



EMPREGO DE REDES NEURAIS ARTIFICIAIS PARA OTIMIZAÇÃO DE RECARGA EM UM NÚCLEO DE UM REATOR PWR

Edyene Oliveira, Victor F. Castro, Carlos E. Velásquez, ¹Cláudia Pereira

¹claudia@nuclear.ufmg.br

Programa de Pós-Graduação em Ciências e Técnicas Nucleares

Universidade Federal de Minas Gerais

Av. Pres. Antônio Carlos, 6627

31270-901 Pampulha, Belo Horizonte, MG.

Abstract. Um reator, como, por exemplo, o PWR (Pressurized Water Reator) contém várias estruturas de tubos de zircônio preenchidas com urânio utilizadas como combustíveis. O Brasil possui dois reatores PWR em funcionamento atualmente, sendo o mais antigo, Angra I com 121 elementos combustíveis no núcleo. A recarga destas estruturas ou elementos combustíveis deve seguir certos padrões. O principal é manter três tipos de combustíveis, sendo o primeiro um conjunto novo, o segundo com meia vida e o terceiro mais gasto dos três lotes. Além disso, eles devem ser inseridos no núcleo misturados a fim de distribuir o calor e, principalmente, manter níveis de segurança. Desta forma, a recarga exige alta expertise por parte do engenheiro nuclear, pois, a entrada do problema é grande, neste caso, por exemplo, para 121 elementos combustíveis, existem 121 possibilidades. Há algumas décadas o tema vem sendo pesquisado e, vários algoritmos com viés heurísticos e evolucionários foram propostos, porém, ainda não foi encontrado resultado satisfatório para este problema. O objetivo deste trabalho foi criar um modelo que utiliza Rede Neural Artificial (RNA) que, ao receber um conjunto de posições de elementos combustíveis de um reator PWR, tenha como saída uma configuração otimizada. Foi gerado um modelo de contexto Elman e o algoritmo de treinamento utilizado foi o PSO (Particle Swarm Optimization). Os dados para testes foram obtidos a partir de códigos nucleares que são programas de computador específico para esta finalidade. O modelo de RNA Elman gerado neste trabalho manteve 87% de elementos combustíveis na posição original. Os outros 13% a RNA sugeriu trocar de posição.

Palavras Chaves: Inteligência Computacional, Otimização, Redes Neurais Artificiais, Reatores Nucleares, PWR

INTRODUÇÃO

Desde que os reatores nucleares começaram a ser utilizados para produção de energia elétrica existe preocupação em minimizar a queima e maximizar a produção. Vários estudos têm sido realizados a fim de estender o tempo de operação dos reatores, sobretudo dos PWR (Pressurized Water Reator). Tais modelos são os mais utilizados em todo o mundo desde sua fabricação (Cacucci, 2013).

Um reator PWR geralmente permanece em operação durante 18 ou 24 meses e, durante este período, o combustível nuclear, em sua maioria, urânio enriquecido (U_{235}) é queimado. Este processo causa desgaste no combustível, e assim, o reator vai perdendo potência ao longo do tempo (Cochran, 1999). Os operadores das centrais nucleares vêm buscando formas de aumentar este tempo, pois, a parada do reator significa perda financeira. Em países que possuem alta dependência de energia elétrica produzida por meio de reatores nucleares, é necessário recorrer à importação durante o tempo de parada do reator. Por isso, estudos têm sido realizados a fim de estender o período de operação do reator.

Na literatura foram encontrados trabalhos que utilizaram algoritmos evolutivos, além de RNAs. Os autores (Souza, 2006) usaram uma RNA do tipo RBF (Radial Base Function) a fim de obter configuração satisfatória do núcleo de um reator. Segundo os autores foi possível reduzir a margem de pico de potência em até 5%.

Um dos autores deste trabalho, (Faria, 2003), utilizou uma RNA feedforward juntamente com o algoritmo Backpropagation, a fim de otimizar o núcleo de um reator PWR. O trabalho consistiu em realizar os testes e validar a saída da RNA em um código nuclear.

Também, foram encontradas pesquisas de outros autores usando algoritmos baseado em comunidade social dos pássaros, como o PSO (Particle Swarm Optimization), realizados pelos autores (Augusto, 2011; Oliveira, 2015).

Na linha matemática, existe o trabalho de (Poursalehi, 2012) em que os autores usaram programação inteira. O método utilizado, segundo os autores, não só poderia economizar o consumo de combustível nuclear, mas também reduzir o ponto de potência radial.

