



MX0500036

Congreso Internacional Conjunto Cancún 2004 LAS/ANS-SNM-SMSR/International Joint Meeting Cancun 2004 LAS/ANS-SNM-SMSR
XV Congreso Anual de la SNM y XXII Reunión Anual de la SMSR/XV SNM Annual Meeting and XXII SMSR Annual Meeting
Cancún, Q.R., México, 11-14 de Julio, 2004/Cancún, Q.R., Mexico, July 11-14, 2004

BUTREN-RC un Sistema Híbrido para la Optimización de Recargas de Combustible Nuclear en un BWR

Juan José Ortiz Servin, José Alejandro Castillo Méndez¹

Instituto Nacional de Investigaciones Nucleares

Km. 36.5 Carretera México-Toluca, Ocoyoacac, Estado de México 52245, México

jjortiz@nuclear.inin.mx; jacm@nuclear.inin.mx

Edmundo del Valle Gallegos

Instituto Politécnico Nacional

Unidad Profesional "Adolfo López Mateos", Col. Lindavista 07738, D. F., México

edmundo@nuclear.esfm.ipn.mx

Resumen

Se presentan los resultados obtenidos con el sistema híbrido BUTREN-RC que obtiene recargas de combustible nuclear para un reactor tipo BWR. El sistema tiene implementados los métodos de optimización heurística búsqueda tabú y redes neuronales. La optimización se llevó a cabo con la técnica de búsqueda tabú, y las redes neuronales, previamente entrenadas, se utilizaron para predecir el comportamiento de las recargas de combustible, en sustitución de códigos comerciales de simulación del reactor. Las recargas de combustible nuclear obtenidas corresponden a 5 diferentes ciclos de operación de la Central Nucleoeléctrica de Laguna Verde, Veracruz en México. Los resultados obtenidos se compararon con los diseños de dichos ciclos. La ganancia de energía con las recargas de combustible propuestas es de aproximadamente un 4.5% con respecto a las de diseño. El tiempo de cómputo consumido fue considerablemente menor que cuando se utiliza un código comercial de simulación para reactores.

1. INTRODUCCIÓN

Cuando se utiliza un reactor nuclear para generar energía eléctrica, es importante analizar dos aspectos involucrados en su funcionamiento. En primer lugar es necesario obtener la máxima energía posible, lo cual cubre el aspecto económico; en segundo lugar es necesario que la operación se realice bajo condiciones de seguridad. En este sentido, la obtención de un patrón de recargas de combustible nuclear óptimo juega un papel sumamente importante.

En una buena parte de las plantas nucleares a nivel mundial, el diseño de patrones de recarga de combustible nuclear (PRCN) se realiza con base en la experiencia de los ingenieros dedicados a dichas tareas. En la actualidad, y gracias a las diversas investigaciones en el área de optimización, se han implementado diversos métodos con el objetivo de maximizar la energía obtenida satisfaciendo los límites térmicos de operación.

¹ Estudiante de Doctorado de la Universidad Autónoma del Estado de México

Algunos de los métodos utilizados para resolver el problema del diseño de PRCN son los siguientes: algoritmos genéticos [1], sistemas expertos [2] y búsqueda tabú [3], entre otros.

Algo que tienen en común muchos de los sistemas de optimización propuestos en la literatura especializada, es que utilizan algún código para simular el funcionamiento del reactor o resuelven la ecuación de difusión de neutrones, para determinar si la recarga de combustible es aceptable o no. Esto tiene como desventaja, que si es necesario evaluar miles de recargas, los recursos de cómputo consumidos sean mayores.

Los sistemas RENOR [4] y RECOPIA [5] son híbridos que combinan métodos de búsqueda con redes neuronales entrenadas específicamente para predecir si una recarga de combustible es aceptable o no. RENOR utiliza una red neuronal de tipo Hopfield y RECOPIA utiliza un algoritmo genético como método de optimización. Estos dos sistemas dieron resultados bastante interesantes en tiempos muy cortos. Una búsqueda típica puede tomar alrededor de 2 minutos en una computadora personal en un PC a 350 MHz. También es interesante resaltar que RENOR requiere de menos iteraciones que RECOPIA. En promedio RENOR realizó 1000 evaluaciones de recargas de combustibles, mientras que RECOPIA realizó 1400 evaluaciones.

En este trabajo se presenta una combinación de los métodos de búsqueda tabú, que se encarga de la optimización de la recarga de combustible; y de redes neuronales para predecir su comportamiento.

En la siguiente sección se explica en detalle el problema que se quiere resolver. En la Sección 3 se describen las técnicas de Redes Neuronales y de Búsqueda Tabú. En la Sección 4 se describe la implementación del sistema híbrido. Posteriormente se analiza su desempeño para optimizar recargas de combustible en la Sección 5. Finalmente, se mencionan las conclusiones y el trabajo futuro.

2. DESCRIPCIÓN DEL PROBLEMA

La Figura 1 muestra el corte transversal del núcleo completo de los reactores de la Central Nucleoeléctrica de Laguna Verde (CNLV) que cuentan con 444 canales representados por las celdas. En este caso, se tiene una distribución de ensambles de combustible nuclear, de diferentes tipos y con diferentes características, si bien algunos pueden ser iguales, en general se tienen alrededor de 5 tipos de ensambles diferentes. La optimización de la recarga de combustible consiste en colocar 444 ensambles en los canales del reactor de modo que se tenga una operación segura y eficiente. Para resolver este problema de optimización combinatoria se hacen intercambios entre las posiciones de los ensambles de combustible hasta alcanzar los objetivos y satisfacer las restricciones.

En este trabajo se utiliza un cálculo Haling para el diseño de las recargas, por lo cual sólo se está garantizando la seguridad del reactor en dos puntos del ciclo, el inicio y el final del mismo. Sin embargo debe quedar claro, que la seguridad del núcleo debe verificarse a lo largo del ciclo y eso se hace mediante el diseño de patrones de barras de control. Por lo tanto, los criterios que se están revisando son los siguientes:

- el Factor de Pico de Potencia Radial (FPPR) al inicio del ciclo no sobrepase el límite máximo permisible ($FPPR_{Max}$),
- el Exceso de Reactividad en Caliente al inicio del ciclo (ERC) no supere el umbral máximo permitido (ERC_{Max}),
- la Mínima Razón de Potencia Crítica (MCPR) al final del ciclo sea superior al mínimo valor permitido ($MCPR_{Min}$) y
- la Máxima Tasa Lineal de Generación de Calor (MLHGR) al final del ciclo sea inferior al máximo valor permitido ($MLHGR_{Max}$);

Una vez cubiertos estos aspectos de seguridad, se busca aquella configuración que logra extraer la mayor cantidad de energía del combustible.

