

## **Sistema para Resolver Tres Diseños de la Administración de Combustible**

*José Alejandro Castillo Méndez, Juan José Ortiz Servín,  
José Luis Montes Tadeo, Raúl Perusquia del Cueto*  
*Instituto Nacional de Investigaciones Nucleares*  
*Carretera México-Toluca s/n, La Marquesa, Ocoyoacac, Estado de México*  
*México CP 52750*

[alejandro.castillo@inin.gob.mx](mailto:alejandro.castillo@inin.gob.mx); [juanjose.ortiz@inin.gob.mx](mailto:juanjose.ortiz@inin.gob.mx);  
[jose Luis.montes@inin.gob.mx](mailto:jose Luis.montes@inin.gob.mx); [raul.perusquia@inin.gob.mx](mailto:raul.perusquia@inin.gob.mx)

**Roberto Marinez Rizos**  
*Universidad Autónoma de Campeche*  
*Av. Agustín Melgar s/n entre Calle 20 y Juan de la Barrera, colonia Buenavista,*  
*San Francisco de Campeche, Campeche, México CP 24039*  
[roberto\\_oblivion@hotmail.com](mailto:roberto_oblivion@hotmail.com)

### **Resumen**

En este trabajo se presentan los resultados preliminares, obtenidos con el desarrollo de un sistema de cómputo que resuelve tres de las etapas de la administración de combustible nuclear, las cuales son: los diseños axial y radial de combustible, así como el diseño de recargas de combustible nuclear. La novedad del sistema es que la solución se obtiene resolviendo las 3 etapas mencionadas, en forma acoplada. Para realizar lo anterior, se utilizan técnicas heurísticas para cada una de las etapas, en cada una de estas se tiene una función objetivo que se aplica al problema en particular, pero en todos los casos, los resultados parciales obtenidos se utilizan como datos de entrada para la siguiente etapa. Las técnicas heurísticas que se emplearon para resolver el problema acoplado son la búsqueda tabú, las redes neuronales y un híbrido entre la búsqueda dispersa y el re-encadenamiento de trayectorias. El sistema aplica un proceso iterativo desde el diseño de una celda de combustible, hasta el diseño de la recarga, debido a que se trata de resultados preliminares la recarga se diseña aplicando la estrategia de operación tipo Haling. En cada una de las etapas se vigilan parámetros nucleares inherentes al diseño. Los resultados obtenidos hasta el momento muestran la ventaja de resolver el problema en forma acoplada, aun cuando se utilice una gran cantidad de recursos de cómputo.

## **1. INTRODUCCIÓN**

La administración de combustible nuclear incluye diferentes etapas, cuatro de ellas son los diseños axial y radial de combustible, así como los diseños de la recarga y el respectivo patrón de barras de control. Los diseños mencionados son bastante conocidos y han sido resueltos aplicando diferentes métodos, sin embargo hasta el día de hoy pocos han sido los esfuerzos realizados para resolver el problema en forma acoplada. Algunos ejemplos de lo anterior son los trabajos desarrollados por Castillo et al 2007 y Ortiz et al 2007, quienes resolvieron en forma

acoplada los diseños de recarga y patrón de barras de control. En el mismo sentido Kobayashi 2002 realizó un esfuerzo similar para atacar ambos problemas.

Por lo anterior resulta interesante poder resolver el problema de tres de los diseños en forma acoplada, motivo que incentivó esta investigación. Para realizar esto se utilizaron diferentes técnicas heurísticas en cada una de las etapas, cada una de las cuales utiliza una función objetivo que involucra los parámetros nucleares inherentes a dicha etapa. Las técnicas utilizadas son la búsqueda tabú, las redes neuronales y un híbrido entre la búsqueda dispersa y el re-encadenamiento de trayectorias.

El trabajo está dividido de la siguiente manera: el capítulo 2 incluye la metodología aplicada para la solución del problema planteado; dicho capítulo incluye a su vez la explicación de las técnicas empleadas, así como las funciones objetivo de cada etapa y la implementación para llevar a cabo el proceso iterativo. En el siguiente capítulo se muestran los resultados obtenidos hasta el momento. Finalmente, se presentan las conclusiones obtenidas con la presente investigación.

