio Euclidiano de dimensión  $d$ , y sea  $\eta(\sigma)$  una función de ruido con valor esperado cero que depende únicamente de la desviación estándar  $\sigma$ .

Considere el problema de minimización  $\tilde{f}(\mathbf{x}) = f(\mathbf{x}) + \eta(\sigma)$  sujeto a  $\mathbf{x} \in \mathcal{F}$ , donde  $\mathcal{F}$  denota un conjunto no vacío de puntos factibles dados por  $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$ , donde  $A$  es una matriz de dimensión  $k \times d$ , y  $\mathbf{b}$  es un vector de dimensión  $k$ , a objeto de representar un conjunto de  $k$  restricciones lineales definidas por desigualdades.

Una matemática formulación del problema, puede expresarse como:

#### Problema 1

$$\min_{\mathbf{x} \in \mathbb{R}^d} \tilde{f}(\mathbf{x}), \quad (1a)$$

$$\text{sujeto a: } A\mathbf{x} \geq \mathbf{b}, \quad (1b)$$

En el problema considerado, la función de desempeño a optimizar está representada por la función  $\tilde{f}(\mathbf{x})$ .

El método desarrollado por Brea (2004), identifica el óptimo por operaciones sobre el patrón llamado *símplex*,

el cual en dos dimensiones es un triángulo, en tres dimensiones es un tetraedro, y así sucesivamente.

El método se basa en las operaciones de Nelder y Mead (1965) evitando la degeneración del simplex, cuando él colapsa sobre las fronteras de las restricciones.

A fin de presentar un resumen del método, será presentado la siguiente proposición que permitieron la extensión del método de Nelder-Mead (NM).

### 2.1.1 Operaciones de NM

Con el propósito de estudiar las principales operaciones del NM algoritmo, serán resumidas basadas en el ordenado simplex dado por

$$S_v^{[q]} = [\mathbf{x}_1^{[q]} : \mathbf{x}_2^{[q]} : \dots : \mathbf{x}_{v-1}^{[q]} : \mathbf{x}_v^{[q]}], \quad (2)$$

donde  $v$  es el número de vértices y  $\mathbf{x}_i \in \mathbb{R}^d$  es cada vértice en la  $q$ -ésima simplex, el cual son los ordenados vértices según el valor de la función objetivo  $\tilde{f}(\cdot)$  en cada vértice  $\mathbf{x}_i^{[q]}$ . Es decir,

$$\tilde{f}(\mathbf{x}_1^{[q]}) \leq \tilde{f}(\mathbf{x}_2^{[q]}) \leq \dots \leq \tilde{f}(\mathbf{x}_{v-1}^{[q]}) \leq \tilde{f}(\mathbf{x}_v^{[q]}).$$

El algoritmo opera sobre cada ordenado  $q$ -ésimo simplex  $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \dots : \mathbf{x}_{v-1} : \mathbf{x}_v]$  identificando, entre los vértices del simplex, el vértice donde ocurre el mínimo, el vértice próximo al peor y el peor vértice, es decir, donde la función objetivo es más desfavorable. Estos vértices son:  $\mathbf{x}_{min}$ ,  $\mathbf{x}_{ntw}$  y  $\mathbf{x}_{max}$ , los cuales corresponden a  $\mathbf{x}_1$ ,  $\mathbf{x}_{v-1}$  y  $\mathbf{x}_v$ , respectivamente.

#### Reflexión

La operación de reflexión arroja el llamado punto reflejado de prueba  $\mathbf{x}_{refl}$  el cual es definido por la proyección del vértice  $\mathbf{x}_{max}$  a través del centroide  $\mathbf{x}_{cen}$  de la remanente hipercara del simplex  $H_v^{[q]}$ , donde  $\mathbf{x}_{cen}$  obviamente es el centroide de  $H_v^{[q]} = \{\mathbf{x}_i \in S_v^{[q]} \mid i = 1, \dots, v-1\}$ .

Esta operación es calculada por

$$\mathbf{x}_{refl} = (1 + \alpha)\mathbf{x}_{cen} - \alpha\mathbf{x}_{max}. \quad (3)$$

donde  $\alpha = 1$  es el coeficiente de reflexión del método de NM, y el centroide  $\mathbf{x}_{cen}$  de  $H_v^{[q]}$  es calculado por

$$\mathbf{x}_{cen}^{[q]} = \frac{1}{v-1} \sum_{i=1}^{v-1} \mathbf{x}_i^{[q]},$$

#### Expansión

El punto de prueba de expansión  $\mathbf{x}_{exp}$  es calculado por medio del punto  $\mathbf{x}_{refl}$  en la dirección  $\mathbf{x}_{cen}$  y  $\mathbf{x}_{refl}$ , empleando la ecuación

$$\mathbf{x}_{exp} = (1 - \gamma)\mathbf{x}_{cen} + \gamma\mathbf{x}_{refl}. \quad (4)$$

donde  $\gamma = 2$  es el usual coeficiente de expansión del método de NM.

Una expresión alterna a (4) es obtenida por sustitución de (3) en (4), quedando así

$$\mathbf{x}_{exp} = (1 + \alpha\gamma)\mathbf{x}_{cen} - \alpha\gamma\mathbf{x}_{max}. \quad (5)$$

#### Contracción

La operación de contracción, también llamada contracción interna, da como resultado el denominado el punto de prueba  $\mathbf{x}_{cont}$ , cuyas coordenadas están definidas por el segmento  $\mathbf{x}_{max}$  y  $\mathbf{x}_{cen}$ . Este punto es calculado por

$$\mathbf{x}_{cont} = (1 - \beta)\mathbf{x}_{cen} + \beta\mathbf{x}_{max}. \quad (6)$$

donde  $\beta = 0.5$  es el coeficiente de contracción usualmente empleado en el método de NM.

Es importante mencionar que también existe la llamada operación contracción externa, pero ésta no será empleada en este método.

#### Reducción

La reducción del simplex se realiza a través del movimiento de los vértices del simplex hacia uno de sus vértices, en este caso, hacia el vértice  $\mathbf{x}_{min}$ . La operación es ejecutada por medio de la ecuación

$$\mathbf{x}_j = (1 - \delta)\mathbf{x}_{min} + \delta\mathbf{x}_j, \quad \forall j = 2, \dots, v. \quad (7)$$

donde  $\delta = 0.5$  es en normalmente empleado coeficiente de reducción.

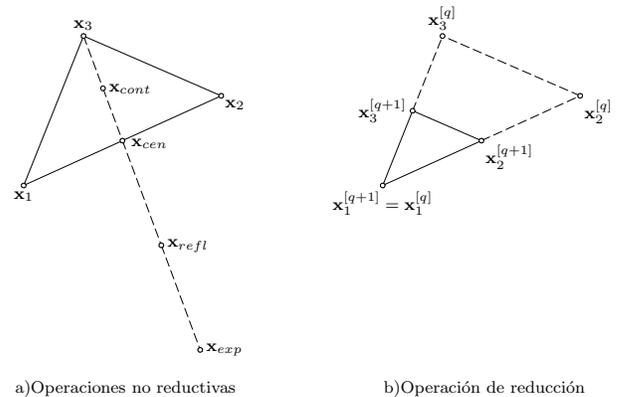


Figura 1: Operaciones en el Simplex

La Figura 1(a) presenta los puntos de pruebas que son calculados por las operaciones de reflexión, expansión y contracción. Mientras la Figura 1(b) muestra los puntos de pruebas obtenidos por la operación de reducción, de la iteración  $q$  a la  $(q+1)$ .

