PHYSOR 2002, Seoul, Korea, October 7-10, 2002

ON THE USE OF ARTIFICIAL NEURAL NETWORKS IN LOADING PATTERN OPTIMISATION OF ADVANCED GAS-COOLED REACTORS

A. K. Ziver, C. C. Pain, J. N. Carter, C. R. E. de Oliveira, A. J. H. Goddard
Imperial College of Science Technology and Medicine, Earth Science and Engineering,
Prince Consort Road, London, United Kingdom
a.k.ziver@ic.ac.uk
and
R. S. Overton
British Energy Generation, Barnett Way, Barnwood, Gloucester.

ABSTRACT

Artificial Neural Networks (ANNs) are applied to in-core fuel management optimisation of Advanced Gas-Cooled Reactors operated by British Energy in the United Kingdom to predict various parameters generated by the reactor core analysis code PANTHER. ANNs are biologically inspired computational models. Loading Pattern (LP) optimisation based on genetic algorithms (GAs) is being used in nuclear reactor fuel management studies, which demands substantial CPU times (at least order of weeks on a 866MHz single processor PC) for multi-cycle problems. This is due to the assessment and qualification of large number of LPs required to solve an optimisation problem with PANTHER. This paper reports on the use of ANNs in predicting core physics parameters to improve the speed to obtain optimal candidate LPs. The construction and training of a number of ANNs to accelerate the optimisation process are described with the aim of using these networks as surrogate models. The supervised learning method has been used to carry out network training. We used three-layered feedforward networks composed of one input, one hidden and one output layer with the backpropagation of error algorithm as the learning function. In addition the merits of using the scaled conjugate gradient learning algorithm has also been investigated. Several ways of using ANNs to accelerate optimisation are presented. Results have shown that ANNs recognise LPs that violate a radial power shape constraint and exclude (filter) those LPs from the search. We have demonstrated that ANNs can be used as accelerator algorithms within the GA. An attempt has been made to replace the PANTHER code with ANNs to perform a full multi-cycle optimisation for two AGR stations, which apply off-load and on-load refuelling. Results from these cases have shown orders of magnitude increases in speed.

1. INTRODUCTION

An optimiser, GAOPT [1] based on genetic algorithms (GAs) has been developed for the AGR in-core fuel management to predict loading patterns (LPs) cycle by cycle over a refuelling period. Fuel management of AGRs is very different when compared to other thermal reactors such as the PWRs. The main characteristics from the point of view of LP optimisation that are applied in AGR in-core fuel management in the United Kingdom are presented in references [2], [3] and will not be described in detail here. We have developed two main algorithms and implemented in the code GAOPT applied to two reactors, the first employing on-load refuelling (Reactor A) and the second applies off-load refuelling (Reactor B). Main features implemented in the optimiser can be summarised as:

- Based on a non-generational GA-based algorithm in which children created are independent from each other. This algorithm is suitable for parallel computations.
- Performs multi-cycle LP optimisation for user-defined (variable) batch sizes (number of fuel assemblies to be inserted) and allows shuffling of assemblies i.e. relocation of assemblies in different reactor channels.
- Applies station-specific safety and reactor physics constraints, which are listed in Table I. A detailed description of constraints is presented in reference [3].
- Generates the initial (start-up) population members randomly using a stochastic (Monte Carlo) algorithm.
- Searches through user-defined fuel assembly types (enrichments and burnable poison rings). There is no limit on number of assembly types to be included in the search. This is entirely left to the user.
- It is linked to the British Energy reactor analysis code PANTHER [4] to perform fuel management calculations. The AGR version of PANTHER is employed, which applies power shaping flux calculations as a function of burnup [5].
- Depending on the problem, GAOPT offers a number of single objective function optimisation to the user. Minimisation of the radial form factor (RFF a parameter similar to FΔh in water reactors), or maximisation of the average discharge irradiation can be performed. A station-specific function to maximise the profit over a planning period can be also used. RFF is defined as the ratio of the peak assembly power in the core to the average core assembly power, which normally should be less than the limit of 1.37.
- Allows shuffling of fuel from outer rings of the core to inner rings, which is the current operational practice being applied. The maximum number being controlled by the user.