## 2. METODOLOGÍA

Esta sección se dividirá en tres partes para una mejor comprensión de lo realizado en esta investigación. En la primera parte se explican brevemente las técnicas aplicadas en el sistema, posteriormente se incluyen las funciones objetivo que se utilizaron en cada una de las etapas y en la última parte, se explica el proceso iterativo que realiza el sistema para obtener una solución completa.

Para ubicar el problema en forma adecuada, es importante mencionar que se consideró un ciclo de equilibrio de 18 meses como solución referencia. El ciclo en cuestión utiliza dos lotes de combustible fresco de 52 y 60 ensamblajes respectivamente, los cuales identificaremos como LT1 y LT2 respectivamente. Ambos lotes tienen un porcentaje promedio de  $^{235}\text{U}$  de 3.66 y la diferencia entre ambos es el contenido de gadolinia, el cual es igual a 40% en el primero y de 50% en el segundo. Dicho porcentaje se calcula multiplicando el número de barras con gadolinia, por el porcentaje que tiene la misma, esto es, 8 barras con gadolinia y donde cada una tiene 5% de dicho material es igual a  $8 \times 5 = 40\%$ . La longitud del ciclo es de 10,896 MWD/TU.

### 2.1. Técnicas Heurísticas

A continuación se explica brevemente las técnicas que se implementaron en el sistema en cuestión. Cabe señalar que en caso de que el lector desee profundizar sobre el tema, se incluyen algunas referencias para tal fin.

#### 2.1.1. Re-encadenamiento de trayectorias

Esta técnica surge de un estudio exhaustivo de otra técnica de optimización combinatoria conocida como Búsqueda Dispersa (BD). Por lo tanto, el re-encadenamiento de trayectorias se puede considerar como una extensión de BD, no obstante, debido a sus propias características, en la actualidad se le puede ver como una extensión de otras técnicas conocidas en la optimización combinatoria, tales como Búsqueda Tabú, GRASP, Algoritmos Genéticos, por mencionar

algunas. El objetivo fundamental del Re-encadenamiento de Trayectorias (RT) (Glover, 1999) es la construcción de trayectorias tomando como base un conjunto de soluciones previamente generadas, esto se puede ver como la generación de rutas (o trayectorias) dentro y fuera de dichas soluciones, donde las soluciones resultantes sirven para generar rutas adicionales. Las soluciones que se encuentran en las trayectorias generadas, pueden o no contener atributos de las soluciones iniciales. La incorporación de buenos atributos genera una solución de mejor calidad que las que la originaron.

Como se mencionó anteriormente, el RT se basa en generar rutas entre soluciones generadas previamente a partir de métodos de búsqueda local o de búsqueda global, un aspecto importante de señalar es que las características de dichas trayectorias están relacionadas con los elementos que les sean añadidos, eliminados, o por los movimientos realizados para alcanzar una solución a partir de otra diferente.

Para trazar las trayectorias deseadas, los movimientos que se realizan deben considerar la siguiente regla: se tiene como origen a una solución denotada como solución inicial y una segunda llamada solución guía, los movimientos deben introducir progresivamente atributos tanto de la solución inicial, como la de la solución guía, con la idea de reducir la distancia entre los atributos de ambas. La calidad de dichos atributos se evalúa con la función objetivo para determinar qué movimientos se deben efectuar.

Durante el proceso iterativo, puede ocurrir que las nuevas rutas generadas no mejoren la solución, sin embargo, pueden tomarse como origen para generar nuevas trayectorias que mejoren la calidad de la misma.

Los elementos principales del RT son los siguientes:

- a) Generar un conjunto de soluciones previas con algún método de búsqueda global o local.
- b) Trazar trayectorias entre dos soluciones de dicho conjunto de soluciones.
- c) Actualización del conjunto de soluciones, en caso de que alguna trayectoria generada sea mejor que alguna solución considerada.