Estudiando las ecuaciones (3), (5) y (6), se puede obtener la ecuación general

$$\mathbf{x}_{new} = (1 - \theta)\mathbf{x}_{cen} + \theta\mathbf{x}_{max}, \quad (8)$$

donde  $\mathbf{x}_{new}$  denota el punto de prueba según la operación del método de NM, y  $\theta$  está definida por

$$\theta = \begin{cases} \alpha & \text{para un operación de reflexión,} \\ \alpha\gamma & \text{para una operación de expansión,} \\ -\beta & \text{para una operación de contracción.} \end{cases}$$

En este escrito, las operaciones de reflexión y expansión, serán denominadas operaciones externas.

### 2.1.2 El algoritmo de Nelder-Mead Linealmente Restringido

Una extensión del método de NM, llamada algoritmo de Nelder-Mead Linealmente Restringido (NMLR) fue desarrollado con el propósito de hallar una solución a los problemas propuesto por en el Problema 1. Para asegurar que todas las operaciones sobre el simplex arrojen puntos factibles fue modificada las operaciones las cuales están basadas en la proposición que será enunciada en el curso del artículo.

#### Definiciones Básicas

En este punto, serán presentados un conjunto de definiciones que ayudaran la explicación del algoritmo NMLR.

**Definición 1 (Matriz de aristas)** Sea  $E_p^{[q]}$  una matriz de dimensión  $d \times (v - 1)$ , cuyas columna  $j$ -ésima representa la arista entre los vértices  $\mathbf{x}_j$  y  $\mathbf{x}_p$  del simplex  $S_v^{[q]}$  para todo  $p \neq j = 1, 2, \dots, v$ . Dado esto se puede establecer que

$$E_p^{[q]} = [\mathbf{x}_1 - \mathbf{x}_p : \dots : \mathbf{x}_{j \neq p} - \mathbf{x}_p : \dots : \mathbf{x}_v - \mathbf{x}_p]. \quad (9)$$

**Definición 2 (Simplex completo)** Sea  $R_c(E_i^{[q]})$  el rango de columna de la matriz de arista  $i$ -ésima  $E_i^{[q]}$ . Un simplex se dice ser completo en el espacio Euclidiano de dimensión  $d$ , si se satisface que

$$\min_{i=1,2,\dots,v} [R_c(E_i^{[q]})] = d.$$

**Definición 3 (Restricción activa)** Una restricción lineal  $i$ -ésima se dice ser activa o activada por

un simplex, si todos los vértices del simplex pertenecen a la frontera de la  $i$ -ésima restricción. Matemáticamente, esta condición puede ser verificada mediante

$$\mathbf{a}'_i[\mathbf{x}_1 : \mathbf{x}_2 : \dots : \mathbf{x}_v] - b_i \mathbf{1}'_v = \mathbf{0}'_v. \quad (10)$$

donde  $\mathbf{1}'_v$  y  $\mathbf{0}'_v$  son los vectores de filas de dimensión  $v$  y cuyos elementos son unos o ceros, respectivamente.

#### Definición 4 (simplex colapsado)

Considere el caso de una variable en el espacio de dimensión  $d$ , esto es,  $\mathbf{x} \in \mathbb{R}^d$ . Un simplex de  $v$  vértices se dice estar colapsado sobre una restricción  $i$ -ésima, si todos sus vértices pertenecen a la frontera de la  $i$ -ésima restricción.

**Definición 5 (Grado de colapso)** Considere el caso de una variable en el espacio de dimensión  $d$ . Un simplex de  $v$  vértices se dice estar colapsado en grado  $r$ , si sus  $v$  vértices pertenecen a  $r$  fronteras de restricciones lineales. En consecuencia, esas  $r$  restricciones están activadas por el simplex.

**Definición 6 (Simplex degenerado)** Un simplex de  $v$  vértices se dice estar degenerado si todos sus vértices pertenecen al menos a una de sus hipercaras, sin que el simplex haya activado alguna restricción.

**Definición 7 (Menor simplex)** Considere el caso de una variable en el espacio de dimensión  $d$ . Un simplex  $S_v^{[q]}$  se dice ser menor o suficientemente definido sobre  $r$  restricciones linealmente independiente, si el número de sus vértices ( $v$ ) es igual a  $d + 1 - r$  y se cumple que

$$\min_{i=1,2,\dots,v} [R_c(E_i^{[q]})] = d - r.$$

### 2.1.3 Procedimientos

A objeto de extender el método de NM, fueron incluidos un conjunto de procedimientos, basados en las siguientes proposiciones.

#### Operaciones restringida de Nelder-Mead

Como cada punto de prueba es generado por dos puntos definidos en el simplex, es posible determinar, de donde viene y hacia donde va. Este información permite evaluar si un punto generado por alguna operación del método de NM, denotado por  $\mathbf{x}_{new}$  es factible o no.

Basado en este principio, se cuenta con la siguiente proposición, a fin de determinar el punto de prueba más lejano y factible de la operación de NM.

#### Proposición 1 (Operación externa restringida)

Sea  $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \dots : \mathbf{x}_v]$  un  $q$ -ésima simplex ordenado de  $v$  vértices tal que todos sus vértices estén ordenados según el valor de la función objetivo  $f(\mathbf{x}_j)$

$\forall j = 1, \dots, v$  y, todos sus vértices pertenezcan a la región de factibilidad  $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{A}\mathbf{x} \geq \mathbf{b}\}$ , definida por  $k$  desigualdades lineales  $\mathbf{a}'_i \mathbf{x} \geq b_i \forall i = 1, \dots, k$ . Si una operación externa del método de NM ocurre (operación de reflexión o expansión), entonces una restringida operación externa sobre un  $q$ -ésima simplex es definida por

$$\mathbf{x}_{c-new} = (1 + \min(\theta, \lambda))\mathbf{x}_{cen} - \min(\theta, \lambda)\mathbf{x}_v, \quad (11)$$

donde  $\lambda = \min(\lambda_1, \dots, \lambda_k)$  tal que  $\lambda_i \geq 0 \forall i = 1, \dots, k$ , y

$$\lambda_i = \frac{b_i - \mathbf{a}'_i \mathbf{x}_{cen}}{\mathbf{a}'_i (\mathbf{x}_{new} - \mathbf{x}_{cen})}, \quad \forall i = 1, \dots, k.$$

**Prueba.** Ver (Brea, 2004, Capítulo 3). ■

### 2.1.4 Procedimiento de espacio de intersección

Como el algoritmo NMLR está dividido en etapas, en cada una de ellas, el algoritmo NMLR requiere determinar el espacio de intersección a fin de reconstruir un nuevo simplex en ese espacio, una vez el algoritmo finalice la etapa previa. En este caso, el algoritmo rearmará nuevamente para ubicar a nuevo óptimo, y así compararlo con el óptimo obtenido en la etapa anterior para decidir si parar o continuar el algoritmo.