Existem vários trabalhos desde a década de 1970 quando ainda não usavam computadores para estas finalidades, devido aos recursos e tecnologias limitadas. Porém, ainda hoje, não foi desenvolvido um modelo que pudesse receber um conjunto de dados de um núcleo de qualquer reator e pudesse indicar a configuração ideal. Neste caso, ideal significa otimizada, respeitando todos os critérios de segurança.

O uso da RNA neste trabalho é diferente de outros no sentido de que o objetivo da RNA foi indicar posições espaciais adequadas para cada elemento combustível. Não apenas manter o elemento no lugar original, mas propor melhores posições para ele garantindo segurança e nivelando os picos de cada elemento no núcleo.

Este trabalho começará descrevendo as RNAs além do algoritmo utilizado. Em seguida, a metodologia será detalhada, depois os resultados serão apresentados e comentados. Finalmente, serão feitos os comentários finais.

1. MATERIAIS E MÉTODOS

1.1 Redes neurais artificiais

Redes Neurais Artificiais (RNA) são estruturas que fazem parte de um conjunto de ferramentas computacionais com capacidade de generalizar e inferir (Rezende, 2005).

As RNAs são usadas em aplicações consideradas inteligentes como, por exemplo: reconhecimento de faces em visão computacional, classificação de padrões de escrita e de fala, avaliação de imagens captadas por satélites, identificação automática de perfis de crédito para clientes de instituições financeiras, classificação de carne bovina utilizando ressonância magnética nuclear e redes neurais artificiais, entre outros (Silva, 2016).

Segundo Haykin (2009) a RNA foi projetada com inspiração no cérebro humano. Este possui neurônios e sinapses. Ao receber uma informação o neurônio irá processá-la e poderá se comunicar com outros neurônios por meio das sinapses divulgando tal informação ou não, vai depender de vários fatores, como experiência, por exemplo. Ao nascer o cérebro possui grande estrutura e habilidade para desenvolver suas próprias regras. Isso irá acontecer à medida que o indivíduo obtém experiência. Esta fase ocorre com maior intensidade até os dois anos de vida, porém, o aprendizado continua bem além desta idade (Haykin, 2009).

Da mesma forma, uma Rede Neural Artificial possui modelo similar ao cérebro com seus neurônios, projetada para realizar tarefas particulares. São construídas por meio de programação em um computador digital (Russel, 2010). A Figura 1 mostra exemplo de uma RNA.

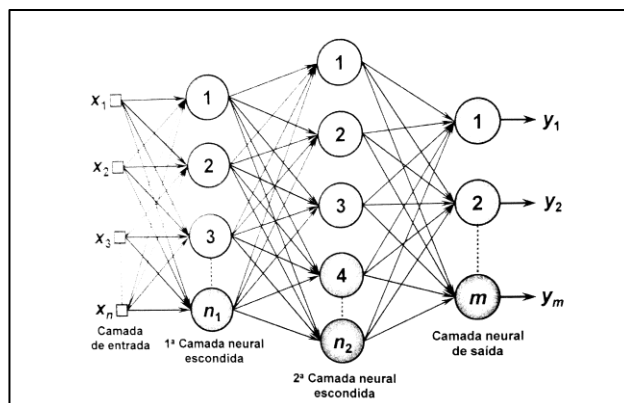


Figura 1. Modelo de Rede Neural Artificial.

Para que uma Rede Neural Artificial possa ser treinada existe um algoritmo específico para esta finalidade. São programas tradicionais escritos usando paradigma estruturado denominado algoritmo de aprendizagem. Tais algoritmos podem ser ou não supervisionados (Rezende, 2005). A definição de qual tipo utilizar depende do tipo de problema abordado. Caso seja problema de classificação ou previsão, o algoritmo de aprendizagem será o supervisionado. Para estes casos, ao informar o padrão, deve ser informado também a resposta correspondente ao padrão apresentado. Por outro lado, se for um problema de categorização, o algoritmo usado deve ser não supervisionado. Afinal, o objetivo é gerar classes, logo, a RNA irá receber os padrões de entrada e irá agrupá-los segundo a similaridade (Samarasinghe, 2007).

O algoritmo clássico das RNAs é o Backpropagation, porém, existem outros tipos e, inclusive, para certos problemas podem apresentar resultados considerados melhores do que o Backpropagation, pois, evitam cair em mínimos locais. Um exemplo é o PSO (Particle Swarm Optimization). Na próxima seção este algoritmo será abordado brevemente.

