



MX0300162

Una comparación entre algoritmos genéticos y redes neuronales para optimizar recargas de combustible en BWR'S

Ortiz J. J.^a y Requena I.^b

^a Depto. Sistemas Nucleares. ^b Universidad de Granada, España.

Resumen

En este trabajo se presentan resultados de un Algoritmo Genético (AG) y una Red Neuronal Recurrente Multi Estado (RNRME) para optimizar la recarga de combustible de 5 ciclos de la Central Nuclear de Laguna Verde (CNLV). Las recargas de combustible obtenidas por ambos métodos son comparadas y se observa que la RNRME crea mejores distribuciones de combustible que el AG. También se hace una comparación de la conveniencia de usar una u otra técnica.

Introducción

El diseño de la recarga de combustible para un ciclo de operación de un reactor nuclear es un problema complejo, máxime si es un reactor de agua en ebullición (BWR: Boiling Water Reactor). El problema es el siguiente:

Hay que colocar N elementos combustibles (EC) en igual número de canales en el núcleo del reactor de modo que se garantice la seguridad de cada uno, que el reactor sea crítico y que se extraiga la máxima energía a todos ellos durante la operación del reactor.

En la literatura se mencionan varios trabajos donde se resuelve este problema aplicando sistemas basados en reglas¹, algoritmos genéticos², redes neuronales³, búsqueda tabú⁴, simulated annealing⁵, etc. Sin embargo muchos de esos trabajos se hicieron para reactores de agua a presión (PWR: Pressure Water Reactor). Otro aspecto en común en muchos de esos trabajos es que no contemplan la colocación de los EC's manteniendo una simetría entre las distintas zonas del reactor. Algunos hacen el análisis trabajando con una octava parte del núcleo, y agregan que bastaría con colocar EC's de similares características en las posiciones correspondientes en las otras 7 partes del reactor. Las complicaciones surgen cuando no existen EC's de similares características en cantidad suficiente para ser colocados en dichas posiciones.

Lo ideal entonces sería hacer el estudio usando todo el núcleo, pero esto haría muy lento el proceso porque cada recarga de combustible debe ser evaluada con un simulador del reactor para ver si cumple con las restricciones. Un punto intermedio es utilizar una simetría de $\frac{1}{4}$ porque al simulador le lleva menos tiempo hacer la evaluación y porque será más probable encontrar 4 EC's de similares características.

Otra característica encontrada en algunos trabajos reportados en la literatura es que se hace una optimización parcial, es decir que se busca maximizar una variable pero no se observan otras igualmente importantes.

En AEB2002⁶ y en FLINS2002⁷ presentamos resultados preliminares para el Ciclo 4 de la Unidad 1 de la CNLV, usando AG y RNRME respectivamente. En estos trabajos se contempló la generación de recargas de combustible usando una simetría de $\frac{1}{4}$ del núcleo, además se buscó maximizar los valores de K_{eff} tanto al inicio como al final del ciclo sin descuidar los aspectos de seguridad. En este trabajo extendemos el estudio a 5 ciclos de la Unidad 1 de la CNLV.

En la Sección 2 se exponen brevemente las características de un AG y de una RNRME para optimizar la recarga de combustible de un reactor BWR. En la Sección 3 se presentan resultados para 5 ciclos de la CNLV. Una discusión de los resultados y las bondades y deficiencias de cada técnica son presentadas en la Sección 4. Finalmente, se presentan unas conclusiones en la última sección.

Algoritmos Genéticos y Redes Neuronales

Una descripción completa de la forma en que se aplicaron estas técnicas a la optimización de la recarga de combustible se puede encontrar en ⁶ y ⁷. Aquí solo se mencionan brevemente las características principales de ellas.

a Algoritmo Genético

El AG empleado usa una codificación de orden⁸ en los cromosomas. Esto significa que cada gen del cromosoma se puede ver como un canal del reactor (hay 111 canales en $\frac{1}{4}$ del núcleo) y que la etiqueta asignada al gen indicará el EC que se coloca en el canal correspondiente (hay 111 etiquetas y cada una identifica a un EC).

La cualificación de un cromosoma para representar de forma adecuada la solución a un problema se hace por medio de una función de fitness. En este caso la función fitness depende de los valores de k_{ef} y de que se satisfagan los aspectos de seguridad. Mientras más alto sean los valores de k_{ef} (al inicio y final del ciclo) mayor será el valor de la función.