Este hecho condujo a la siguiente proposición.

**Proposición 2** Sea  $\mathcal{X}$  un conjunto no vacío en  $\mathbb{R}^d$  y sea  $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ . Adicionalmente sea  $l_i(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$  una  $i$ -ésima función lineal dada por  $l_i(\mathbf{x}) = \mathbf{a}'_i \mathbf{x} - b_i$  para todo  $i \in \mathcal{I}$ , donde  $\mathcal{I}$  es el conjunto de subíndices de las funciones lineales. Considere el Problema 1 de minimizar  $\tilde{f}(\mathbf{x})$  sujeto a  $\mathbf{x} \in \mathcal{X}$  y  $l_i(\mathbf{x}) \geq 0$  para cada  $i \in \mathcal{I}$ . Sea  $\mathbf{x}_{best}$  un punto factible tal que  $\mathbf{x}_{best}$  se apoya sobre todas las restricciones activas  $i$ -ésima que pertenezcan al conjunto no vacío  $\mathcal{A}(\mathbf{x}_{best}) = \{i \in \mathcal{I} \mid l_i(\mathbf{x}_{best}) = 0\}$ , y este punto fue obtenido a través de la convergencia de algún método de minimización, pero no necesariamente es un óptimo del Problema 1. Si  $\tilde{f}(\mathbf{x})$  es diferenciable en  $\mathbf{x}_{best}$  y  $\tilde{f}(\mathbf{x})$  tiene un desconocido mínimo local  $\mathbf{x}^*$  del Problema 1, tal que  $\mathcal{A}(\mathbf{x}_{best}) = \mathcal{A}(\mathbf{x}^*)$  y  $\nabla \tilde{f}(\mathbf{x}_{best})' \nabla \tilde{f}(\mathbf{x}^*) > 0$ , entonces el desconocido mínimo local  $\mathbf{x}^*$  del Problema 1 está ubicado en el espacio de intersección de las restricciones lineales activadas por  $\mathbf{x}_{best}$ , tal que

$$\sum_{i \in \mathcal{A}_p(\mathbf{x}_{best})} \tilde{u}_i \nabla \tilde{f}(\mathbf{x}_{best})' \mathbf{a}_i > 0, \quad (12)$$

donde  $\tilde{u}_i \geq 0$  para todo  $i \in \mathcal{A}(\mathbf{x}^*) = \mathcal{A}(\mathbf{x}_{best})$  son los multiplicadores de Lagrange.

**Prueba.** Ver (Brea, 2004, Capítulo 3). ■

Basado en la Proposición 2, se enunció el siguiente criterio de selección de restricciones lineales, a fin de construir un menor simplex en el espacio de intersección.

#### Criterio 1 (Selección de restricciones activas)

Sea  $\mathcal{A}(\mathbf{x}_{best})$  un conjunto de desigualdades lineales activadas por  $\mathbf{x}_{best}$ . Sea  $\mathcal{A}_{\nabla f}$  un subconjunto de  $\mathcal{A}(\mathbf{x}_{best})$  tal que el producto escalar  $\nabla f(\mathbf{x}_{best})' \mathbf{a}_i > 0$  para todo  $i \in \mathcal{A}(\mathbf{x}_{best})$ . Selecciones todas la desigualdades lineales de  $\mathcal{A}_{\nabla f}$  a objeto de identificar el espacio de intersección que satisfaga la Proposition 2.

Suponga ahora que se tiene  $r$  restricciones activadas de las  $k$  restricciones dadas por (1b) como resultado de aplicar el Criterio 1 u otro criterio de selección de restricciones activadas por  $\mathbf{x}_{best}$ . Esto significa en el contexto del método de NM, que todos los vértices del actual simplex convergieron al punto  $\mathbf{x}_{best}$ , el cual podría ser un mínimo o no. Dado esto, se define  $\mathcal{A} = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$  como el conjunto de subíndices de las restricciones activas, cuyo elemento  $\alpha_i$  es seleccionado de las desigualdades (1b).

#### Proposición 3 (Espacio de Intersección)

Sea

$$A_{r \times d} \mathbf{x}_{d \times 1} = \mathbf{b}_{r \times 1}, \quad (13)$$

el sistema de ecuaciones definidas por las restricciones activas  $\mathcal{A} = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$  las cuales fueron seleccionadas por el Criterio 1 u otro criterio, donde  $\text{rank}(A_{r \times d}) = r$ . Entonces el espacio de intersección que satisface (13) está definido por el conjunto de puntos  $\mathbf{x} = [x_1, x_2, \dots, x_d]'$  tal que sus  $r$  componentes  $x$  de  $\mathbf{x}$  dependen de  $(d - r)$  componentes independientes  $x$  de  $\mathbf{x}$ .

**Prueba.** Ver (Brea, 2004, Capítulo 3). ■

### 2.1.5 Idea básica del algoritmo

El algoritmo inicia con un punto factible, el cual es empleado para la construcción de un simplex completo factible. Dado este simplex inicial, la función objetivo  $\tilde{f}(\mathbf{x}_i)$  es evaluada en cada  $i$ -ésimo vértice del actual simplex, para luego ordenar los vértices según el valor de la función objetivo  $\tilde{f}(\mathbf{x}_i)$ .

Dado  $S_v^{[0]}$ , es ejecutada una operación del algoritmo con el propósito de determinar si reemplazar el peor vértice del simplex por el nuevo punto de acuerdo a la lógica del método de NM.

Como el simplex es transformado en cada iteración por una operación, es necesario chequear si el actual simplex ha activado alguna restricción lineal o no, en término de la Definición 3. En el caso que ocurra la activación de  $r$  restricciones, el algoritmo ajusta su número de vértices ( $v$ ) a  $d + 1 - r$ , mediante la remoción de los  $r$  peor vértices del simplex colapsado. De esta manera,

el algoritmo podría reducir el número de evaluaciones de la función objetivo.

Si el símplex converge sobre la frontera de  $r$  restricciones lineales a un punto  $\mathbf{x}_{opt}^{[q]}$ , el algoritmo identifica dentro de las restricciones activadas aquellas cuyos productos escalares  $\nabla \tilde{f}(\mathbf{x}_{opt}^{[q]})^T \mathbf{a}_i$  sean positivos. Dado este conjunto de restricciones, el algoritmo construye un nuevo símplex en el espacio de intersección definido por las fronteras de las restricciones, de acuerdo a la Proposición 3. Así, el algoritmo, inicia la siguiente etapa ( $s+1$ ), en el espacio de intersección.

Cuando el espacio de intersección esta definido por un número de restricciones mayor o igual a la dimensión  $d$ , el algoritmo construye un símplex completo cercano a el punto de convergencia  $\mathbf{x}_{opt}^{[s]}$ .