Los pasos anteriores se repiten hasta que el número máximo de iteraciones se haya alcanzado, 300 en este caso, o hasta que la función objetivo no mejore después de un número determinado de iteraciones, 50 en este caso. Entendiéndose la mejora de la función objetivo con respecto a su evaluación.

### **2.1.2. Búsqueda dispersa**

La Búsqueda Dispersa (BD) (Glover, 1998) es una técnica de optimización combinatoria adecuada para encontrar buenas soluciones, lo cual logra considerando la estructura del problema y realizando una búsqueda en forma inteligente. El punto nodal de esta técnica es construir combinaciones lineales entre dos soluciones para obtener una nueva, la cual por sus orígenes se puede considerar de mayor calidad. La BD se fundamenta en la elección sistemática y estratégica sobre un conjunto pequeño de soluciones, con una longitud de entre 10 y 20.

La BD consta, en términos generales, de los siguientes elementos:

- a) Un generador de soluciones diversas: Está compuesto por alrededor de 100 soluciones, que fueron originadas a partir de una solución semilla, ésta puede ser obtenida mediante métodos de búsqueda local o búsqueda global. De aquí se extraen un subconjunto pequeño (de entre 10 y 20 soluciones) para realizar las combinaciones lineales.
- b) Conjunto referencia 1. Está compuesto por las mejores soluciones del conjunto disperso, su cardinalidad será la mitad del tamaño del conjunto referencia, tomando como criterio la calidad de la función objetivo.
- c) Conjunto referencia 2. Se obtiene con base en las máximas distancias entre los elementos del conjunto disperso y el conjunto referencia 1. Es necesario definir una métrica para esto.
- d) Un método de combinación. Se encargará de realizar las respectivas combinaciones lineales entre los conjuntos referencia 1 y 2, esto se hace considerando parejas de elementos.
- e) Actualización. El conjunto referencia (que será la unión de los conjuntos referencia 1 y 2) se puede o no actualizar con las soluciones resultantes de la combinación. Para que una solución pueda incorporarse al conjunto referencia ésta debe mejorar a las soluciones que se encuentran dentro del conjunto, por lo que a lo largo de la búsqueda el conjunto referencia mantiene un tamaño constante, pero que va mejorando conforme avanza el proceso iterativo. Hay dos tipos de actualización, la dinámica y la estática, el usuario debe seleccionar cuál usará de acuerdo a las necesidades del problema.

### 2.1.3. Búsqueda tabú

La Búsqueda Tabú (BT) (Glover, 1968) parte de dos conceptos básicos, vecindad y movimiento prohibido o como se llamará a partir de este momento, movimiento tabú. En el primer caso se parte del hecho de que se puede construir una vecindad  $N(x)$  de puntos. Cada uno de los puntos me conducirá a una solución factible, en general, diferente. Entonces,  $N(x)$  es el conjunto de todas las soluciones factibles que se pueden conseguir a partir de  $x$ , donde  $x$  es cualquier punto del dominio de la función que se desea maximizar. Por otro lado, definiremos el estatus tabú como la prohibición de emplear un punto determinado del dominio de la función, para encontrar una solución factible, dicho estatus se asignará durante un número determinado de iteraciones, es importante mencionar que dicho estatus permanece aun cuando la solución factible obtenida a partir de dicho punto sea mejor que otras ya encontradas. La construcción de la vecindad dependerá del problema en particular que se resuelva, no obstante, ésta se sugiere que se genere en forma aleatoria. En cada paso del proceso iterativo nos movemos, aleatoriamente, tratando de encontrar la mejor solución  $x^*$  en  $N(x)$ , independientemente de que  $F(x^*)$  sea mejor o no que  $F(x)$ , donde  $F$  es la función que se desea maximizar. Si la vecindad  $N(x)$  es muy grande o la evaluación de la función objetivo es muy cara se puede analizar únicamente un subconjunto de ella, por ejemplo  $V(x)$ , donde el tamaño de dicha subvecindad dependerá del problema en particular. El proceso se va desarrollando a partir de movimientos simples entre puntos de una vecindad, obteniendo con ello diferentes soluciones factibles.