Application of ANNs to in-core fuel management of PWRs has been performed by Kim et. al. [6], [7], [8] and Jang et. al. [9], [10]. In the former work ANNs are constructed to predict BOC parameters (assembly powers and k_{eff}) a few hundred times faster than the core analysis code. Kim and Lee [6] have developed parallel computing adaptive schemes using ANNs for PWRs. Jang et. al. have applied optimisation layer by layer (OLL) learning algorithm to predict assembly powers, burnup and critical boron concentration in PWRs. To date and to our knowledge no attempt has been made to use ANNs in AGR in-core fuel management. For AGRs, a unique reactor-dependent approach has to be considered in constructing the database for neural networks. We have developed and experimented with this database using PANTHER. A code has been written to construct input/output data to be fed into the layers of the network.

We have observed that running GAOPT as stand-alone (without linking it to ANNs) requires long CPU times especially for multi-cycle optimisations. This is shown in Table II for a number of test cases carried out for the two AGR stations. In these test cases LPs have been assessed using a 2D whole-reactor models, which were collapsed by performing a 3D condensation calculations using PANTHER. ANNs have been constructed and trained to predict parameters used in the optimisation to reduce computation times. In this paper, we describe development of

ANNs to AGR reloading optimisation and present the results of our research. The application of ANNs has been performed for two AGR stations referred as “Reactor A and Reactor B” to demonstrate advantages of their use in core management calculations. In the next section a brief description of ANNs is presented, followed by the techniques used to construct and train ANNs given in the section 3. In sections 4 and 5 we present results and conclusions respectively.

2. ARTIFICIAL NEURAL NETWORKS FOR IN-CORE FUEL MANAGEMENT OF AGRS

Artificial Neural Networks [11] (ANNs) can be described as computational models of the human (or mammalian) brain, which represent a branch of Artificial Intelligence. ANNs provide an adaptive control of a process by simulating functions of a brain. An ANN contains a layer of simple processing elements called input nodes or neurons, a layer of output nodes and a number of layers of hidden nodes. This whole system is referred as multi-layer perceptron (MLP). Information transfers from node to node like electrical signals pass from neuron to neuron in body’s nervous system. An important property of ANNs is the ability to learn. It is this property that makes ANNs attractive, which is used to tackle problems that other explicit algorithms take long time to solve. ANNs have provided successful algorithms (solutions) in many branches of science, engineering, medicine as well as finance, retail and logistics [12].

There are different types of ANNs, a most common type include the multi-layer perceptron (MLP), which is generally trained with the backpropagation of error algorithm. Some ANNs are classified as feedforward (signals flow from input to output layers) while others are recurrent (signals flow both directions between input and output layers) depending on how data is processed through the network. Another way of classifying ANN types is by their method of learning (or training), as some ANNs employ supervised training while others are referred to as unsupervised or self-organizing. During the training stage we used the backpropagation with momentum scheme in which the sum squared error E is minimised in a three-layered network (see Fig. 1). The formulation below outlines this scheme:

$$E = \sum_{p=1}^{P} E_p = \sum_{p=1}^{P} \sum_{j=1}^{m} \left(t_j^p - y_j^p \right)^2 \tag{1}$$

In (1), E is calculated for each pattern p, t is the target (supervised) output and y is the calculated output by the network given in Fig. 1.Adjusting the weights (W_{jk}) to minimise E is performed using the relationship:

$$\Delta W_{jk} = \eta \delta_j N_i \tag{2}$$

where in (2) η is the learning parameter and δ_i is given by one of the following relationships Depending on whether neuron j is a hidden (3) or an output neuron (4):

$$\delta_j = \frac{\partial f}{\partial net_j} = \left(\sum_m W_{mj} \delta_m \right) \tag{3}$$

$$\delta_j = \frac{\partial f}{\partial net_j} = \left(t_j^s - y_j \right) \tag{4}$$