1.2 Algoritmo PSO

O algoritmo PSO (Particle Swarm Optimization) foi desenvolvido por Kennedy e Eberhart (1995). É um algoritmo flexível modelado segundo o comportamento de grupos, como pássaros, enxame de abelhas, nuvem de partículas e cardume de peixes.

A característica principal do PSO é imitar o comportamento social dos pássaros e isto é obtido por meio da modelagem da posição de cada partícula por meio de um vetor que contém os valores das velocidades. O processo é realizado em duas partes onde a primeira representa o próprio conhecimento da partícula e o segundo o conhecimento do enxame sobre o espaço de busca. Deste modo, o vetor velocidade é utilizado para atualizar a posição de cada partícula no enxame.

A programação do PSO é simples, pois, o algoritmo possui poucas etapas. Abaixo os passos para a programação do PSO:

- 1) Geração de partículas com posições e velocidades aleatórias;
- 2) Definição da função objetivo;
- 3) Ajuste do $pbest$ de cada partícula, isso é realizado fazendo comparações entre as melhores posições encontradas por elas.
- 4) Ajuste do $gbest$, da mesma forma que o procedimento anterior. É realizada comparação entre as melhores posições do grupo. Trata-se da melhor informação entre os indivíduos do grupo;
- 5) Atualizar a velocidade a partir da Equação 1:

$$x_{k+1}^i = x_k^i + V_{k+1}^i \Delta t \quad (1)$$

onde: x_{k+1}^i é a posição de cada partícula i na iteração $k+1$

v_{k+1}^i : é o vetor de velocidade desta partícula

Δt : equivale ao intervalo de tempo considerado.

O vetor velocidade é atualizado da seguinte forma:

$$v_{k+1}^i = w v_k^i + c_1 r_1 \left(\frac{p^i - x_k^i}{\Delta t} \right) + c_2 r_2 \left(\frac{p_k^s - x_k^i}{\Delta t} \right) \quad (2)$$

Neste trabalho os tipos indicados na Figura 2 que correspondem aos valores 1, 2 e 3 referem-se à porcentagem do núcleo analisado. Valores de enriquecimento:

1. 1.9%
2. 2.5%
3. 3.2%

Como saída, o atributo utilizado foi a posição do elemento combustível que, neste trabalho, correspondeu às posições de 1 a 31. Cada elemento combustível poderia ser posicionado em qualquer das 31 posições possíveis. Assim, a partir das 5 variáveis correspondentes a um elemento combustível, a RNA deveria indicar qual posição seria adequada para ele, levando em conta a sensibilidade causada no sistema. Ao mover qualquer um deles, não só o pico de potência do mesmo será afetado, bem como de todos os seus vizinhos. Trata-se de um problema de otimização espacial do núcleo de um reator PWR.

O modelo foi desenvolvido em Matlab R2015a. Foram testados RNAs com 150, 250 e 500 neurônios. Além disso, foram exploradas 1 e 2 camadas ocultas, bem como variadas taxas de aprendizagem e entropia cruzada para análise dos erros. Para treinamento do modelo Elman o algoritmo de treinamento escolhido foi o PSO. A escolha do modelo de RNA Elman se deu devido ao bom desempenho, além de não ter sido encontrado na literatura utilização desta RNA para este tipo de problema.

Para os dois modelos, o conjunto total de dados contendo 2.976 registros foi dividido em três partes sendo 75% para treinamento, 15% para validações e 10% para testes. Em todos os testes, esta porcentagem foi mantida. Os dados de entrada foram normalizados para auxiliar a rede neural artificial, conforme Equação 3.

$$x_{Ti} = \frac{x_i - x_{imin}}{x_{imax} - x_{imin}} \quad (3)$$

onde x_{imin} e x_{imax} são os valores mínimos e máximos das variáveis de x_i .

A Tabela 1 mostra exemplo de entrada da RNA.

Tabela 1. Características de entrada da RNA

Combustível	Ciclo	Queima	Pico de potência	Enriquecimento
1.00	0.20	-0.96	0.92	0.22
0.67	0.00	-0.84	1.00	0.43
0.33	0.00	-0.87	0.46	0.38

A saída da rede neural artificial foi composta pela posição do elemento combustível seguindo todas as regras apresentadas, principalmente de segurança. Como foi utilizado apenas $\frac{1}{4}$ de núcleo de um reator PWR, logo são 31 posições que deverão ser rearranjados a fim de obter melhor distribuição. As posições foram inseridas na RNA como dados binários. A Tabela 2 mostra exemplo de saída da RNA.