El tamaño del espacio muestral de soluciones al problema es muy grande. Por tal motivo se consideran algunas simplificaciones y reglas que lo reducen de forma significativa. Así, cuando la simetría de $1/4$ del núcleo, con respecto a los ejes vertical y horizontal (como se muestra en la Figura 1) es tomada en cuenta, el problema planteado se reduce a 111 ensamblajes. Además, cuando se considera la simetría de $1/8$ del núcleo (las celdas en negro de la Figura 1 dividen $1/4$ del núcleo en dos mitades), la cantidad de canales se reduce a 60. Esto quiere decir que en principio hay $60!$ (8.32098×10^{81}) posibles recargas de combustible cuando en los 60 canales se deben acomodar 60 combustibles diferentes.

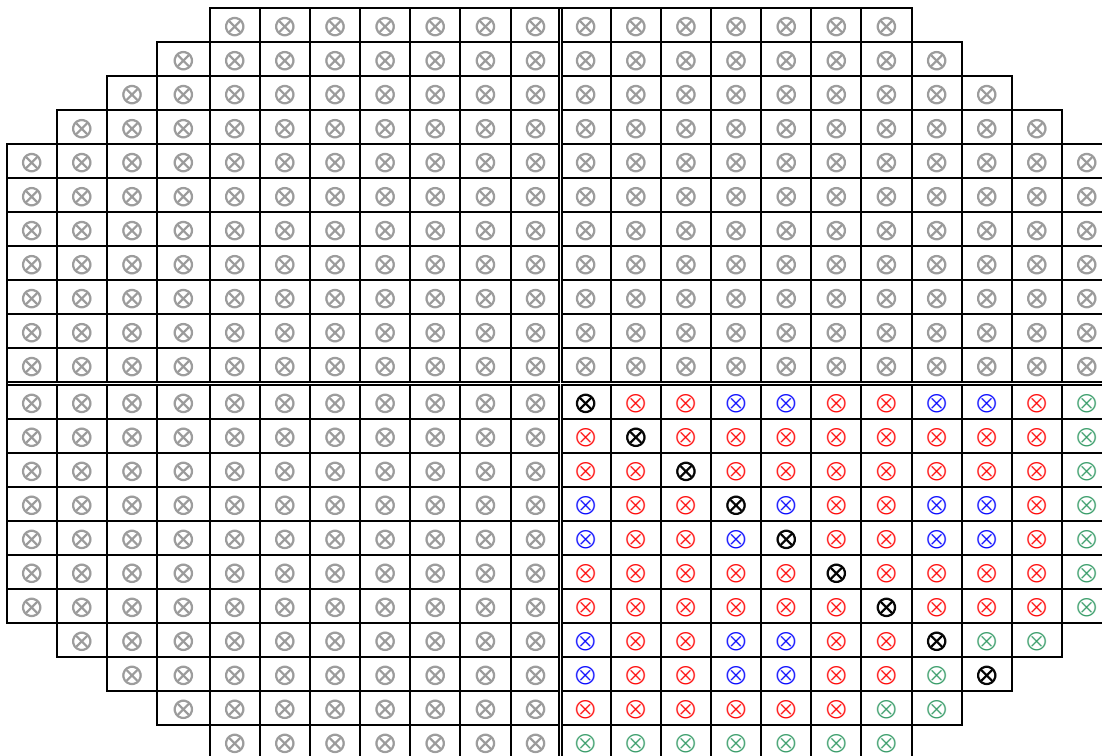


Figura 1. Esquema simplificado del núcleo constituido por 444 ensamblajes.

Las siguientes reglas, algunas de ellas de tipo heurístico, reducen aún más el número de posibilidades; en la Figura 1 se resalta el cuadrante inferior derecho del núcleo indicando dichas reglas:

- 1) En la periferia sólo se pueden colocar ensambles muy quemados (color verde).
- 2) Los ensambles que se encuentran en la diagonal sólo se pueden intercambiar entre ellos, siempre y cuando no infrinjan alguna otra regla (color negro).
- 3) La estrategia de carga Control Cell Core (CCC) [6] establece que en torno a las barras de control de la Secuencia A2 (color azul) no se pueden colocar ensambles nuevos, ni tampoco ensambles que en ciclos anteriores hubiesen estado en estas posiciones.

Las posiciones restantes (color rojo) no tienen restricción alguna. La regla 1 ayuda a reducir la fluencia de neutrones en la vasija, mientras que la regla 2 evita romper la simetría en ambos octavos del núcleo. Finalmente la regla 3, facilita la operación del reactor al disminuir el número de barras de control que se utilizan. Con estas reglas, el problema se reduce a 7.361×10^{54} posibilidades de intercambio entre los ensambles, lo cual es todavía considerable.

3. METODOLOGÍA

Como ya se ha mencionado los métodos empleados en el sistema BUTREN-RC, son la búsqueda tabú y redes neuronales. A continuación se explica cada uno de ellos.

3.1 Redes Neuronales

Las redes neuronales artificiales [7-9] están inspiradas en el modelo biológico del cerebro de un ser vivo. Se componen de neuronas o elementos de procesamiento que siguen un esquema de interconexión altamente paralelo. Cada elemento de procesamiento está formado por un cuerpo que procesa información, varios canales o vías de entrada de estímulos, y por una vía de salida de la respuesta de la neurona a los estímulos que recibe.

Los elementos de procesamiento están conectados entre sí de forma ponderada, por medio de pesos denotados por w_{ij} . La adaptación de esos factores de ponderación entre las distintas neuronas para responder de manera adecuada ante un estímulo inicial recibe el nombre de aprendizaje.

El aprendizaje de una red neuronal usualmente se lleva a cabo en un proceso iterativo en el que ésta analiza sucesivamente un conjunto de muestras de entrenamiento de modo que los pesos de las conexiones entre las distintas neuronas se van ajustando para desempeñar una tarea concreta.

Existen varios esquemas para el ajuste de pesos, pero para los motivos que se persiguen en este trabajo, se explicará con cierto detalle el modelo de retropropagación de errores. Por otro lado, también existen muchos tipos de redes neuronales y el que nos interesa es el Perceptron Multicapa. Este tipo de red está basado en el modelo propuesto por Rosenblatt en 1958 [7, 8]. A ese modelo primitivo se le incorporaron varias modificaciones como la introducción de más de una capa de neuronas, una nueva regla de aprendizaje y métodos para el aceleramiento del

entrenamiento.