El algoritmo cuenta con dos reglas de paradas. El primero mide en cada iteración si la mayor arista del símplex, definida por  $\Delta_{ij} = \max_{i \neq j} \|\mathbf{x}_i - \mathbf{x}_j\|$ , es menor que una tolerancia dada. La segunda regla compara la distancia entre los dos últimos puntos de convergencia de dos sucesivas etapas, con un error especificado. Es decir, si  $\|\mathbf{x}_{opt}^{[s]} - \mathbf{x}_{opt}^{[s-1]}\| < \delta$  el algoritmo para.

Por otra parte, el algoritmo pasa a la siguiente etapa si el grado de colapso del símplex es mayor a uno, en caso contrario, el algoritmo supone que el punto de convergencia es el óptimo.

Este criterio tiene la ventaja de reducir el número de exploraciones en la región de factibilidad. No obstante, el criterio de parada podría satisfacer la convergencia a un óptimo local o global, que cumpla las condiciones de Kuhn-Tucker.

Es importante destacar, que el algoritmo emplea el método de construcción propuesto por Humphrey y Wilson (2000). Sin embargo, Walters et al. (1991) muestra varios métodos de construcción, en donde presenta las ventajas y desventajas de cada uno.

Brea y Cheng (2003a) presenta un diagrama de flujo del algoritmo aquí introducido.

### 2.1.6 Ejemplos numéricos

En este grupo de ejemplos, fue ejecutado el algoritmo de NMLR a fin de representar las curvas de frecuencias acumulativas de la distancia al punto cierto, denotada como DTP, para un grupo de problemas de minimización de la función de Wood afectadas con ruido.

Sea  $f_w(\mathbf{x}) : \mathbb{R}^4 \rightarrow \mathbb{R}$  la función de Wood, cuya expresión matemática es dada por

$$f_w(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10(x_2 + x_4 - 2)^2 + 10(x_2 - x_4)^2,$$

Sea  $\mathcal{F}_o$  una región de factibilidad con ángulo sólido obtuso  $\mathbf{x} \in \mathbb{R}^4$  tal que

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 &\geq b_1, \\ 2x_1 + x_2 + x_3 + x_4 &\geq b_2, \end{aligned}$$

Sea  $\mathcal{F}_s$  una región de factibilidad con ángulo sólido agudo  $\mathbf{x} \in \mathbb{R}^4$  tal que

$$\begin{aligned} 2x_1 - x_2 + x_3 + x_4 &\geq b_1, \\ -x_1 + 2x_2 + x_3 + x_4 &\geq b_2, \end{aligned}$$

y sea  $\eta(\sigma)$  una variable aleatoria con distribución Gaussiana de media cero y varianza  $\sigma^2$ .

Para cada ejemplo, fue considerado el punto inicial  $\mathbf{x}'_0 = (10, 10, 10, -10)$ , y el óptimo de cada ejemplo está ubicado en  $\mathbf{x}'_{opt} = (1, 1, 1, 1)$ , el cual, dependiendo del problema, está dentro o sobre la frontera de la región.

A fin de obtener un número suficiente de muestras de los ejemplos, fue ejecutado 100 réplicas cada una con diferentes secuencias de números pseudoaleatorios y un nivel de ruido dado por  $\sigma$  igual a 0.1, 0.5, 1, 5 y 10.

Con respecto a los parámetros para la ejecución del algoritmo, fueron fijados los siguientes:  $\alpha = 0.95$ ,  $\beta = 0.5$ ,  $\gamma = 2$ ,  $\delta = 0.5$  y  $\rho = 0.99$ . El parámetro de tamaño del símplex inicial  $\tau$  fue fijado a 1 empleando el método de construcción de Humphrey y Wilson (2000).

#### Ejemplo 1 (Óptimo en la frontera de $\mathcal{F}_o$ )

$$\min_{\mathbf{x} \in \mathbb{R}^4} [f_w(\mathbf{x}) + \eta(\sigma)]$$

sujeto a  $\{\mathbf{x} \in \mathcal{F}_o \mid b_1 = 3 \wedge b_2 = 5\}$

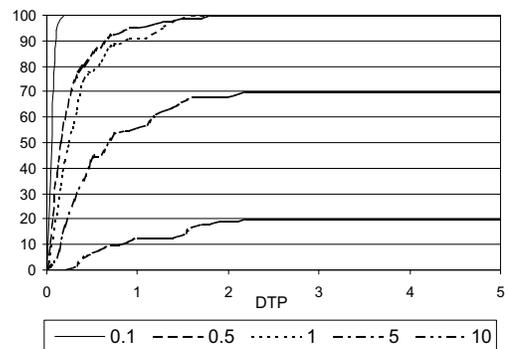


Figura 2: Frecuencias acumulativas para el ejemplo 1

#### Ejemplo 2 (Óptimo dentro de la región $\mathcal{F}_o$ )

$$\min_{\mathbf{x} \in \mathbb{R}^4} [f_w(\mathbf{x}) + \eta(\sigma)]$$

sujeto a  $\{\mathbf{x} \in \mathcal{F}_o \mid b_1 = -3 \wedge b_2 = -5\}$

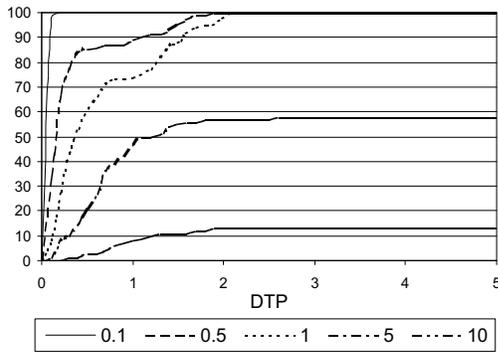


Figura 3: Frecuencias acumulativas para el ejemplo 2

Las Figuras 2 y 3 muestran las curvas de frecuencias acumulativas de DTP para 100 muestras, cuando la región de factibilidad tiene un ángulo sólido obtuso. Note que el algoritmo de NMLR identifica el óptimo en un gran número de muestras, aún cuando la función objetivo es afectada por ruido. Obviamente, este hecho depende del nivel de ruido dado por  $\sigma$ .