Tabela 2. Características de saída da RNA

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
.																															
.																															
31	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Como os modelos de RNAs utilizadas são supervisionadas, cada vetor de entrada correspondente às 5 variáveis foi associado a uma saída, existente na Tabela 2. Exemplo de vetor de entrada: [1.00, 0.20, -0.96, 0.92, 0.22].

Vetor de saída: [0,0]. Este exemplo indica que as características na linha do vetor correspondem a um elemento combustível localizado na quinta posição no núcleo do PWR.

Os dados do núcleo usados para treinamento eram compostos por conjuntos otimizados. Assim, era esperado que, após treinamento, a RNA pudesse receber uma coleção de dados de um núcleo sem otimização, ou seja, com vários picos de potência ao longo de sua superfície. Após processamento, pudesse apresentar uma distribuição mais plana possível, mantendo os níveis de segurança necessários aos reatores.

2. RESULTADOS

Como dito anteriormente, sugerir uma troca é algo complexo, pois isso altera não somente o pico de potência do próprio combustível, mas também de todos os seus vizinhos. Portanto, trata-se de um processo bastante sensível.

O modelo de RNA Elman gerado neste trabalho manteve 87% de elementos combustíveis na posição original. Os outros 13% a RNA sugeriu trocar de posição.

Na Figura 3 pode ser verificado a performance da RNA que convergiu na época 224, ou seja, devido às validações, após não obter resultados melhores que os anteriores, o algoritmo interrompeu o treinamento. Isso ocorreu por volta da época de número 218.

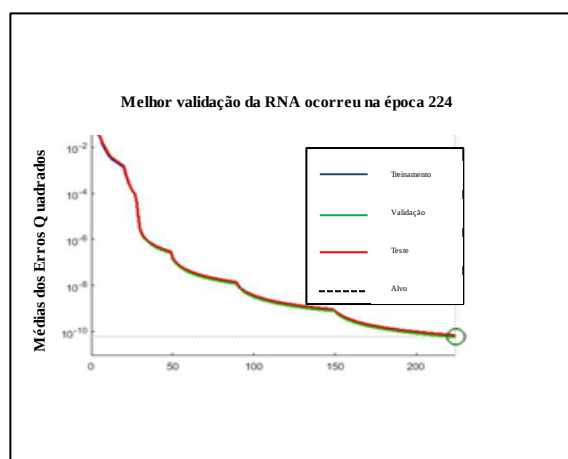


Figura 3. Performance da RNA durante os testes.

A Figura 4 mostra o gráfico onde pode ser visualizado o nivelamento original dos picos de potência. As regiões indicam as posições do núcleo do reator, sendo as regiões 1 e 2 com valores de potência maiores, porém, os elementos combustíveis foram inseridos no núcleo de forma alternada, conforme pode ser verificado na Figura 2. Desta forma, foi possível manter achatamento dos picos. Tais regiões são localizadas do meio em direção ao centro do reator e, portanto, apresentam maiores concentrações de calor.

A região 3 é posicionada na periferia do reator, ao longo das bordas. Por ser uma região de menor concentração de calor, é permitido posicionar elementos combustíveis com valores discrepantes, gerando picos de potência.