Con lo anterior la BT se podría ver como un método tradicional de optimización de paso descendente, sin embargo, la diferencia entre uno y otro lo hace la definición de estatus tabú o movimiento prohibido como en algunos casos se le llama, lo cual puede conducir a una peor solución, entre un movimiento y otro, esto último permite salirse de óptimos locales. Los

movimientos prohibidos se manejan por medio de un arreglo  $T$  llamado lista tabú de longitud fija  $|T| = t$ ,  $t$  entero. El arreglo contiene, al menos las dos entradas siguientes asociadas a cada elemento de la lista, primero: el movimiento realizado o alguna forma de identificarlo, en segundo lugar, el valor entero  $t$  para identificar el estatus tabú de dicho movimiento. El estatus tabú se aplica a movimientos realizados recientemente y dicho estatus permanece por un número determinado de iteraciones que se fija previamente. La literatura sobre el tema sugiere, según la experiencia, que el valor de la lista tabú sea entre 4 y 25, no obstante también se sugiere realizar algún análisis estadístico para fijar dicho valor.

#### **2.1.4. Redes neuronales**

Para la presente investigación se utilizó una Red Neuronal (RN) de tipo Hopfield multivaluada (Mérida, et al. 2001) la cual contiene  $N$  neuronas conectadas entre sí. Cada neurona tiene asociado un estado neuronal, que puede tomar valores en el rango de 1 a  $M$ . Cuando todas las neuronas tienen un estado neuronal asociado, se habla del estado global de la red. Es importante decir que a diferencia de muchas redes neuronales, este tipo de red no se entrena, sino que los pesos se predeterminan, para llegar a la solución buscada.

En este tipo de red, para el funcionamiento adecuado de la misma, es necesario contar con una función de energía y una regla de transición de estados. En este caso, la función de energía indica el nivel de energía del estado global de la red, mientras que la regla de transición de estados le permite a las neuronas cambiar sus estados neuronales, de modo que el nivel de energía de la red neuronal disminuya. Por cada cambio de un estado neuronal, se evalúa la función de energía. Con lo anterior se busca que el nivel de energía sea mínimo o máximo, dependiendo del objetivo. Los pesos entre las neuronas suelen involucrarse en la función de energía.

En el caso particular de diseño de celdas de combustible, la función de energía de la red se toma como la función objetivo, la cual será mostrada en la siguiente sección. Los estados neuronales son números enteros que denotan combinaciones de uranio y gadolinia para formar las barras que son utilizadas en el diseño de la celda. Las neuronas de la red se corresponden con las posiciones de la celda de combustible.

En el caso del diseño de la recarga de combustible, la función de energía también se muestra en la siguiente sección. Los estados neuronales son números enteros que denotan los ensambles combustibles ordenados de acuerdo a su quemado. Los primeros números indican ensambles frescos y números grande indican ensambles muy quemados. Las neuronas de la red se corresponden con los canales de un cuarto del núcleo del reactor. Esto quiere decir que si la neurona  $j$  tiene el estado neuronal  $k$ , significa que en el canal  $j$  se debe insertar el ensamble  $k$  de la lista de ensambles ordenados por quemado.

#### **2.2. Funciones Objetivo**

Para llegar a una solución completa contemplando los tres diseños ya mencionados, es necesario utilizar funciones objetivo para cada etapa del proceso, debido a que los parámetros que se involucran dependen de cada diseño a realizar. Las funciones objetivo que se emplearon son las siguientes:

### 2.2.1. Diseño radial de combustible

La etapa de diseño radial de combustible se ha resuelto hasta el momento con dos técnicas heurísticas: las redes neuronales y el híbrido entre la búsqueda dispersa y el re-encadenamiento de trayectorias, ambas técnicas emplearon la Ecuación 1 como función objetivo.

$$\min G = FPPL \cdot w_1 + K(k_\infty) \quad (1)$$

donde

$$K(k_\infty) = \begin{cases} |k_\infty| \cdot w_2 & \text{si } |k_\infty - 1.18| > 0.05 \\ 0 & \text{en otro caso} \end{cases}$$

donde

FPPL : factor de pico local de potencia de la celda.

$k_\infty$ : el factor de multiplicación infinita de neutrones de la celda.