### Ejemplo 3 (Óptimo en la frontera de $\mathcal{F}_s$ )

$$\min_{\mathbf{x} \in \mathbb{R}^4} [f_w(\mathbf{x}) + \eta(\sigma)]$$

sujeto a  $\{\mathbf{x} \in \mathcal{F}_s \mid b_1 = 3 \wedge b_2 = 3\}$

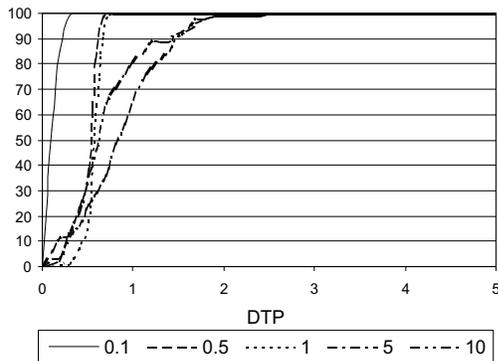


Figura 4: Frecuencias acumulativas para el ejemplo 3

### Ejemplo 4 (Óptimo dentro de la región $\mathcal{F}_s$ )

$$\min_{\mathbf{x} \in \mathbb{R}^4} [f_w(\mathbf{x}) + \eta(\sigma)]$$

sujeto a  $\{\mathbf{x} \in \mathcal{F}_s \mid b_1 = -2 \wedge b_2 = -2\}$

Como puede apreciarse de las Figuras 4 y 5, el algoritmo de NMLR evidentemente identifica mejor el óptimo cuando el nivel de ruido es bajo, que cuando es alto. Adicionalmente, se puede apreciar que el algoritmo

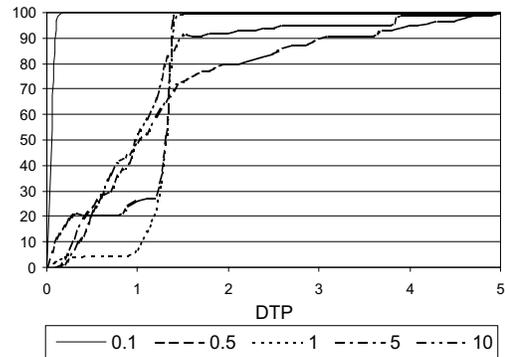


Figura 5: Frecuencias acumulativas para el ejemplo 4

trabaja adecuadamente a situaciones donde la función objetivo es alterada por ruido.

Para esta grupo de ejemplos, el algoritmo de NMLR tiene un mejor comportamiento cuando el ángulo sólido de la región es agudo, a cuando es obtuso.

## 2.2 Método de Particiones Jerarquizadas

De acuerdo a Boender y Romeijn (1995), los métodos de búsquedas aleatorios pueden ser definidos como un método algorítmico que optimiza una función objetivo a través de puntos de pruebas, cuya generación ha sido previamente especificada por alguna distribución probabilística dentro de la región de factibilidad.

Este tipo de método de optimización está basado en la generación de un punto de prueba factible a partir de una distribución probabilística, definida por una secuencia de puntos independientes e idénticamente distribuida. Como ejemplo de estos métodos se pueden nombrar: el método de pura búsqueda aleatoria, búsqueda aleatoria, búsqueda aleatoria adaptativa, entre otros (Boender y Romeijn, 1995). Otro método ampliamente conocido es el método de Monte Carlo, el cual es ampliamente descrito en (Rubinstein, 1992).

En particular, el método de Particiones Jerarquizadas (PJ) de Shi y Ólafsson (2000a) es un tipo de método de búsqueda aleatorio. Este método fue ampliamente estudiado por Brea (2002), el cual es presentado a continuación.

El método de PJ propuesto por Shi y Ólafsson (2000a) fue primeramente desarrollado para resolver problemas de optimización global en modelos deterministas. Una modificación del método para la resolución de problemas a adjudicación de recursos en SED es discutido en (Shi y Chen, 2000). El método sistemáticamente divide o particiona la región de factibilidad y concentra la búsqueda en la región más prometedora. La inspección de la región más prometedora la

realiza a través de un muestreo aleatorio en la región de factibilidad considerada. Más aún, el método de PJ ofrece convergencia con probabilidad uno en un tiempo finito, sin necesidad de verificar toda la región de factibilidad (Shi y Ólafsson, 2000b).

El algoritmo de PJ cuenta con cuatro principales pasos en cada iteración. Estos pasos son: partición, muestreo aleatorio, ordenamiento y selección del mejor punto de diseño, y por último, retroceso o parada del algoritmo.

Un práctico ejemplo del método es mostrado en (Brea y Cheng, 2003b), donde es aplicado el método a un problema de optimización de una red de cuatro colas

### a) Partición

El primer paso dentro de la iteración es la partición de la región prometedor  $\sigma(k)$  en  $M_k$  subregiones  $\sigma_i(k)$ ,  $i = 1, \dots, M_k$ , y agregar las sobrantes subregiones o también llamado conjunto complemento  $\sigma(k)$ , y denotado por  $\Theta \setminus \sigma(k)$ , en un sólo conjunto. En consecuencia, dentro de cada  $k$ -ésima iteración, se tiene  $M_k + 1$  conjuntos disjuntos que definen la región de factibilidad  $\Theta$ . El método de partición empleado en este paso es muy importante, debido a la tasa de convergencia del algoritmo, el cual dependerá de la manera de como los diferentes puntos de diseños del problema son agrupados. Estos grupos de soluciones factibles, pueden ayudar a concentrar la búsqueda en la región más prometedora más rápidamente.

### b) Muestreo Aleatorio

En este segundo paso, el algoritmo selecciona aleatoriamente una muestra de cada subregión o subconjunto  $\sigma_i(k)$ ,  $i = 1, \dots, M_k$  y del conjunto complemento  $\sigma_{M_k+1}(k) = \Theta \setminus \sigma(k)$ . En este paso, el método es suficientemente flexible, por cuanto el método puede emplear cualquier esquema de muestreo, bien sea, soportándose del conocimiento del sistema, o de cualquier heurística. Una amplia discusión del empleo de muestreo ponderado es presentado por Shi y Chen (2000).

### c) Ordenamiento y Selección

El tercer paso es la selección de la región más prometedora, entre las subregiones  $\sigma_i(k)$ ,  $i = 1, \dots, M_k$  y la región complemento  $\sigma_{M_k+1}(k)$ , mediante la comparación de la medida de desempeño (función objetivo) de cada punto de diseño.

El método estima un índice  $i = 1, \dots, M_k + 1$  de la región más prometedora. Aquí el método de PJ ofrece flexibilidad en determinar tal índice. Una manera es a través de comparación estadística de todos los puntos

de diseño. De esta forma, el método determina a que subregión o subconjunto pertenece el mejor punto de diseño. Una mejora del método de PJ fue propuesta por Shi y Chen (2000) basada en la adjudicación de muestras por punto de diseño (Chen et al., 2000).

### d) Más Particiones, Retroceso o Parada

El cuarto paso del método es para ejecutar la partición de la subregión o subconjunto que resultó más prometedora, el cual es denotado por  $\sigma_B(k)$ , donde el subíndice  $B$  significa el subregión más prometedora, a fin de concentrar la búsqueda en esa subregión. Si la mejor subregión es el conjunto complemento, el método retrocede a una subregión más grande y que contenga la subregión  $\sigma_B(k)$ , la cual es llamada superregión de  $\sigma(k)$ . Dos reglas de retrocesos son sugeridas por Shi y Chen (2000). Adicionalmente, existe a regla de parada del algoritmo propuesta por Shi y Ólafsson (2000b), basada en el enfoque de cadena de Markov. Esto es demostrado cuando existe un único óptimo del problema.