Los operadores de mutación y cruce del AG son los clásicos⁹, con probabilidades del 5 y 70% respectivamente. El operador de cruce tiene por objeto mezclar los genes de dos cromosomas padre para crear nuevos cromosomas. El operador de mutación intercambia de forma aleatoria el contenido de 2 genes. La secuencia de ejecución del AG es:

1. Crear una población aleatoria de 50 cromosomas (50 recargas de combustible aleatorias).
2. Evaluar los cromosomas con ayuda de un simulador del reactor u otro modelo disponible.
3. Evaluar la función de fitness de cada cromosoma y ordenarlos de mayor a menor valor de dicha función.
4. Seleccionar una población de padres de acuerdo al método de la ruleta.
5. Crear los descendientes por medio de cruces entre los padres seleccionados y de acuerdo a la probabilidad de cruce.
6. Aplicar el operador de mutación de acuerdo a la probabilidad de mutación.
7. La nueva población se forma con los descendientes obtenidos en el paso 6 y con los padres que no se reprodujeron en el paso 5.

8. Repetir los pasos 2 a 7 hasta que el valor de fitness de los cromosomas no cambie en 10 generaciones sucesivas.

El sistema RECOPIA (REcarga de COM-bustible Por Inteligencia Artificial) fue desarrollado usando el algoritmo anterior.

b Red Neuronal Recurrente Multi Estado

Esta red neuronal fue propuesta por Mérida¹⁰. Consiste de una capa de N neuronas completamente conectadas entre sí. Cada neurona tiene asociado un estado neuronal, donde un estado neuronal es un número entre 1 y N. En nuestro caso, N vale 111 porque se tienen 111 canales en $\frac{1}{4}$ del núcleo del reactor. Cada neurona de la RNRME representa un canal del reactor y su estado asociado indica el número de EC asignado al canal (previamente los EC's se ordenan de acuerdo a algún criterio).

Para trabajar adecuadamente, la RNRME requiere de dos especificaciones: una función de energía y una regla de transición de estados entre las neuronas.

La regla de transición de estados entre neuronas es la forma en que dos o más neuronas intercambian sus estados. En este caso se hace escogiendo dos neuronas aleatoriamente y se intercambian sus estados; esto es equivalente a intercambiar 2 EC's de distintos canales. Además, para de mantener la simetría en cada octavo del núcleo, las neuronas seleccionadas para el intercambio de estado solo pueden ser de un octavo del núcleo y por consiguiente, las neuronas correspondientes del otro octavo del núcleo también intercambian sus estados.

La función de energía cuantifica la energía total de la red cuando todas las neuronas tienen asociado un estado. Durante el proceso de intercambios de estado, es deseable que el valor de la función de energía disminuya y se estabilice en un mínimo. En nuestro caso, se desea maximizar el valor de k_{ef} al inicio del ciclo por medio de la regla de transición de estados. Por eso es conveniente definir una función de energía adecuada. En ⁷ se utilizó la siguiente función que puede ser derivada a partir de la Ecuación de Difusión de Neutrones en dos grupos de energía¹¹:

$$F = \frac{\sum_{i=1}^{111} \phi_{2i} \left[\frac{\nu_1 \Sigma_{f1}^i \Sigma_{2a}^i}{\Sigma_{1 \rightarrow 2}^i} + \nu_2 \Sigma_{f2}^i \right] + \Delta N_{jk}}{\sum_{i=1}^{111} \phi_{2i} \Sigma_{2a}^i \left[\frac{\Sigma_{1a}^i}{\Sigma_{1 \rightarrow 2}^i} + 1 \right] + \Delta D_{jk}}$$

donde intervienen las secciones eficaces de absorción (Σ_{1a} y Σ_{2a}), dispersión ($\Sigma_{1 \rightarrow 2}$) y fisión (Σ_{f1} y Σ_{f2}) de neutrones rápidos y térmicos; también está involucrado el flujo

térmicos de neutrones (ϕ_{2i}). El subíndice i se aplica a cada uno de los 111 EC's, lo cual quiere decir que las secciones eficaces involucradas deben ser para el EC. ΔN_{jk} y ΔD_{jk} se calculan por medio de:

$$\Delta N_{jk} = \left\{ \frac{V_1 \Sigma_{j1}^j \Sigma_{2a}^j}{\Sigma_{1 \rightarrow 2}^j} + V_2 \Sigma_{j2}^j \right\} \phi_j + \left\{ \frac{V_1 \Sigma_{j1}^k \Sigma_{2a}^k}{\Sigma_{1 \rightarrow 2}^k} + V_2 \Sigma_{j2}^k \right\} \phi_j -$$