El Perceptron multicapa consiste en una capa de neuronas que distribuye la información de entrada, a un conjunto de capas secuenciales de neuronas. A la última de estas capas se le llama capa de salida. El cuerpo de la neurona procesa la información calculando primero el estímulo neto o excitación que recibe de la siguiente manera:

$$I_{neta}^j = \sum_{i=1}^N w_{ij} x_i + \theta_j \quad (1)$$

donde N es el número de neuronas conectadas a la j -ésima neurona, x_i es el estímulo que fluye de la neurona i a la neurona j y w_{ij} es el peso de dicha conexión. θ_j es un término de umbral de disparo de la neurona. En función del valor de I_{neta}^j , la neurona dispara una respuesta excitatoria o inhibitoria dada por $y^j = F_j(I_{neta}^j)$, donde F_j es una función de activación que puede tomar diferentes formas, la más común es la sigmoide y tiene la siguiente forma:

$$f(I_i) = \frac{1}{1 + e^{-\lambda I_i}} \quad (2)$$

donde λ es un factor de ajuste. Con esto, de las Ecs. (1) y (2), la salida y_j de la neurona j está dada entonces por la expresión:

$$y_j = f_j(I_j^{neta}) = f_j\left(\sum_i^n w_{ij} x_j + \theta_i\right) \quad (3)$$

Este modelo de red neuronal se entrena siguiendo un esquema supervisado en el que la red neuronal aprende a relacionar un conjunto de vectores de entrada con otro conjunto de vectores de salida.

Así, es posible calcular un error E entre los valores predichos por la red neuronal y la respuesta esperada. Durante el entrenamiento, para un conjunto de pares de datos (X, T) , se busca que el error entre la respuesta de la red y la respuesta deseada de todo el conjunto de pares sea mínimo:

$$\min E = \frac{1}{2} \sum_{k=1}^M \sum_{i=1}^N (T_i - y_j) \quad (4)$$

donde M es el número de ejemplos de entrenamiento, y_i es la salida de la neurona i y N la cantidad neuronas en la capa de salida. La Regla Delta Generalizada [7, 8] indica la forma en que cambian los pesos en la capa de salida para reducir el error en la Ecuación (4) y se expresa como:

$$w_{kj}^o(t+1) = w_{kj}^o(t) + \alpha \delta_{pk}^o f^h(I_{pj}^h) \quad (5)$$

mientras que para las capas ocultas se tiene:

$$w_{ji}^h(t+1) = w_{ji}^h(t) + \alpha \delta_{pj}^h x_i \quad (6)$$

donde

$$\delta_{pk}^o = (T_{pk} - y_{pk}) f'_k(I_{pk}^o) \quad (7)$$

$$\delta_{pj}^h = f'_j(I_{pj}^h) \sum_k \delta_{pk}^o w_{kj}^o \quad (8)$$

y

- w_{ji}^h : es el peso de la conexión entre las neuronas i y j de las capas de entrada y oculta
- w_{kj}^o : es el peso de conexión entre la neurona de la capa oculta y la neurona de la capa de salida
- T_{pk} : es la salida deseada para la neurona k y el ejemplo de entrenamiento p
- y_{pk} : es la salida de la neurona k para el ejemplo p
- f'_k y f'_j : son las funciones de activación para las capas ocultas y de salida
- I_{pk}^o e I_{pj}^h : son las entradas netas a las neuronas k y j respectivamente
- x_i : es el estímulo que entra en la neurona i de la capa de entrada
- α : es la tasa de aprendizaje de la red que toma valores en $[0,1]$

Es habitual agregar un término de momentum en el ajuste de pesos, que permita acelerar el entrenamiento. Éste está expresado por:

$$Momentum = \beta \Delta w_{ij}(t-1) \quad (9)$$

donde β es el término de momentum y toma valores en el intervalo $[0,1]$. Se ha visto que se obtienen buenos resultados si $\alpha < \beta$ [7].

En caso de que existan más capas ocultas, la adaptación de los pesos es similar a lo indicado en las Ecs. (6) y (8), pero considerando las δ_{ij} correspondientes a la capa siguiente (anterior en la retropropagación) de la red.

Cuando se introduce una señal en la capa de entrada de la red, ésta se propaga hacia las capas siguientes. La respuesta de la red se compara con el valor deseado y se produce una señal de error que se retropropaga hacia las capas precedentes, posteriormente, los pesos se cambian según la Regla Delta Generalizada. Con los nuevos pesos se propaga la señal de entrada nuevamente hacia la capa de salida y se calcula una nueva señal de error. Este proceso se repite hasta que el error se hace mínimo o alcanza un valor especificado previamente.

3.2 Búsqueda Tabú

La búsqueda tabú [10-12] es un proceso iterativo que inicia con una solución factible para obtener un óptimo global (máximo o mínimo) de una función objetivo F . Esto se logra por medio de movimientos que generen nuevas soluciones. Se basa en los conceptos de vecindad y de movimientos o intercambios clasificados como prohibidos o tabú. Las características anteriores le permiten al método salirse de regiones donde existan óptimos locales, a diferencia de los métodos

clásicos de optimización, que dependen del punto inicial y en general sólo se garantiza la obtención de un óptimo local.

En general, la búsqueda tabú parte del hecho de que se puede construir una vecindad durante el proceso iterativo de búsqueda. Basados en esto, se puede definir un conjunto M de modificaciones que se aplican a una solución dada x para alcanzar otra solución x^* . Por lo tanto, la notación $x^*=m(x)$, $m \in M$ significa que m transforma x en x^* , siendo m un movimiento o intercambio simple.

Como ya se mencionó, un ingrediente importante en la búsqueda tabú es el concepto de vecindad, la cual se denota por $N(x)$. En cada paso del proceso iterativo se genera un subconjunto $V^* \subset N(x)$ con k elementos. La idea es alcanzar la mejor solución x^* en V^* , partiendo de x , independientemente de que $F(x^*)$ sea mejor o no que $F(x)$. En general, si $N(x)$ no es muy grande se puede tomar $V^*=N(x)$.

Los movimientos prohibidos se manejan por medio de un arreglo llamado lista tabú, la cual puede tener un tamaño fijo o variable. Esta característica evita que una región se visite repetidamente a lo largo del proceso. Cuando a un movimiento se le asigna el estatus tabú, significa que permanece prohibido durante las siguientes t iteraciones. El propósito es impedir movimientos entre soluciones que reincorporen ciertos atributos de soluciones anteriores.