In equation (4) t^s is the supervised (target) value for neuron j and net_j is the total weighted sum of input signals to neuron *j*. The equation (3) gives the evaluation of weights of hidden neurons (*m*), which are connected to the output of j. Equations (2) to (4) are solved iteratively beginning from the output layer by calculating the δ terms and weights for all connections in the network then minimising the error function of equation (1). We have used an accelerated backpropagation algorithm (BPM), which includes an added momentum term, β to equation (2), giving for iteration *r* as:

$$\Delta W_{jk}^{(r+1)} = \eta \delta_j N_i + \beta \Delta W_{jk}^{r} \tag{5}$$

To perform construction and training of ANNs, we have used the software package called Stuttgart Neural Network (SNNS) [13]. The networks created using the SNNS package have been linked to the AGR loading pattern optimiser using ancillary software developed specifically to perform certain tasks. A set of ANNs is constructed to predict the following parameters that are calculated using PANTHER. This is also shown in Fig. 2 as a block diagram.

- Maximum radial form factor in a reactor cycle.
- Irradiation profile at the end of cycle (EOC).
- Channel power profile at beginning of cycle (BOC).
- Total burnup days.
- Control Rod Balance Constraints.

We have used ANNs in LP optimisation in a number of ways:

- *ANN as a filter*: ANNs are robust classifiers as they are very good in pattern recognition. This property of ANNs is used to identify LPs that have unacceptable radial form factors (RFFs) in in-core fuel management of AGRs.

- *ANNs to create a random initial population for use in GAs*: Optimisation using genetic algorithms requires a starting (initial) population to be created. For large population sizes ANNs can be used to accelerate this process.

- *ANNs to duplicate PANTHER predictions (as surrogate models)*: The ultimate aim of our research is to use ANNs as surrogate models. For this reason we have developed ANNs to take over the functions of the reactor analysis code (PANTHER) so that all the data required by the optimiser can be predicted by a series of ANN software.

3. DEVELOPMENT OF ANNs

We have constructed ANNs that are suitable for two typical AGR cores. In the first type (Reactor A) no geometrical symmetry is applied and in the second (Reactor B) rotational symmetric loading is applied excepting the central 12 channels. Due to these differences and the size of the cores number of input neurons have been varied for each case. The scheme is illustrated in Fig. 3, which is also described below:

(1) *Create data-base for training*: Channel irradiations BOC and EOC and fresh assembly k_{∞}'s, RFF, channel powers (BOC), cycle length and control rod constraints are generated for each LP and each reactor, which constitute the database.

(2) *Construction of networks for training*: The three layers (input-hidden-output) are constructed for each network. The number of neurons in each layer has been changed to adapt the network for each reactor type and for the parameter to be predicted. For Reactor A, RFF prediction is carried out with a network of (616-200-1), giving number of neurons at each layer. For Reactor B, however this was reduced (due to rotational symmetry) to (180-100-1). This system has been used to construct ANNs for control rod constraints and cycle-length predictions. An algorithm is developed to predict whole-core irradiations (and channel powers) by employing only one eighth of the reactor core. It should be noted that a series of try and error networks are constructed in which the hidden neurons are varied until an acceptable network is created. In this process training time achieved to minimise the error (E) in equation (1) was used as a measure of the efficiency of a network.

(3) *Choosing a learning function*: The standard backpropagation with momentum (BPM) is used as the learning function in all cases. For the RFF prediction, the scaled conjugate gradient method (SCG) has also been investigated.

(4) *Train the network*: Supervised training has been carried out using the SNNS package for each parameter. It was necessary to perform training for at least 5000 patterns. For the case of RFF, it was necessary to increase this to about 10000 LPs to improve the accuracy of prediction.

(5) *Testing the networks*. After training, networks are created for each parameter and tested against the seen (database already included in the training set) and the unseen data (new database not included in the training set, untrained data). Performing this step verifies the ANN constructed and establishes whether further training is necessary. Figure 4 gives percentage errors obtained from testing 1000 LPs for RFF prediction after the ANN is trained using the BPM and SCG methods. It was found that SCG method has given better performance as a learning function with a penalty of longer training sessions. Figure 5 presents results obtained for the RFF prediction, which show the target RFF against those predicted by the ANN for 5000 LPs tested. It can be seen that all predictions lie within the ±10% error band.