Los parámetros  $w_1$  y  $w_2$  son factores de peso, los cuales se obtuvieron mediante un análisis estadístico. De acuerdo a la Ecuación 1, es claro que cuando la  $k_\infty$  cumple la restricción impuesta, la función objetivo solamente minimiza sobre el parámetro FPPL. En el mismo sentido, para evaluar las diferentes configuraciones durante el proceso de optimización se utilizó el código en 2D CASMO-4 (Rhodes and Edenius, 2004).

### 2.2.2. Diseño de recargas

Como se mencionó al inicio, para el diseño de la recarga se aplicó una estrategia de carga tipo Haling, con la idea de facilitar los cálculos en esta etapa preliminar del desarrollo del sistema, la función objetivo es la siguiente:

$$\max H_1 = k_{eff} \cdot w_1 + \Delta Lim_1 \cdot w_2 + \Delta Lim_2 \cdot w_3 + \Delta Lim_3 \cdot w_4 + \Delta Lim_4 \cdot w_5 + \Delta Lim_5 \cdot w_6 \quad (2)$$

donde

$k_{eff}$ : el factor de multiplicación efectivo de neutrones

$\Delta Lim_1 = MFLPD_{lim} - MFLPD_{obtenido}$  (límite térmico)

$\Delta Lim_2 = MAPRAT_{lim} - MAPRAT_{obtenido}$  (límite térmico)

$\Delta Lim_4 = MFLCPR_{lim} - MFLCPR_{obtenido}$  (límite térmico)

$\Delta Lim_5 = SDM_{obtenido} - SDM_{lim}$  (margen de apagado en frío)

Al igual que en el caso anterior,  $w_i$ ,  $i=1, \dots, 5$  son factores de peso obtenidos estadísticamente. Los primeros cuatro parámetros se calculan al final del ciclo de acuerdo al Principio Haling. El SDM se calcula al inicio del ciclo. En este caso, la obtención de los parámetros involucrados en la evaluación de la función objetivo se realizó empleando el código en 3D SIMULATE-3 (Dean 2005).

### 2.3. Proceso Iterativo

El proceso iterativo inicia con el diseño de una celda de combustible y siempre concluye con el diseño de la recarga de combustible. En la Figura 1 se muestra un diagrama de flujo del proceso en cuestión.