4. SISTEMA BUTREN-RC

Utilizando las dos técnicas presentadas en la sección anterior, se construyó el sistema híbrido llamado BUTREN-RC (BUSqueda Tabú y REdes Neuronales para Recargas de Combustible). Como ya se dijo antes, este sistema híbrido utiliza una red neuronal previamente entrenada, para sustituir al simulador del reactor en el proceso de búsqueda y acelerarlo. En la referencia [13] se explica con detalle cómo se entrenó la red neuronal, sin embargo, incluimos en este trabajo lo más relevante del entrenamiento de dicha red neuronal.

Se entrenó una red neuronal por cada ciclo considerado. Estas redes fueron construidas con 2 capas de neuronas además de la capa de entrada. Como entrada a la red neuronal se tienen 111 neuronas que reciben el valor del factor de multiplicación infinito de neutrones (k_{∞}) menos 1, expresado en pcm's, correspondientes a los 111 ensambles de combustible de $\frac{1}{4}$ parte del núcleo de los reactores bajo estudio. En la capa oculta se tienen 10 neuronas y en la capa de salida se tienen 5 neuronas que proporcionan los valores siguientes: el FPPR, el ERC, el factor de multiplicación efectivo de neutrones (k_{ef}) al final del ciclo de operación (para un quemado determinado), el MCPR al final del ciclo y el MLHGR al final del ciclo. El final del ciclo fue aprendido por la red neuronal de acuerdo a un cálculo Haling para la longitud de cada uno de los ciclos considerados. De este modo, si BUTREN-RC encuentra valores de k_{ef} al final del ciclo, mayores a los que se obtuvieron en los ciclos considerados, entonces se tendrá la garantía de que con las recargas propuestas, se extrae más energía del combustible.

Las redes neuronales fueron entrenadas con muestras de entrenamiento obtenidas con el código de simulación CM-PRESTO [15]. Es decir, se genera una recarga de combustible aleatoria y se ejecuta con el simulador, entonces se toman los 111 valores de $k_{\infty}-1$, así como los 5 valores que

caracterizan a la recarga. Esto se repite para un total de 1200 recargas de combustible por cada ciclo. La calidad del entrenamiento de las redes fue probada con otras 300 recargas no vistas durante el aprendizaje y se comprobó que las predicciones estaban dentro de los rangos de tolerancia esperados. Estos rangos fueron los siguientes:

$$\begin{aligned} |FPPR^{RN} - FPPR^{CM-PRESTO}| &< 0.05 \\ |k_{ef}^{RN} - k_{ef}^{CM-PRESTO}| &< 0.001 \\ |MCPR^{RN} - MCPR^{CM-PRESTO}| &< 0.03 \\ |MLHGR^{RN} - MLHGR^{CM-PRESTO}| &< 10 \text{ W/cm} \end{aligned}$$

donde los superíndices RN denotan predicciones de la red neuronal y los superíndices CM-PRESTO denotan los cálculos realizados por el simulador del reactor. Esta metodología es válida para cualquier ciclo que se estudie, pues las recargas de entrenamiento son creadas de forma aleatoria. El entrenamiento de la red neuronal requirió de aproximadamente 30 minutos (tiempo real) en una PC a 350 MHz y 16 MB de RAM.

Por otro lado, el proceso de optimización de las recargas de combustible se realizó con la búsqueda tabú. A continuación se detallan algunos aspectos importantes de la implementación.

En la implementación se manejó un tamaño de lista tabú variable, construyendo una matriz llamada *tiempo_tabu*, la cual es un arreglo que permite verificar los movimientos prohibidos en cada iteración. En este caso el tamaño de la lista está entre 7 y 15. Este intervalo se obtuvo mediante un análisis estadístico.

La función objetivo empleada para la búsqueda tabú es la siguiente:

$$F = w_1 k_{ef}^{EOC} + w_2 A(FPPR) + w_3 B(ERC) + w_4 C(MLHGR) + w_5 D(MCPR) \quad (10)$$

donde

$$A(FPPR) = \begin{cases} 0 & \text{si } FPPR \leq FPPR_{Max} \\ -|FPPR - FPPR_{Max}| & \text{en otro caso} \end{cases} \quad (11)$$

$$B(ERC) = \begin{cases} 0 & \text{si } ERC \leq ERC_{Max} \\ -|ERC - ERC_{Max}| & \text{en otro caso} \end{cases} \quad (12)$$

$$C(MLHGR) = \begin{cases} 0 & \text{si } MLHGR \leq MLHGR_{Max} \\ -|MLHGR - MLHGR_{Max}| & \text{en otro caso} \end{cases} \quad (13)$$

$$D(MCPR) = \begin{cases} 0 & \text{si } MCPR \geq MCPR_{Min} \\ -|MCPR - MCPR_{Min}| & \text{en otro caso} \end{cases} \quad (14)$$

con

k_{ef}^{EOC} es el factor de multiplicación de neutrones al final del ciclo para un quemado determinado,
 $FPPR_{Max}$ es el máximo valor permitido del FPPR al inicio del ciclo,
 ERC_{Max} es el máximo valor permitido para el Exceso de Reactividad en Caliente al inicio del ciclo,
 $MLHGR_{Max}$ es el máximo valor permitido para el MLHGR al final del ciclo,
 $MCPR_{Min}$ es el mínimo valor permitido para el MCPR al final del ciclo,
 w_i son factores de peso.

Los valores de las variables límite, involucradas en la Ec. (10), se muestran en la Tabla I. Asimismo, se indican los valores de los pesos asociados a cada uno de ellos. Cabe mencionar que los valores de los pesos se determinaron haciendo un análisis del rango de valores que puede tomar cada variable.

Estos pesos se mantuvieron constantes para los cinco ciclos estudiados. Debido a que la evaluación de la función objetivo se realiza con ayuda de la red neuronal y su costo es mínimo, se decidió revisar la vecindad completa, es decir, $V^* = N(x)$.

Tabla I. Valores límite y factores de peso para la Función Objetivo

Variable	Límite Permisible	Factor de Peso
k_{ef}^{EOC}	-	1.00
$FPPR_{Max}$	1.5500	10.00
ERC_{Max}	1.0250	100.00
$MLHGR_{Max} (W/cm)$	430.00	0.01
$MCPR_{Min}$	1.45	2.00

Adicionalmente se incorporó un criterio de aspiración, el cual permite que se ignore el estatus tabú, en caso de que la solución encontrada sea mejor que todas las soluciones encontradas hasta el momento. De igual forma, se maneja un vector de frecuencias V_f , que contiene el historial de movimientos durante el proceso, dicho vector se inicializa con valores iguales a cero en todas sus entradas y se actualiza de la siguiente forma:

$$V_f(i) = V_f(i) + 1 \tag{15}$$

$$V_f(j) = V_f(j) + 1 \tag{16}$$

donde i y j son las posiciones de un movimiento propuesto en una iteración dada.