$$\left\{ \frac{V_1 \Sigma_{j1}^j \Sigma_{2a}^j}{\Sigma_{1 \rightarrow 2}^j} + V_2 \Sigma_{j2}^j \right\} \phi_j + \left\{ \frac{V_1 \Sigma_{j1}^k \Sigma_{2a}^k}{\Sigma_{1 \rightarrow 2}^k} + V_2 \Sigma_{j2}^k \right\} \phi_j$$

$$\Delta D_{jk} = \left\{ \phi_{2k} \Sigma_{2a}^j \left[\frac{\Sigma_{1a}^j}{\Sigma_{1 \rightarrow 2}^j} + 1 \right] + \left[\phi_{2j} \Sigma_{2a}^k \left[\frac{\Sigma_{1a}^k}{\Sigma_{1 \rightarrow 2}^k} + 1 \right] \right\} -$$

$$\left\{ \phi_{2j} \Sigma_{2a}^j \left[\frac{\Sigma_{1a}^j}{\Sigma_{1 \rightarrow 2}^j} + 1 \right] + \left[\phi_{2k} \Sigma_{2a}^k \left[\frac{\Sigma_{1a}^k}{\Sigma_{1 \rightarrow 2}^k} + 1 \right] \right\}$$

en estas expresiones los primeros dos sumandos presentan una contribución al valor de F cuando la neurona j intercambia su estado con la neurona k ; en cambio los 2 últimos sumandos restan la contribución de las neuronas cuando tenían asociados los estados k y j respectivamente.

Debido a éstas dos últimas ecuaciones no es necesario evaluar la expresión de F para conocer la energía del nuevo estado neuronal. Basta con que $\Delta N_{jk} > \Delta D_{jk}$, para que F aumente de valor; por el contrario si $\Delta N_{jk} < \Delta D_{jk}$, F disminuirá su valor. Tomando $k_{ef} = F$, estaremos maximizando su valor. Los pasos del algoritmo son:

1. Proponer de manera aleatoria un estado para la red neuronal que cumpla con las condiciones de simetría.

2. Escoger aleatoriamente una neurona a y su simétrica correspondiente.

3. Para cada una de las neuronas restantes en el cuarto de núcleo, intercambiar su estado con la neurona a y evaluar el nuevo estado neuronal con un simulador del reactor. Si no cumple con los criterios de seguridad, desear el intercambio. Si los cumple, emplear las ecuaciones para ΔN_{jk} y ΔD_{jk} y guardar ambos valores.

4. Se aceptará como nuevo estado de la red aquel intercambio de estados neuronales cuyo $\Delta N_{jk} > \Delta D_{jk}$ sea mayor.

5. Repetir los pasos 2 a 4 hasta que la función de energía no cambie o se tenga una solución aceptable.

El sistema RENOR (REdes Neuronales para Optimizar Recargas) fue desarrollado utilizando el algoritmo mencionado antes.

Resultados y discusión

Se utilizaron los lotes de recarga de los ciclos 2 a 6 de la Unidad 1 de la CNLV y se crearon nuevas recargas de combustible utilizando las dos técnicas descritas anteriormente.

Cabe mencionar que no se empleó un simulador del reactor para evaluar el comportamiento de las recargas de combustible generadas en cada etapa del proceso. En su lugar se usó una red neuronal basada en retropropagación de errores y que fue entrenada con los lotes de recarga de los ciclos mencionados. En ¹¹ se puede encontrar una descripción detallada de la forma del entrenamiento.

La Tabla I muestra la longitud del ciclo para las recargas de combustible encontradas por RECOPIA y por RENOR, además se muestran estas longitudes del ciclo con las que realmente tuvieron los ciclos estudiados. Con números gruesos se resaltan las mayores longitudes del ciclo.

Se puede apreciar en la Tabla I que las recargas creadas por RENOR y por RECOPIA para los 5 ciclos, tienen mayores longitudes del ciclo que las que tuvieron en la CNLV. También se observa que las recargas de RENOR tienen mayores longitudes del ciclo que las de RECOPIA.

De los resultados presentados se concluye que la RENOR es mejor método que RECOPIA para optimizar la recarga de combustible. Además, analizando las dos técnicas se puede observar que la RNRME presenta otras ventajas con respecto al AG.

Tabla I: Comparación de las longitudes del ciclo de las recargas propuestas por distintas técnicas.