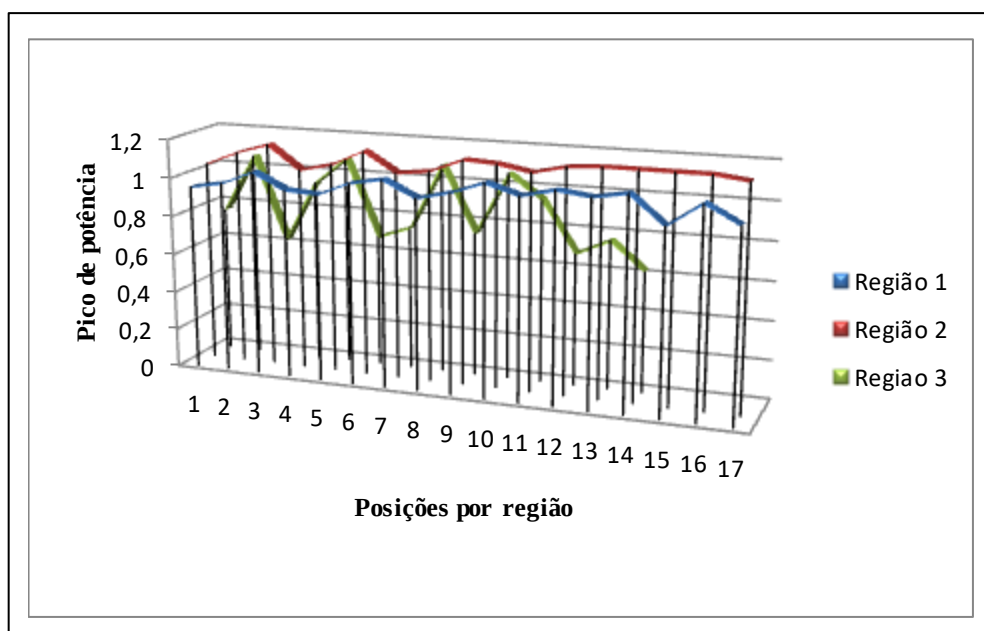


Figura 4. Gráfico de nivelamento dos picos de potência de $\frac{1}{4}$ do núcleo original.

Na Figura 5 percebe-se que os picos com maiores altitudes estão localizados na região 3 e a região 1 é a mais nivelada.

Isso indica que esta distribuição apresenta nivelamento adequado, pois os picos foram posicionados na periferia do núcleo. Nesta região, a potência é menor, ao passo que a região central concentra maior quantidade de calor (Cochran, 1999).

Contudo, esta distribuição pode melhorar, ou seja, os picos podem ser nivelados mesmo que de forma ligeira. A Figura 5 mostra o gráfico com a distribuição proposta pela RNA, lembrando que ela manteve 87% dos elementos combustíveis na mesma posição e sugeriu que 13% fossem trocados de lugar.

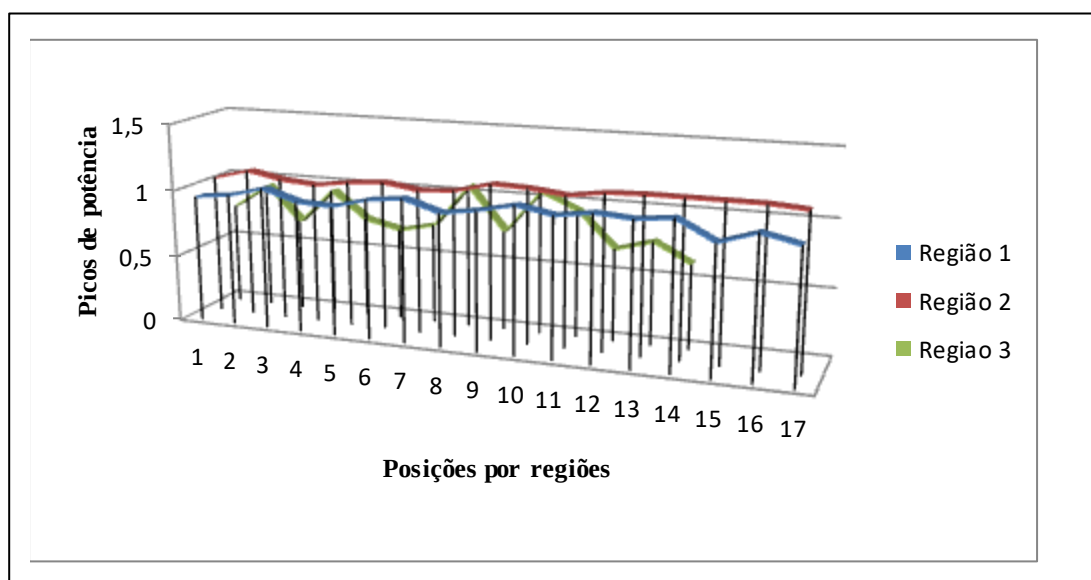


Figura 5. Gráfico de nivelamento dos picos de potência proposto pela RNA.

A distribuição espacial dos elementos combustíveis de $\frac{1}{4}$ do núcleo proposto pela RNA apresenta ligeira alteração. Houve rebaixamento de pico nas três regiões, sendo a mais acentuada na região 3. Alguns picos foram mantidos, inclusive a região dois permaneceu como o original, quase não houve propostas de mudanças. A região 3 é composta pelos elementos combustíveis posicionados na periferia do núcleo, logo, a potência é menor. De acordo com os valores analisados de $\frac{1}{4}$ de núcleo exibidos na Figura 2 as regiões 1 e 2 foram intercaladas em direção ao centro do reator. Com isso, estas duas regiões apresentam maior geração de calor.