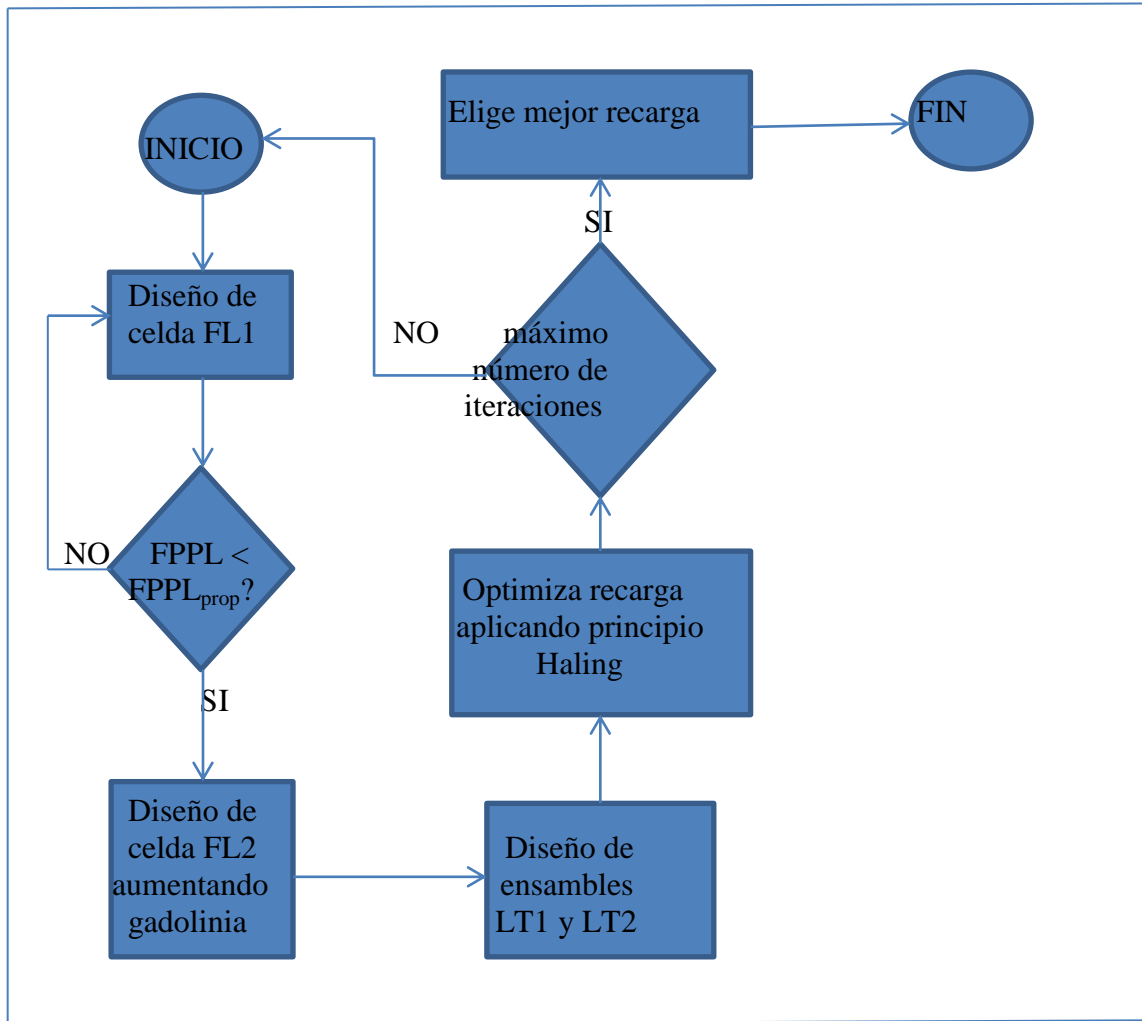


Figura 1. Diagrama de flujo del proceso iterativo.

De la Figura 3 es importante hacer los siguientes comentarios:

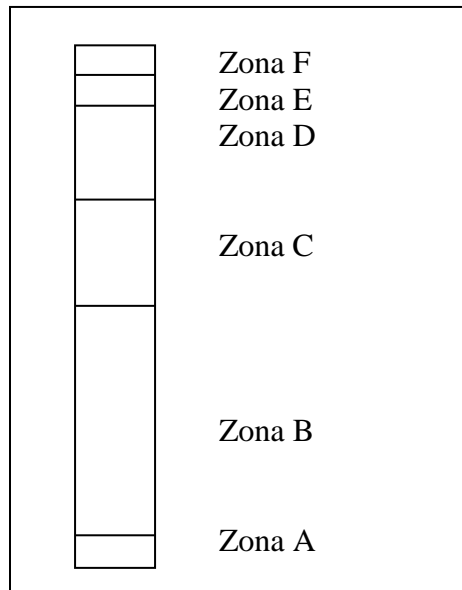
- 1) El valor para el  $FPPL_{prop}$  es un poco mayor que el valor de referencia, en este caso 1.201.
- 2) Para el diseño de la celda FL2 se aumenta el 10% de gadolinia y se conserva la misma configuración. Las posiciones en las que se coloca dicha gadolinia se elige al azar.
- 3) Los dos lotes de ensamblajes se diseñan extrapolando las celdas de la parte baja, previamente obtenidas.

El proceso iterativo concluye después de 10 iteraciones, o si la diferencia entre el factor de multiplicación efectiva de neutrones al final de operación a plena potencia de dos iteraciones continuas, es menor a una tolerancia propuesta.

### 3. RESULTADOS

Hasta el momento el sistema se ha ejecutado pocas veces, debido sobre todo a la gran cantidad de recursos computacionales requeridos. Por lo anterior, en el presente trabajo solamente se muestran los resultados obtenidos con las dos mejores ejecuciones.