El vector señalado se utiliza para evaluar la función objetivo de la siguiente forma:

$$F = F - V_f(i) - V_f(j) \tag{17}$$

La evaluación anterior, permite diversificar la búsqueda hacia otras regiones no exploradas.

Finalmente debe indicarse que la red neuronal se utiliza durante todo el proceso de búsqueda y cuando se tiene una recarga de combustible aceptable, ésta es simulada con CM-PRESTO para corroborar los resultados.

5. RESULTADOS

Como ya se mencionó, se tomaron 5 ciclos de la Unidad 1 de la CNLV, éstos fueron los ciclos 2 a 6. Se hicieron ejecuciones del sistema BUTREN-RC para todos los ciclos y algunos resultados se muestran en la Tabla II utilizando los valores corroborados con el simulador CM-PRESTO. En esta tabla se proporciona la ganancia de energía extraída en porcentaje con respecto a la energía obtenida en los diseños de cada uno de los ciclos estudiados. Además se muestra la ganancia obtenida, también en porcentaje, con las recargas propuestas por los sistemas RENOR y RECOPIA. Con letra oscura se resaltan las mayores ganancias de energía encontradas.

Como se ve, las recargas del sistema BUTREN-RC logran extraer más energía que en los diseños originales de la planta. La ganancia promedio de energía es de casi 4.5 %. Así mismo, dichas recargas de combustible también superan a las encontradas por el Algoritmo Genético. En cambio, al compararse con la red neuronal, BUTREN-RC logra encontrar mejores recargas de combustible en 3 de los cinco ciclos estudiados. La diferencia entre BUTREN-RC y RENOR en el ciclo 3 es de 1% de ganancia de energía, mientras que en el ciclo 4 la diferencia es de poco más de 2.5%.

Tabla II. Comparación de la ganancia de energía extraída (%) con las recargas propuestas por BUTREN-RC y con otros sistemas.

Ciclo	RECOPIA	RENOR	BUTREN-RC
2	0.97	1.13	1.99
3	2.83	4.17	3.16
4	4.16	8.25	5.61
5	2.00	2.42	5.17
6	2.76	4.26	6.06

La función objetivo empleada por RECOPIA contempla las mismas variables empleadas en BUTREN-RC, sin embargo RENOR utiliza una función que maximiza la k_{ef} al final del ciclo de operación y los aspectos de seguridad son considerados como restricciones que si no cumplidas se descarta la recarga de combustible.

En las Figura 2 a 6 se presentan los perfiles radiales de potencia relativa de las recargas encontradas por BUTREN-RC y se comparan con las de diseño de los ciclos estudiados (denotados por LV en la figura). En la Figura 7 se presenta la regionalización radial considerada en las figuras.

Estas figuras muestran que BUTREN-RC tiende a crear una zona de alta potencia en torno a las regiones 5 a 7, mientras que en las recargas originales no se aprecia lo mismo. Esto tiene como ventaja que no se tengan picos excesivos de potencia en el centro que conduzcan a la violación de límites térmicos. La otra ventaja obtenida, fue la generación de más energía con el mismo combustible.

El sistema encuentra recargas de combustible satisfactorias en 30 segundos en una Alpha a 233 MHz y 128 MB de RAM, con un promedio de 200 iteraciones y 73500 evaluaciones de la función objetivo en promedio.

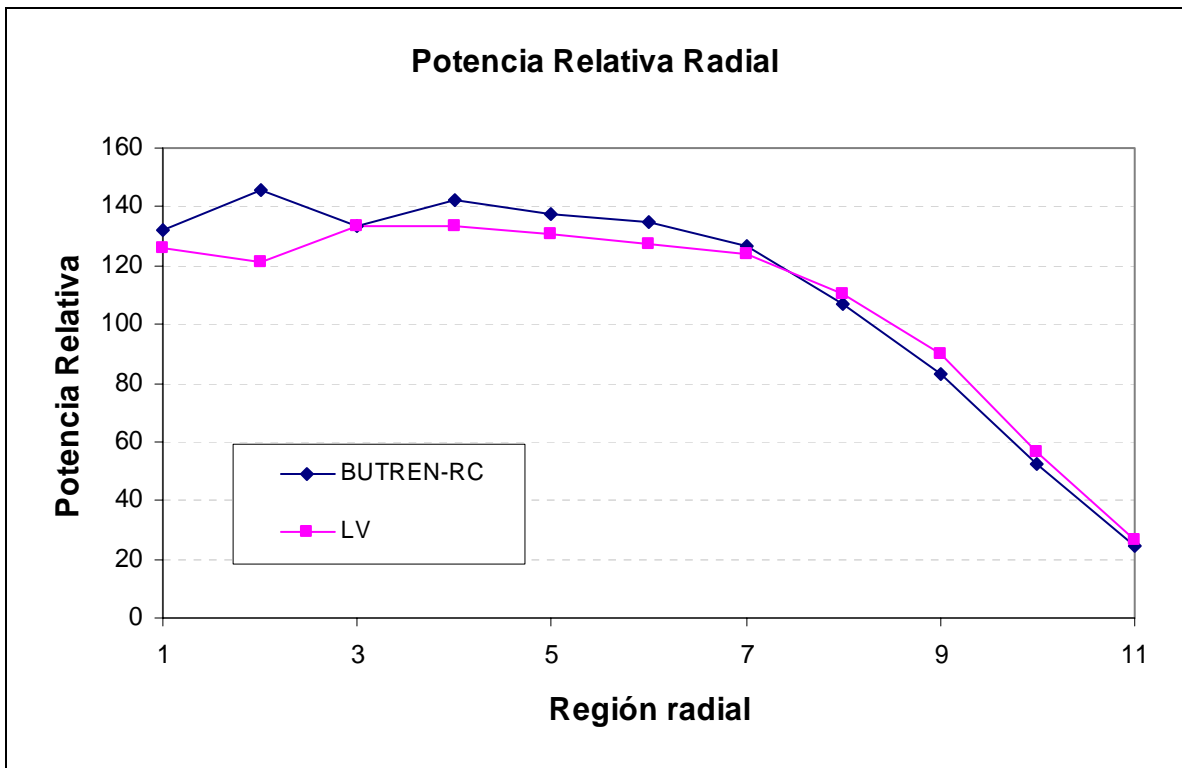


Figura 2. Comparación de la potencia relativa radial en el ciclo 2.