De acordo com a segurança nuclear nas regiões 1 e 2 devem ser evitados picos de potência, afinal, quando concentrados geram grande quantidade de calor provocando perturbação no núcleo. Picos de potência dificulta a retirada de calor.

3. COMENTÁRIOS FINAIS

Este trabalho faz parte de um projeto que ainda não está finalizado. Serão realizados outros testes com diferentes redes neurais artificiais e outros algoritmos. Serão utilizados deep learning que é um sistema inteligente com capacidade considerada superior às RNAs.

Além disso, é importante que a proposta de distribuição realizada pela RNA seja validada por um código nuclear. Como dito ao longo deste trabalho, a simples troca de um elemento combustível altera todo o conjunto de assemblies e o núcleo do reator é todo afetado. É um processo complexo e sensível.

As redes neurais artificiais utilizadas neste trabalho indicaram que existe possibilidade de gerar um modelo inteligente capaz de propor distribuições espaciais adequadas de elementos combustíveis para um núcleo. O trabalho mostrou que o sistema possui capacidade de generalizar a partir dos exemplos apresentados.

Os exemplos utilizados foram estudados e propostos por engenheiros nucleares e validados por códigos nucleares, logo eram padrões adequados para serem utilizados em um reator nuclear PWR. A posição de cada elemento combustível trouxe consigo o tipo de combustível utilizado em sua fabricação, o ciclo de queima indicando maior ou menor potência individual, além do pico de potência que a estrutura de combustível proporcionava naquela posição com determinados vizinhos.

Analisando o gráfico, percebe-se que as trocas apresentadas pela RNA, ou seja, 13% mantiveram os picos de potência nivelados. Porém, isto não é suficiente sob o contexto de operação do reator. Como dito, trata-se de um processo complexo, intrincado com várias variáveis que devem ser levadas em considerações. Apenas um código nuclear poderá validar se a proposta da RNA foi melhor do que o projeto do núcleo original.

Em projetos futuros dando sequencia a este trabalho esta etapa será complementada.

AGRADECIMENTOS

Agradecemos à Universidade Federal de Minas Gerais pelo apoio a esta pesquisa.

REFERÊNCIAS

- Augusto, C. S. P. João, Nicolau, S. Andressa, Schirru, Roberto, 2015. *PSO with Dynamic Topology and Random Keys method applied to Nuclear Reactor Reload*, Brazil, Vol. 83, pp. 191-196.
- Cacuci, D.G, 2013. *Handbook of Nuclear Engineering*, Springer Verlag, Alemanha.
- Cochran, Robert G. Tsoulfanidis, Nicholas, 1999. *The Nuclear Fuel Cycle: Analysis and Management*, American Nuclear Society.
- Duderstadt, James and Louis J. Hamilton, 1976. *Nuclear Reactor Analysis*, Wiley.

- Faria, F. Eduardo, Pereira, Cláudia, 2003. “*Nuclear Fuel Loading Pattern Optimization Using A Neural Network*”, Brazil, Pergamon, Vol. 30, pp. 603-613.
- Haykin, Simon, 2009. *Neural Networks and Learning Machines*, 3ª edição, New Jersey, Prentice Hall.
- Oliveira, M. Ioná, Schirru, Roberto, 2011. *Swarm Intelligence of Artificial Bees Applied To In-Core Fuel Management Optimization*, Brazil, Vol. 38, pp. 1039-1045.
- Rezende, Solange Oliveira, 2005. *Sistemas Inteligentes – Fundamentos e Aplicações*, São Paulo, Manole.
- Russell, Stuart; Norvig, Peter, 2010. *Artificial Intelligence - A Modern Approach*, Third Edition, New Jersey, Prentice Hall Series em Inteligência Artificial.
- Samarasinghe, Sandhya, 2007. “*Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition*”, New Jersey, Auerback Publications.
- Silva, I.N., Spatti, D.H., Flauzino, R.A, 2016. *Redes Neurais Artificiais Para Engenharia e: Ciencias Aplicadas - Curso Pratico*, Basil.
- Souza, M. Rose, Moreira, João, 2006. “Neural network correlation for power peak factor estimation”, Brazil, Elsevier, Vol. 33, pp. 594-608.
- Poursalehi, N, Zolfaghari, Minuchehr, 2012. “*Multi-Objective Loading Pattern Enhancement Of PWR Based on the Discrete Firefly Algorithm*”, Teerã, Elsevier, Vol. 57, pp. 151-163.