#### 2.2.1 Implementación del algoritmo

Un limitado estudio del desempeño del algoritmo fue ejecutado a problemas de optimización, donde las variables de decisiones pueden ser de naturaleza continua o discreta. Para hacer esto, fue implementado el algoritmo usando el Visual C++ Sphar (1999).

La implementación del algoritmo fue una variante del original método de PJ, dado que fue requerido un programa que permita la optimización de problemas asociados a variables de decisiones continuas y discretas. En el caso de variables continuas, el método alternativamente divide cada rango de variable en un número preespecificado de subregiones  $N_{subrange}$ , haciendo aproximadamente el mismo número de particiones para cada rango de variable. Con respecto al muestreo de puntos de diseño, fue empleado un muestreo uniforme.

Sin perder generalidad, será descrito el método a través de un problema con dos variables continuas de decisión  $\theta_1$  y  $\theta_2$ , las cuales toman valores entre  $L_i$  y  $U_i$  para  $i = 1, 2$ , y se escoge un número de subregiones  $N_{subrange} = 2$ . En este caso, la región de factibilidad original es

$$\sigma(0) = \{\theta \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq U_1, L_2 \leq \theta_2 \leq U_2\}$$

Para la región considerada, fue supuesto dos subregiones  $\sigma_1(0)$  y  $\sigma_2(0)$  expresadas por

$$\begin{aligned} \sigma_1(0) &= \{\theta \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq m_1, L_2 \leq \theta_2 \leq U_2\} \\ \sigma_2(0) &= \{\theta \in \mathbb{R}^2 \mid m_1 < \theta_1 \leq U_1, L_2 \leq \theta_2 \leq U_2\} \end{aligned}$$

donde  $m_1$  es el punto medio entre  $L_1$  y  $U_1$ .

Suponga que después de una iteración, la región prometedora fue  $\sigma_1(0)$ . Esta es entonces, dividida en dos subregiones nuevamente, pero en la dirección  $\theta_2$ , que es:

$$\begin{aligned}\sigma_1(1) &= \{\theta \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq m_1, L_2 \leq \theta_2 \leq m_2\} \\ \sigma_2(1) &= \{\theta \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq m_1, m_2 < \theta_2 \leq U_2\},\end{aligned}$$

donde  $m_2$  es el punto medio entre  $L_2$  y  $U_2$ . La subregión entonces es dada por

$$\sigma_3(1) = \{\theta \in \mathbb{R}^2 \mid m_1 < \theta_1 \leq U_1, L_2 \leq \theta_2 \leq U_2\}$$

Este proceso de repetido alternativamente para  $\theta_1$  y  $\theta_2$ , hasta que el tamaño de los lados de la región prometedora cumpla con un valor de tolerancia, definido por el usuario. En el caso que el algoritmo selecciones como región prometedora, el conjunto complemento, el algoritmo retrocedo a la inicial región.

En el caso de coexistir variables de decisiones, continuas y discretas, el algoritmo trabaja de la misma manera de acuerdo al método de PJ.

## 2.2.2 Ejemplos numéricos

Con el propósito de medir la potencialidad del método, fueron ejecutados dos conjuntos de experimentos. El primero consideró el caso donde las variables de decisiones son discretas, mientras el segundo consideró únicamente variables de decisiones continuas. Para esto fue estudiado el siguiente problema:

$$\begin{aligned}\min_{\theta} \quad & \theta^T Q \theta, \\ \text{sujeto a:} \quad & \theta \in [-10, 10]^n,\end{aligned}$$

donde la matriz  $Q$  es una matriz diagonal cuyos elementos  $q_{ii} = i \forall i = 1, \dots, n$ , es decir,  $diag(1, \dots, n)$ .

Note que el mínimo global está ubicado en el origen de coordenadas.

El número de muestras por subregión (m/r) y el número de muestras para el conjunto complemento (m/R) que fueron fijados para cada caso de estudio, son presentados en las tablas, además del número de retrocesos denotado como Ret y el número de evaluaciones de la función objetivo (m) o muestra empleadas por el algoritmo.

### a) Caso: Variables de decisiones discretas

La Tabla 1 muestra un resumen de los resultados obtenidos para esta experimentación, incluyendo el número total de evaluaciones de la función objetivo y el número de veces que el algoritmo retrocede. Observe

que el máximo número de puntos de diseños es igual a  $21^n$ , y su convergencia fue próxima a su mínimo global, en la mayoría de los ejemplos.

Tabla 1: Resumen de Caso Discreto

d	m/r	m/R	m	Ret	$\theta_1$	$\theta_2$	$\theta_3$	$\theta_4$
2	5	5	126	0	0	0	0	0
2	10	5	231	0	0	0	0	0
3	5	2	1165	4	-2	-2	0	0
3	5	2	5359	22	1	0	0	0
3	10	2	1772	3	1	0	0	0
3	10	2	13290	29	0	1	0	0
4	5	2	1631	4	-3	-2	0	0
4	5	2	2922	9	-1	0	0	0
4	10	2	5079	7	1	-1	0	0
4	10	2	11514	17	-1	0	0	0

### b) Caso: Variables de decisiones continuas

En este caso, el máximo tamaño de lado fue fijado a 0.01 a fin de ejecutar los experimentos.

La Tabla 2 muestra un resumen de los resultados arrojados por el algoritmo para cada escenario de dimensión  $n$  y dos replicas por escenario. De la tabla puede ser apreciado la gran cantidad de evaluaciones de la función objetivo, lo cual resulta costoso cuando estas evaluaciones deben hacerse a través de un modelo de simulación.

Tabla 2: Resumen de Caso Continuo

d	m/r	m/R	m	Ret	$\theta_1$	$\theta_2$	$\theta_3$	$\theta_4$	$\theta_5$
2	50	50	2800	0	0,09	0,00			
2	50	50	2800	0	0,11	0,00			
3	50	50	5409	2	0,26	0,06	0,00		
3	50	50	7212	4	-0,08	-0,02	0,00		
4	50	50	54723	39	0,76	1,73	-0,03	0,00	
4	50	50	52518	35	0,06	0,99	-0,05	0,00	
5	50	50	255499	147	0,36	0,97	-0,59	-0,01	0,00
5	50	50	519809	310	-3,13	-1,65	-0,36	0,22	0,00

Como resultado de los ejemplos estudiados por Brea (2002), podría decirse que el algoritmo de PJ no es suficientemente bueno para ser aplicado a optimización de sistemas por simulación, debido a la gran cantidad de evaluaciones de la función objetivo que debe realizar.

Más aún, en el caso de variables de decisiones continuas, el algoritmo de PJ evalúa la función objetivo  $M + m \cdot N_{subrango}$ , donde  $M$  es el número de muestras por conjunto complemento,  $m$  es el número de muestras por subregión y  $N_{subrango}$  es el pre-especificado número particiones por variable.