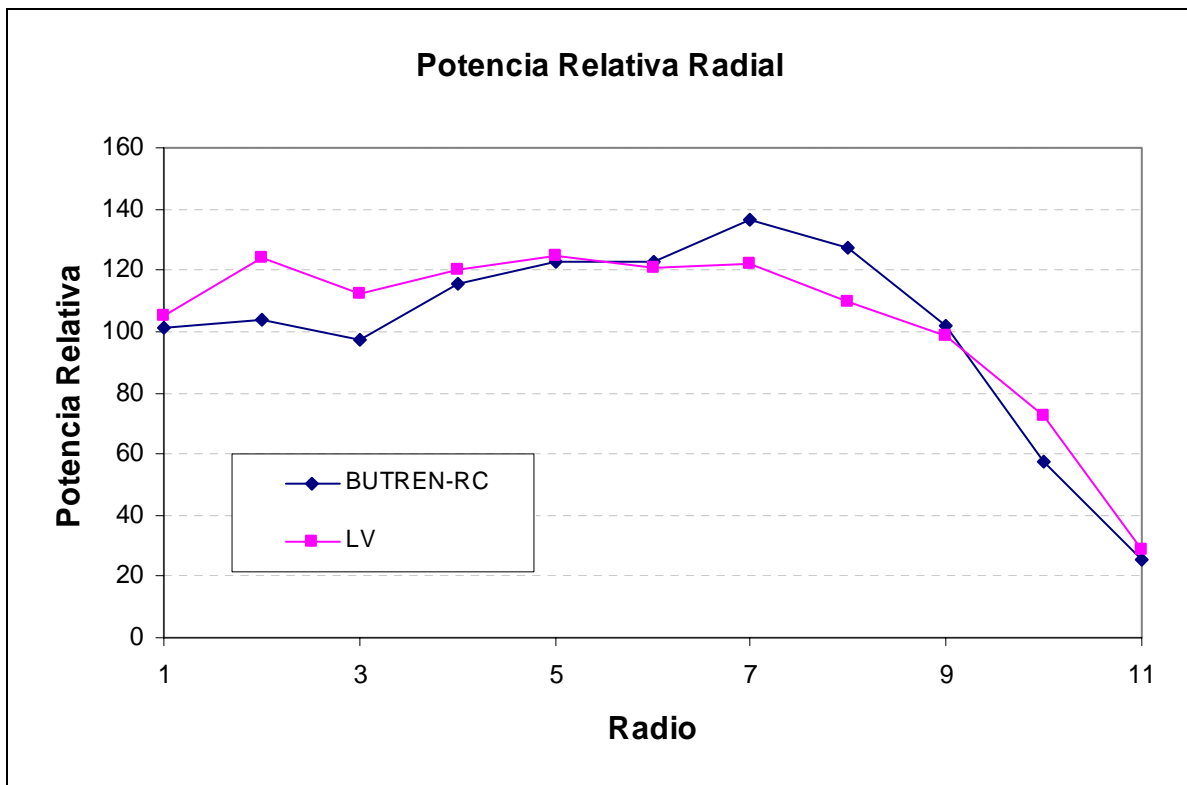


Figura 3. Comparación de la potencia relativa radial en el ciclo 3.

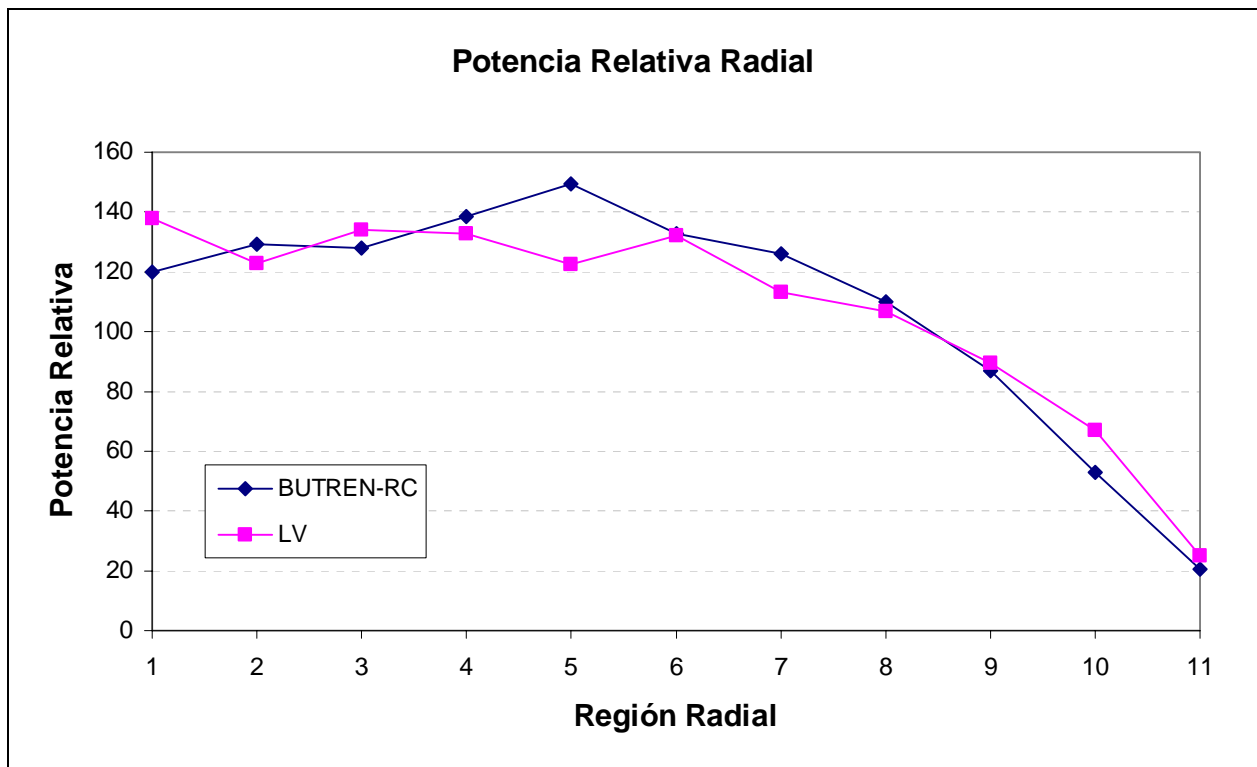


Figura 4. Comparación de la potencia relativa radial en el ciclo 4.