Si bien el primer diseño fue el de una celda, empecemos por mostrar la estructura empleada para el diseño axial del combustible, para entender de mejor manera que parte de este se diseñó con las técnicas mencionadas en el párrafo anterior. La estructura empleada para ambos ensambles frescos es la mostrada en la Figura 2, en este caso, las zonas A, E y F corresponden a celdas de uranio natural, mientras que las zonas B, C y D tienen la misma configuración en cuanto a uranio enriquecido se refiere, pero diferente contenido de gadolinia.



**Figura 2. Zonas axiales para el ensamble combustible.**

En la Tabla I se muestran los resultados para las 6 celdas diseñadas, 3 para cada lote de ensambles frescos, que incluyen las zonas B, C y D. La tabla incluye los valores para el FPPL y la  $k_{\infty}$  respectivamente. De igual manera, la tabla incluye los valores de referencia que se tomaron en cuenta.

Como ya se comentó, el siguiente paso consiste en construir los ensambles de combustible fresco, extrapolando las celdas de la parte baja y añadiendo gadolinia en las zonas C y D. Una vez que se diseñaron los ensambles se procede a diseñar la recarga de combustible empleando el principio Haling, posteriormente se realiza el proceso iterativo de acuerdo al diagrama de flujo mostrado en la Figura I. En la Tabla II se muestran los resultados obtenidos.



**Tabla I. Resultados de las celdas de combustible.**

	Zona B		Zona C		Zona D	
	$k_{\infty}$	LPPF	$k_{\infty}$	LPPF	$k_{\infty}$	LPPF
FL1 referencia	<b>1.148</b>	<b>1.201</b>	<b>1.121</b>	<b>1.222</b>	<b>1.079</b>	<b>1.205</b>
FL 2 referencia	<b>1.099</b>	<b>1.254</b>	<b>1.058</b>	<b>1.305</b>	<b>1.004</b>	<b>1.307</b>
FL1 ejemplo 1	1.144	1.301	1.098	1.362	1.057	1.362
FL2 ejemplo 1	1.133	1.311	1.096	1.349	1.060	1.370
FL1 ejemplo 2	1.146	1.341	1.149	1.337	1.117	1.341
FL2 ejemplo 2	1.115	1.352	1.073	1.411	1.029	1.430

Para concluir, se muestra los resultados obtenidos para la recarga de combustible obtenidas al final del proceso iterativo (Tabla II). Cabe señalar que en todas las ejecuciones realizadas, la mejor recarga se obtuvo en la última iteración.

**Tabla II. Resultados de la recarga.**

	Principio Haling				
	$k_{eff}$ (EOC)	MFLPD	MAPRAT	MFLCPR	SDM
Recarga referencia	<b>0.9978</b>	<b>0.689</b>	<b>0.706</b>	<b>0.772</b>	<b>1.676</b>
Recarga ejemplo 1	0.9981	0.641	0.707	0.718	2.925
Recarga ejemplo 2	0.9873	0.693	0.796	0.786	1.456

Resulta claro que el proceso completo requiere de muchos recursos computacionales, para evaluar las funciones objetivo involucradas en el proceso. Por tal motivo, consideramos importante mostrar el número de veces que cada una de las funciones se evaluó. En la Tabla III se muestran dichos valores, cabe mencionar que una ejecución completa fue de 3 semanas (tiempo real) aproximadamente, usando una estación de trabajo Dell T5660 con plataforma LINUX. Por otro lado, es necesario resaltar que cada evaluación de alguna de las funciones objetivo empleadas requiere de al menos 3 ejecuciones ya sea de CASMO o de SIMULATE según sea el caso, lo cual incrementa sobre manera el costo de cada una de estas (Ecuaciones 1 y 2).

**Tabla III. Evaluaciones de las funciones objetivo.**

	Diseño de celda	Diseño de la recarga (Haling)
Ejemplo 1	10000	7514
Ejemplo 2	10000	8274

#### 4. CONCLUSIONES

Se presentó un sistema que resuelve tres de las etapas involucradas en la administración de combustible nuclear, estas son, los diseños radial y axial de combustible, así como el diseño de la recarga de combustible nuclear. Si bien es cierto que los recursos empleados en un proceso completo son bastantes, se tiene la ventaja de que al concluir, se tienen los diseños completos, satisfaciendo todas las restricciones impuestas; de igual manera es importante destacar la retroalimentación que existe entre las diferentes etapas de diseño.