Este hecho es evidenciado por Shi y Ólafsson (2000a), donde los autores reportan los resultados de un problema de optimización de la función de Goldstein-Price con restricciones, cuya expresión analítica es dada por

$$f(\theta_1, \theta_2) = (1 + f_1 f_2)(30 + f_3 f_4) \quad \forall \theta \in \mathbb{R}^2, \quad (14)$$

donde  $f_i \forall i = 1, \dots, 4$  están dadas por

$$\begin{aligned}
f_1 &= (\theta_1 + \theta_2 + 1)^2 \\
f_2 &= (19 - 14\theta_1 + 3\theta_1^2 - 14\theta_2 + 6\theta_1\theta_2 + 3\theta_2^2) \\
f_3 &= (2\theta_1 - 3\theta_2)^2 \\
f_4 &= (18 - 32\theta_1 + 12\theta_1^2 + 48\theta_2 - 36\theta_1\theta_2 + 27\theta_2^2)
\end{aligned}$$

El problema propuesto por Shi y Ólafsson (2000a) es definido por

$$\begin{aligned}
&\min_{\theta} f(\theta_1, \theta_2), \\
\text{sujeto a: } &-2.5 \leq \theta_i \leq 2.5 \quad \forall i = 1, 2,
\end{aligned}$$

donde  $f(\theta_1, \theta_2)$  es dada por (14). Según los resultado numéricos reportados por Shi y Ólafsson (2000a), el método de PJ requiere entre 617 y 68900 evaluaciones de la función en promedio en 100 replicas del problema, para un pre-especificado número de subregiones de 4, un muestreo de 25, 35 y 45, y una tolerancia de 0.1, 0.01 y 0.001.

Aún cuando el método luce costoso desde la perspectiva del número de evaluaciones de la función objetivo, podría ser empleado como una herramienta de identificación de óptimos, cuando la función objetivo presenta diversos mínimos locales, dado su potencialidad de identificar la región prometedora.

### 3 OPTIMIZANDO LA SIMULACIÓN

El problema planteado por Chen et al. (2000), está fundamentalmente centrado en la experimentación optima en simulación, cuando se debe comparar varios puntos de diseños.

Sin embargo, antes de ver el planteamiento del problema desde una perspectiva matemática, se presentará el problema que debe enfrentar el experimentador del modelo de simulación. Cuando el experimentador de un modelo de simulación desea medir algún índice de desempeño del sistema representado, debe realizar un número importante de corridas de simulación a fin de obtener una muestra representativa para el punto de diseño, y sólo así estimaría las cantidades estadísticas asociadas a ese punto de diseño. Es por esta razón que en simulación son requeridas un gran esfuerzo computacional, a fin de estimar el índice de desempeño en el modelo de simulación para cada punto de diseño.

Basado es esta idea, la optimización ordinal compara los distintos puntos de diseños a fin de determinar el mejor de ellos. No obstante, el punto de vista presentado por Chen et al. (2000), proporciona una optimización en la experimentación en modelos de simulación. Cuando son realizadas en simulación de SED.

El problema formulado por Chen et al. (2000) es

**Problema 2** Sean los diferentes puntos de diseños identificados como  $i$ , los cuales definen los valores de las variables de decisiones que están en un modelo de simulación. Esto significa, que para un modelos de simulación con  $k$  puntos de diseños  $E_i$  se tiene una medida de desempeño dado por la variable  $\tilde{J}_i$  y cuyo valor esperado representa índice de desempeño  $\bar{J}_i$ . El problema de optimización de experimentación está dado como:

$$\max_{N_1, \dots, N_k} P\{SC\} \quad (15a)$$

$$\text{sujeto a: } \sum_{i=1}^k N_i = T, \quad (15b)$$

donde  $SC$ , es el evento de seleccionar correctamente el mejor punto de diseño  $E_i$  y  $N_i$  es el número de corridas que deben realizarse para cada punto de diseño  $i$ .

La probabilidad del evento  $SC$ , está dada por

$$P\{SC\} = P\left\{ \bigcap_{i=1, i \neq b}^k (\tilde{J}_b - \tilde{J}_i < 0) \right\} \quad (16)$$

Empleando la desigualdad de Bonferroni, la Ecuación (16) puede ser aproximada, resultando

$$\max_{N_1, \dots, N_k} 1 - \sum_{i=1, i \neq b}^k p\{\tilde{J}_b > \tilde{J}_i\}, \quad (17)$$

donde  $\tilde{J}_i$  está distribuida según una normal  $N(\bar{J}_i, \frac{\sigma_i^2}{N_i})$

Para la solución de este problema se debe tomar en cuenta los siguientes supuestos.

La solución a este problema de optimización vendría dado por el siguiente teorema.

**Teorema 1** Dado un número total de replicas o muestras de simulaciones  $T$  a ser adjudicados a  $k$  puntos de diseños  $E_i$  y cuyo índice de desempeño es medido por  $\bar{J}_i$  y varianza  $\sigma_i^2$  para todo  $i = 1, \dots, k$  respectivamente. Cuando  $T \rightarrow \infty$ , la aproximada probabilidad de seleccionar correctamente es asintóticamente maximizada cuando

$$\frac{N_i}{N_j} = \left( \frac{\sigma_i / \delta_{b,i}}{\sigma_j / \delta_{b,j}} \right)^2 \quad \forall i, j \in \{1, 2, \dots, k\} \wedge i \neq j \neq b \quad (18)$$

$$N_b = \sigma_b \sqrt{\sum_{i=1, i \neq b}^k \frac{N_i^2}{\sigma_i^2}}, \quad (19)$$

donde  $N_i$  es el número de muestras o replicas,  $\delta_{b,i} = \bar{J}_b - \bar{J}_i$  y  $\bar{J}_b = \min_i \bar{J}_i$

**Prueba.** Ver Chen et al. (2000). ■

Basado en el Teorema 1, queda definido el siguiente algoritmo a fin de optimizar la adjudicación del número de corridas de simulación.

**Algoritmo 1** [Chen, Lin, Yücesan y Chick]  
Dado  $k$  puntos de diseños  $E_i$  para  $i = 1, 2, \dots, k$ , y un número total de corridas de simulación  $T$ , ejecute

1. INITIALIZATION

Inicialice el contador de iteraciones  $q$  en cero y ejecute  $n_0$  replicas con distintas secuencias aleatorias para todo los puntos de diseños, es decir,  $N_1^q = N_2^q = \dots = N_k^q = n_0$  y vaya a 2.

2. CRITERIO DE PARADA

Si  $\sum_{i=2}^k N_i^q \geq T$ , entonces pare el algoritmo, de lo contrario vaya a 3.

3. CÁLCULO DE CORRIDAS

Basado en la estimación de  $\sigma_i^2$  para todo  $i = 1, 2, \dots, k$ , calcule el nuevo número de corridas  $N_i^{q+1} \forall i = 1, 2, \dots, k$ , empleando el Teorema 1.

4. SIMULACIONES

Para cada punto de diseño  $E_i$ , ejecute  $\max[0, N_i^{q+1} - N_i^q]$  corridas de simulación. Incremente el contador de iteraciones  $q$  en uno, y vaya a 2.