Sistema\Ciclo		2	3	4	5	6
CNLV	Días	247.1	312.9	314.9	392	414.8
	MWD/T	5893	7488	7575	9461	10023
RECOPIA	Días	249.5	321.7	328	399.8	426.2
	MWD/T	5950	7700	7890	9650	10300
RENOR	Días	249.7	328	340.9	401	432.4
	MWD/T	5960	7800	8200	9690	10450

El AG tiene muchos parámetros que deben ser definidos como, las probabilidades de cruce y mutación, el ta-

maño de la población de cromosomas, la forma de la función de fitness y las constantes que puedan intervenir.

En cambio en la RNRME solo se definió una función de energía adecuada. También se utilizó una expresión para calcular k_{ef} por medio de la Teoría de Perturbaciones, pero se observó que no había coherencia entre la maximización de k_{ef} dada por la teoría, y la k_{ef} dada por el simulador del reactor.

Otra cualidad de la RNRME es que necesitó menor cantidad de iteraciones para encontrar la solución que el AG. Finalmente, se debe destacar que el empleo de una RN entrenada para predecir el comportamiento de una recarga de combustible, aceleró bastante el proceso de optimización. Como comparación, en ² se indica que SOPRAG obtuvo resultados satisfactorios después de evaluar 10000 recargas de combustible, que con el simulador CM-PRESTO le tomó alrededor de 4 horas de CPU. A los sistemas RENOR y RECOPIA una optimización con la RN entrenada les toma alrededor de 5 minutos.

Conclusiones

En este trabajo se presentaron resultados de dos técnicas basadas en inteligencia artificial para optimizar la recarga de combustible de un reactor BWR. Las técnicas son: un AG cuya función de fitness maximiza el valor de k_{ef} al inicio y final del ciclo; la otra técnica es una RNRME cuya función de energía maximiza el valor de k_{ef} al inicio del ciclo.

Se observó que la RNRME crea recargas de combustible con mayores longitudes del ciclo que las creadas por el AG; y ambas técnicas generan mejores recargas de combustible que las usadas en la CNLV.

Ambos sistemas utilizan los EC's que intervinieron en cada uno de los ciclos estudiados, esto quiere decir, que los sistemas RENOR y RECOPIA no emplean EC's de la piscina de combustible gastado. Dentro de los trabajos futuros se pretende modificar el operador de mutación del AG y que la RNRME contenga tantas neuronas como EC's hay disponibles al inicio de un ciclo y realizar una mejor optimización.

Referencias

1. Kim H. G., Chang S. H. & Lee B. H. 1993. "Optimal fuel loading pattern design using an artificial neural network and a fuzzy rule-based system". *Nucl. Sci & Eng*, V. 115, Pp. 152-163.
2. Francois J. L. & Lopez H. A. 1999. "Soprag - A System for Boiling Water-Reactors Reload Pattern Optimization Using Genetic Algorithms". *Ann of Nucl Energy*. V. 26, Pp 1053-1063.
3. Sadighi M., Setayeshi S. & Salehi A. A. 2002. "PWR fuel management optimization using neural networks". *Ann of Nucl Energy*. V. 29, Pp 41-51.
4. Lin C., Yang J. I., Lin K. J. & Wang Z. D. 1998. "Pressurized Water Reactor Loading Pattern Design Using The Simple Tabu Search". *Nucl Sci & Eng*. V. 129. Pp 61-71.
5. Mahlers Y. P. 1995. "Core Loading Pattern Optimization Based On Simulated Annealing And Successive Linear Programming". *Ann of Nucl Energy*. V 22 Pp. 29-37.
6. Ortiz J. J. & Requena I. 2002. "Computacion Flexible para Optimizar la Recarga de Combustible en un BWR". 1er Congreso Español de Algoritmos Evolutivos Bioinspirados, AEB2002. España. Pp. 146-152.
7. Ortiz J. J. & Requena I. 2002. "Optimization of Fuel Reload in a BWR nuclear reactor using a Recurrent Neural Network". 5th International FLINS Conference on Intelligent Techniques and SoftComputing in Nuclear Science and Engineering. Belgium. Pp. 544-551.
8. Larrañaga P, Kuijpers C. M. H, Murga R. H., Inza I. & Dizdarevic S. 1999. "Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators". *Artif. Intel. Rev*. V 13. Pp 129-170.
9. D. Goldberg E. 1989. "Genetic Algorithms in Search, Optimization and Learning Machine". Addison-Wesley.
10. Mérida C. E., Galán M. Muñoz-Pérez J. 2001. "An Efficient Multivalued Hopfield Network for the TSP". *Neural Proc Letters*. V 14. Pp 203-216.
11. Ortiz J.J. & Requena I. "Using Neural Networks to Predict Core Parameters in a Boiling Water Reactor". In Press in *Nuc. Sci & Eng*.