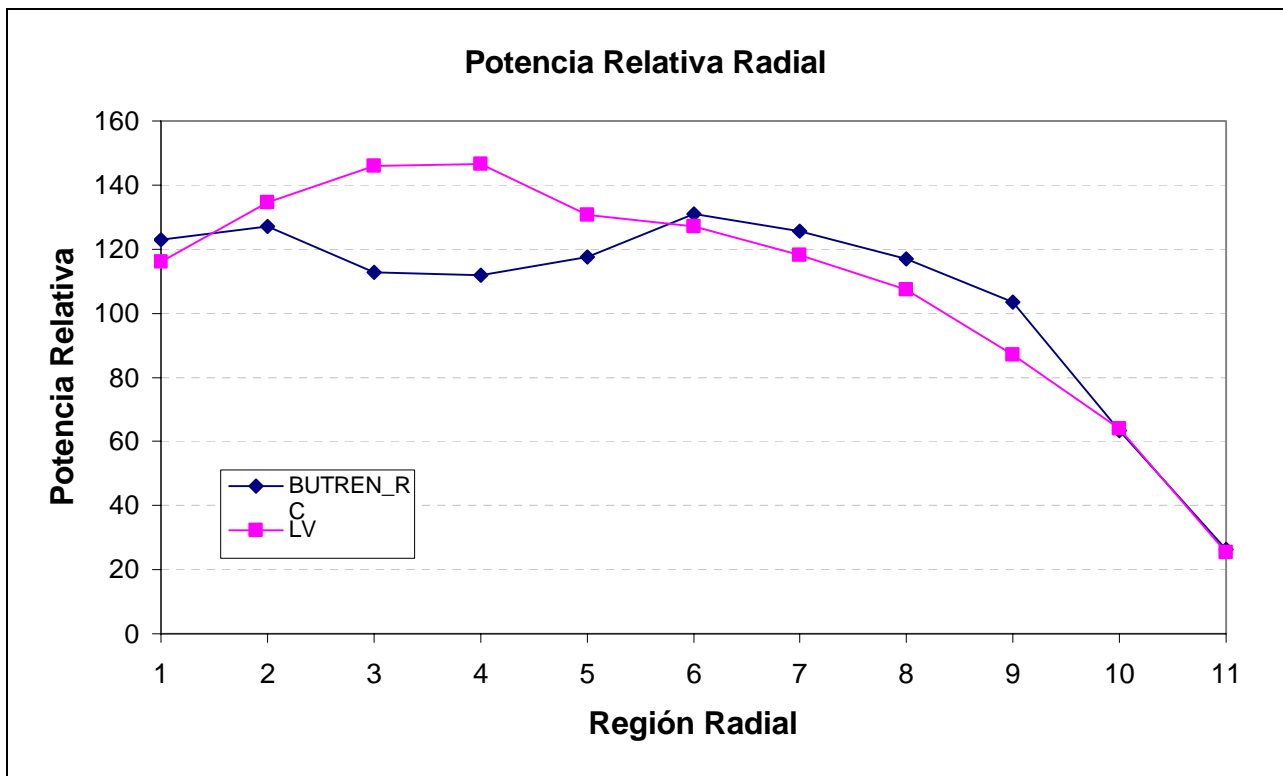


Figura 5. Comparación de la potencia relativa radial en el ciclo 5.

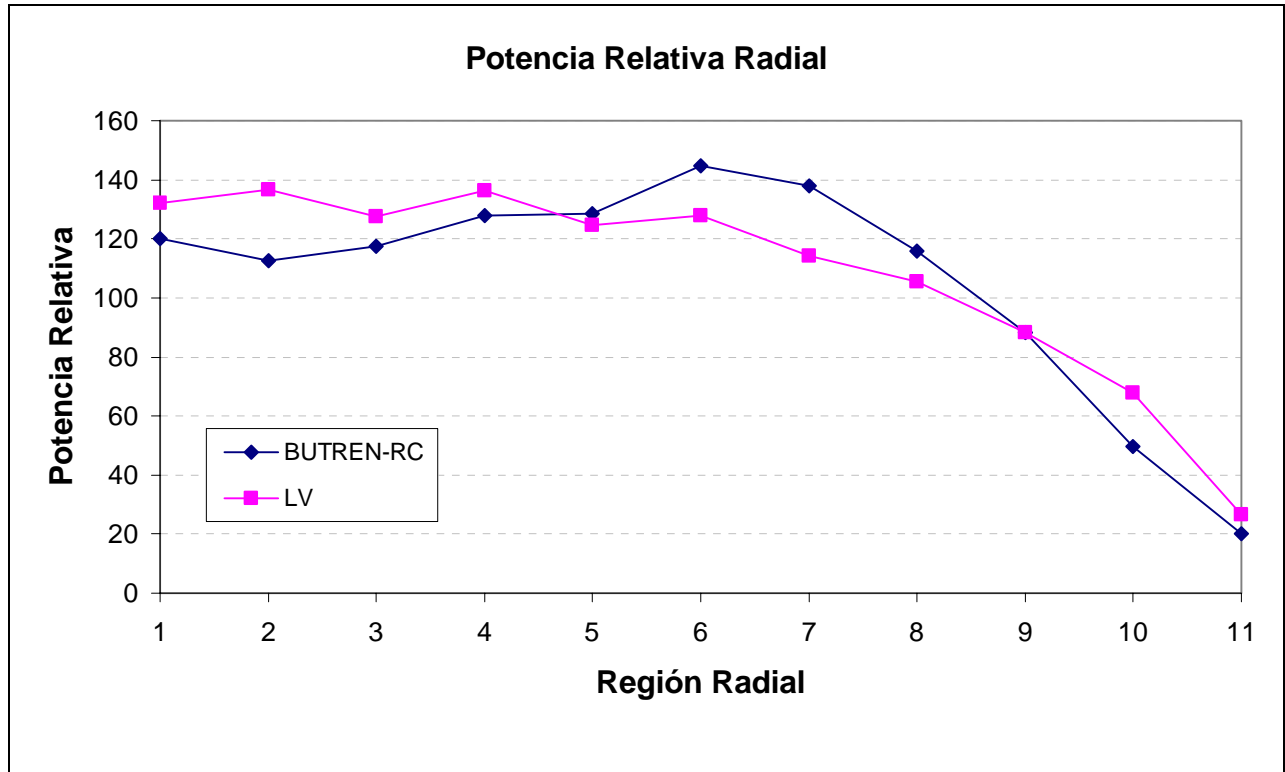


Figura 6. Comparación de la potencia relativa radial en el ciclo 6.