Por otro lado, se puede observar que aun cuando aparentemente los resultados en el diseño de celdas no superan a los de referencia (Tabla I), al final, la recarga satisface los límites térmicos y en uno de los ejemplos supera los valores de referencia (Tabla II). Lo que indica que es posible tener un margen de maniobra más amplio en la primera etapa del proceso.

Como trabajo futuro, es interesante realizar la implementación del sistema añadiendo el patrón de barras de control, para tener completo el proceso.

Por otro lado, consideramos que es útil afinar algunas de las etapas del proceso, por ejemplo realizar algún tipo de optimización al momento de generar la celda FL2, con la idea de obtener mejores resultados.

#### AGRADECIMIENTOS

Los autores agradecen el soporte proporcionado por el Instituto Nacional de Investigaciones Nucleares a través del proyecto CA-215, de igual manera agradecen al Consejo Nacional de Ciencia y Tecnología por los recursos brindados a través del proyecto CB-2011-01-168722 para la realización del presente trabajo; finalmente un profundo agradecimiento al Departamento de Gestión de Combustible de la Comisión Federal de Electricidad de México, por las facilidades brindadas.

#### REFERENCIAS

1. Alejandro Castillo, Juan José Ortiz, José Luis Montes, Raul Perusquía, “Fuel Loading and Control Rod Patterns Optimization in a BWR Using Tabu Search”, *Annals of Nuclear Energy*, vol. 34, p. 207-212 (2007).
2. Castillo J. A., Alonso G., Morales L. B., del Valle E., “BWR fuel reloads design using a Tabu search technique”, *Annals of Nuclear Energy*, vol. 31, p. 151-161 (2004).
3. Dean D. W. 2005. SIMULATE-3. Advanced Three-Dimensional Two-Group Reactor Analysis Code. User’s Manual. Studsvik Scandpower INC. SSP-95/15 - Rev 3.
4. Glover F., “Tabu Search Part I”, *ORSA, Journal of Computing*, vol. 1, p. 190-206 (1968).

5. Glover F., “Scatter Search and Star Paths: Beyond the Genetic Metaphor”, *OR Spektrum*, **vol. 17**, p. 125-137 (1995).
6. Glover F., “A template for Scatter Search and Path Relinking”, *Artificial Evolution, Lecture Notes in Computer Science*, **vol. 1363**, J.K. Hao, E. Lutton, E. Ronald, M. Schoenauer, D. Snyers (Eds.), p. 13-54 (1998).
7. Glover, F. “Scatter Search and Path Relinking”. In: D. Corne, M. Dorigo and F. Glover (Eds), *New Ideas in Optimisation*, Wiley (1999).
8. Juan José Ortiz, Alejandro Castillo, José Luis Montes, Raúl Perusquía, “A new system to fuel loading and control rod pattern optimization in Boiling Water Reactors”, *Nuclear Science and Engineering*, **vol. 157**, p. 236-244 (2007).
9. Mérida CE, Galán M, Muñoz-Pérez, “An Efficient Multivalued Hopfield Network for the TSP”, *Neural Processing Letters*, **vol. 14**, p. 203-216 (2001).
10. Ortiz JJ, Requena I, “Using a Multi-State Recurrent Neural Network to Optimize Fuel Reloads in BWRs”, *Annals of Nuclear Energy*, **vol. 31**, p. 789-803 (2004).
11. Rhodes, J., & Edenius, M., 2004. CASMO-4. A Fuel Assembly Burnup Program. User’s Manual. SSP-01/400 Rev 4. Studsvik Scandpower.
12. Y. Kobayashi, E. Aiyoshi, “Optimization of Boiling Water Reactor Loading Pattern Using Two-Stage Genetic Algorithm”, *Nuclear Science and Engineering*, **vol. 142**, p. 119-139 (2002).