El Algoritmo 1 provee un método sistemática para la adjudicación de número de replicas para cada punto de diseño, permitiendo optimizar la probabilidad de seleccionar correctamente el mejor punto de diseño y de esta manera mejorar la etapa de experimentación. Según Chen et al. (2000) el algoritmo ha resultado ser eficiente también en situaciones donde la medida de desempeño se comporta de manera no normalmente distribuida.

## 4 CONCLUSIONES

Se ha podido apreciar a lo largo de este artículo que tanto la optimización como la simulación, representan campos complementarios, que pueden coadyuvar a la mejora de sistemas complejos. Por otra parte, ambos campos han servido en el análisis de sistemas, sin que ninguno de ellos dominen uno sobre el otro.

## REFERENCIAS

Azadivar, F. and J. Talavage (1980). Optimization of stochastic simulation models. *Mathematics and Computers in Simulation* 22, 231–241.

Banks, J., J. S. Carson, B. L. Nelson, and D. M. Nicol (2001). *Discrete-event system simulation* (3rd ed.). Prentice-Hall Internat. Ser. in Indust. and systems Engng. Upper Saddle River, NJ: Prentice Hall.

Boender, C. G. E. and H. E. Romeijn (1995). Stochastic methods. In R. Horst and P. M. Pardalos (Eds.), *Handbook of global Optimization*, Volume 2 of *Non-convex optimization and its applications*, pp. 829–869. Dordrecht ; Boston: Kluwer Academic Publishers.

Brea, E. (1996). *Comparación de métodos para el análisis de sensibilidad en modelos de simulación de sistemas de eventos discretos*. Caracas: Universidad Central de Venezuela. Trabajo de Grado para optar al título de MSc en Investigación de Operaciones.

Brea, E. (2002). Identification of optimum operation in simulation models. Transfer thesis, University of Southampton, Faculty of Mathematical Studies, Department of Mathematics, Graduate School in Mathematical Studies.

Brea, E. (2004). *Nelder-Mead Optimization Under Linear Constraints*. Ph. D. thesis, University of Southampton.

Brea, E. and R. C. H. Cheng (2003a). Constrained simulation optimization. In D. Al-Dabass (Ed.), *UK-SIM 2003 Sixth National Conf. of the United Kingdom Simulation Soc.*, Cambridge, UK, pp. 99–105. United Kingdom Simulation Society.

Brea, E. and R. C. H. Cheng (2003b). Optimisation of four queue network via Nested Partition method. *Internat. J. of Simulation: Systems, Sci. and Tech.* 4(1/2), 23–28.

Chen, C.-H., J. Lin, E. Yücesan, and S. E. Chick (2000). Simulation budget allocation for further enhancing the efficiency of ordinal optimization. *Discrete Event Dynam. Systems: Theory and Applications* 10(3), 251–270.

Fu, M. C. (1994). Optimization via simulation: A review. In P. L. Hammer (Ed.), *Simulation and Modeling*, Volume 53 of *Ann. of Operations Research*, pp. 199–247. Amsterdam, The Netherlands: J. C. Baltzer AG, Science Publishers.

Fu, M. C. (2002). Optimization for simulation: theory vs practice. *INFORMS J. on Comput.* 14(3), 192–215.

Glasserman, P. (1991). *Gradient estimation via perturbation analysis*. The Kluwer Internat. Ser. in Engrg. and Comput. Sci.. Discrete event Dynam. systems. Boston: Kluwer Academic Publishers. by Paul Glasserman ; forward [sic] by Yu-Chi Ho. ill. ; 25 cm.

Ho, Y.-C. and X.-R. Cao (1991). *Perturbation analysis of discrete event dynamic systems*. Boston: Kluwer Academic Publishers.

Ho, Y.-C., R. S. Sreenivas, and P. Vakili (1992). Ordinal optimization of DEDS. *Discrete Event Dynamic Systems: Theory and Applications* 2, 61–81.

Humphrey, D. G. and J. R. Wilson (2000). A revised simplex procedure for stochastic simulation response surface optimization. *INFORMS J. on Comput.* 12(4), 272–283.

- Kolda, T. G., R. M. Lewis, and V. Torczon (2003). Optimization by Direct Search: New perspectives on some classical and modern methods. *SIAM Rev.* 45(3), 385–482.
- Myers, R. H. and D. C. Montgomery (2002). *Response surface methodology: Process and product optimization using designed experiments*. Wiley Ser. in Probab. and Statist. New York: John Wiley and Sons, Inc.
- Nelder, J. A. and R. Mead (1965). A simplex method for function minimization. *Comput. J.* 7, 308–313.
- Pritker, A. A. B., C. E. Sigal, and R. D. J. Hammesfahr (1989). *SLAM II: Network models for decision support*. London: Prentice-Hall International.
- Rubinstein, R. Y. (1992). *Monte Carlo optimization, simulation, and sensitivity of queuing networks* (Reprint ed.). Malabar, Fla.: Krieger Pub. Co. Reuven Y. Rubinstein. 24 cm. Original published: New York : Wiley, 1986.
- Rubinstein, R. Y. and A. Shapiro (1993). *Discrete event systems : sensitivity analysis and stochastic optimization by the score function method*. Wiley Ser. in Probab. and Math. Statist. Chichester England ; New York: Wiley.
- Shi, L. and C.-H. Chen (2000). A new algorithm for stochastic discrete resource allocation optimization. *Discrete Event Dynam. Systems: Theory and Applications* 10(3), 271–294.
- Shi, L. and S. Ólafsson (2000a). Nested Partitions method for global optimization. *Oper. Res.* 48(3), 390–407.
- Shi, L. and S. Ólafsson (2000b). Nested partitions method for stochastic optimization. *Methodology and Comput. in Appl. Probab.* 2(3), 271–291.
- Sphar, C. (1999). *Learn Microsoft Visual C++ 6.0 Now*. Redmond, WA: Microsoft Press. Microsoft Corporation, Microsoft Visual C++ 6.0 deluxe learning edition.
- Walters, F. H., L. R. Parker Jr, S. L. Morgan, and S. N. Deming (1991). *Sequential simplex optimization : a technique for improving quality and productivity in research, development, and manufacturing*. Boca Raton, Fla.: CRC Press.

gresos nacionales e internacionales, siendo la mayoría de ellos, en modelos de simulación. Actualmente está trabajando en optimización mixta para su empleo en optimización de sistemas por modelos de simulación. Es miembro activo de la Unidad de Modelos y Simulación de la Escuela de Ingeniería Eléctrica de la UCV, además de ser miembro del Colegio de Simulación de INFORMS y SIAM.



## BIOGRAFÍA DEL AUTOR

**EBERT BREA** es Profesor Asociado de la Facultad de Ingeniería de la Universidad Central de Venezuela (UCV). Obtuvo su PhD en Matemáticas en la Universidad de Southampton, Inglaterra, su MSc en Investigación de Operaciones y título de Ingeniero Electricista en la UCV. Trabajó como consultor en la entonces Corpoven, S. A. en el desarrollo de modelos de simulación. Ha participado en más de veinte presentaciones en con-