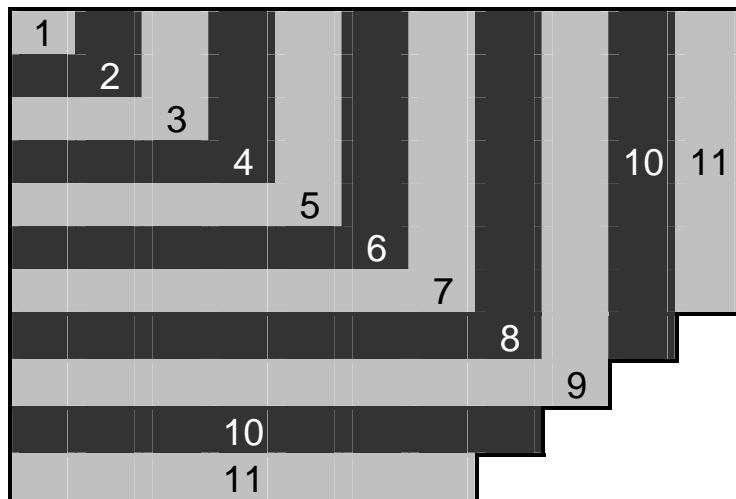


Figura 7. Mapa del núcleo donde se muestra la regionalización radial utilizada.

6. CONCLUSIONES

En este trabajo se presentó el sistema híbrido BUTREN-RC basado en la técnica de búsqueda tabú y redes neuronales. Las redes neuronales se emplean para predecir el comportamiento del reactor bajo los esquemas de recarga de combustible propuestos por el método de búsqueda tabú.

El sistema fue probado con cinco ciclos de la Unidad 1 de la Central Nuclear de Laguna Verde, y se compararon las recargas encontradas con las utilizadas en dichos ciclos, así como las obtenidas con los sistemas RECOPIA (basado en Algoritmos Genéticos) y RENOR (basado en una red neuronal de tipo Hopfield).

El sistema BUTREN-RC encontró recargas de combustible que superan a las recargas de referencia, en la cantidad de energía extraída del combustible. Asimismo, este sistema también supera al sistema basado en el Algoritmo Genético. Por otro lado, BUTREN-RC supera al sistema basado en la red neuronal del tipo Hopfield en los ciclos 2, 5 y 6. Aunque debe notarse que las diferencias entre ambos sistemas no es muy grande en los ciclos 3 y 4.

En general se concluye que con el sistema BUTREN-RC se obtienen recargas de combustible que son satisfactorias sin grandes requerimientos de cómputo. La red neuronal es capaz de ofrecer una respuesta en tiempos no mayores a 1 milisegundo, lo cual favorece que la optimización de una recarga de combustible se pueda tener en unos 30 segundos.

Como trabajo futuro se tiene contemplado entrenar redes neuronales para predecir el margen de apagado en frío e incorporarlo a los sistemas BUTREN-RC, RECOPIA y RENOR. Esto permitirá diseñar esquemas de recarga de combustible más realistas. También se modificará la función objetivo con el objeto de que el ERC se encuentre en una ventana de reactividad: el valor máximo evitará aceptar recargas muy reactivas y el valor mínimo ayudará a cumplir con los requerimientos de energía al no aceptar recargas poco reactivas.

Existen otras variantes que se pueden agregar al sistema como pueden ser el quitar las reglas mencionadas en la Sección 2. Entonces sería necesario agregar penalizaciones en la función objetivo cuando no se cumplen las reglas. De este modo el sistema debe ser capaz de inferir o demostrar la validez de las reglas.

AGRADECIMIENTOS

El segundo y tercer autores agradecen al Consejo Nacional de Ciencia y Tecnología por el apoyo recibido a través del Proyecto de Investigación 33806-U.

REFERENCIAS

1. J. L. François and H. A López, "SOPRAG - A system for boiling water-reactors reload pattern optimization using genetic algorithms", *Annals of Nuclear Energy*. **Vol. 26** pp. 1053-1063 (1999).

2. J.L. Francois, C. Martin del Campo, C.C. Cortes, E. Ramirez, J. Arellano, "Development of an Automated System for Fuel Reload Patterns Design", *Nuclear Engineering and Design*. **Vol. 193**, 13-21, 1999.
3. Alejandro Castillo, Gustavo Alonso, L. B. Morales, Cecilia Martín del Campo, J. L. Francois, Edmundo del Valle, "BWR fuel reloads design using a Tabu search technique", *Annals of Nuclear Energy*. **Vol. 31** pp. 151-161 (2004).
4. J. J. Ortiz e I. Requena, "An order coding genetic algorithm to optimize fuel reload in a nuclear boiling water reactor", *Nuclear Science & Engineering* **Vol. 146** pp. 88-98 (2004).
5. J. J. Ortiz e I. Requena. "Using a multi-state recurrent neural network to optimize fuel reloads in BWRs", *Annals of Nuclear Energy*, **Vol. 31** pp. 789-803 (2004).
6. S. R. Specker, L. E. Fennern, R. E. Brown, K. L. Stark, and R. L. Crowther, "The BWR control cell core improved design. *Transactions of American Nuclear Society*. **Vol. 30** pp. 336-338 (1978).
7. M. Caudill and C. Buttler, "*Understanding neural networks: Computer explorations, Vol. 1*" Basic network. MIT Press. (1994).
8. J. A. Freeman y D. M. Skapura. *Redes Neuronales. Algoritmos, Aplicaciones y Técnicas de Programación*. Addison-Wesley/Diaz de Santos. USA (1993).
9. A. K. Jain, J. Mao & K. M. Mohiuddin, "Artificial neural networks: a tutorial", *IEEE Computer*, pp. 31-44 (1996).
10. F. Glover, "Tabu Search – Part I", *ORSA Journal on Computing*. **Vol 1** pp. 190-206 (1986).
11. Manuel Laguna, "A Guide to Implementing Tabu Search", *Investigación Operativa*. **Vol. 4** pp. 5-25 (1994).
12. F. Glover, M. Laguna. *Tabu Search*. Kluwer Academic Publishers. USA (2001).
13. J. J. Ortiz & I. Requena. Using neural networks to predict core parameters in a boiling water reactor. *Nuclear Science & Engineering*. **Vol. 143**. Pp 1-14. 2003.
14. Scandpower AS. *User Manual CM-Presto 9. Version CM9I4B*. Rev. 6, July 16, 1993.