(6) *Link (incorporate) the network into the optimiser*. When the accuracy achieved testing networks on the unseen data is acceptable, ANNs can be used within the optimiser for prediction of parameters. We have set the following average limits on the accuracy required from testing 1000 LPs that are not in the training set for the predicted parameters: RFF: 5-10%, Channel powers and irradiations: 10%.
(7) *Testing ANNs*: We have tested ANNs that we constructed within the optimiser on realistic refuelling optimisation problems and present our results in the following section.

4. RESULTS

Results from using ANNs in LP optimisation are presented in tables III to V and in figures 6 to 9, which show our findings from the three tests as described below:

- *TEST 1*

We present detailed results from using ANN as a RFF filter applied to Reactor B in tables III to IV and in figures 6 to 8. Table III gives total number of LPs generated with the corresponding CPU times from single cycle maximisation of mean discharge irradiation. We observed that using ANN as a filter has reduced the number of LPs that has been investigated by about a factor of 2, which is given in the three tests for which the population size have been taken as 5, 50 and 75. The speedup factors defined as the ratio of CPU times (without ANN over with ANN), which are shown in Fig. 6. The drop in speedup factors with increasing the population size has indicated that more training is needed, increasing the number of LPs in the training set has increased the speedup factors as can be seen in Fig. 6. Table IV presents the reactor physics related parameters that are calculated for the best LPs for the three population sizes (small and medium sizes) with and without ANN. Results listed in this table show that using ANN leads to acceptable solutions. In Fig.7 the profit function versus the average discharge irradiation has been plotted for the population size of 50 showing acceptable LPs that are generated using the ANN. Fig. 8 shows refuelling positions, which gives the best solution found, when using the ANN as a filter.

- *TEST 2 and TEST 3*

Table V presents speedup factors obtained using ANNs for Reactors A and B respectively. First run represents a single cycle and the second multi-cycle optimisation problem (a more CPU demanding problem). For the latter, a factor of 2000 acceleration has been obtained when ANNs are used to take over (or perform) functions of PANTHER. Fig. 9 presents channel power predictions shown for the quarter core model of an AGR using ANN as ratios of PANTHER values divided by ANN predictions. Maximum discrepancy found was 5% for this LP, which represent a LP with very high RFF (=2.9) and therefore rejected by the optimiser. For this reason higher than normal channel powers has been calculated by the ANN, which compared with PANTHER predictions very well. It is shown that LPs of these characteristics can be recognised by the ANN and excluded from the GA search.

5. CONCLUSIONS

We have developed and applied ANNs to LP recognition in order to accelerate multi-cycle in-core fuel optimisation for two AGR stations, which employ on-load and off-load refuelling. Our results have shown that benefits of using ANNs in the GA-based optimiser cannot be disregarded. Using ANNs as a filter or population generator has accelerated the optimisation and produce results (LPs) that are already tested (accurate) by PANTHER. But this cannot be said for the results obtained from surrogate models, as there would be some loss of accuracy in the final predictions from the surrogate models, which should be finally rechecked by PANTHER, depending on the how well ANNs are constructed and trained. We propose that an on-line training method should be used whereby surrogate models are trained and tested continuously during the optimisation process. By on-line method we mean training, optimisation and testing all three being carried out in parallel on multi-processor computers.

We have demonstrated that when ANNs are used to replace PANTHER, they provide speedup factors of the order of several thousand for multi-cycle optimisation problems. The loss of accuracy in the final predictions is inevitable and may be tolerated for problems demanding very high CPU times. In order to train ANNs a suitable database has to be created, which is strongly reactor/parameter dependent. This database will be linked to ANN software to perform on-line training in GAOPT for AGRs. We plan to investigate on-line training with a built-in error estimator to predict LPs using surrogate models only.

ACKNOWLEDGEMENT

Most of the support for this work was supplied by the British Energy, which owns and operates AGRs in the UK.

REFERENCES

1. A. K. Ziver, J. N. Carter, C. C. Pain, C. R. E. de. Oliveira, A. J. H. Goddard, "The PROGRAM GAOPT Version 3, A user's manual for AGR refuelling optimiser," Applied Modelling and Computation Group, *Imperial College Report*, October 2001.

2. A. K. Ziver, J. N. Carter, C. C. Pain, C. R. E. de. Oliveira, A. J. H. Goddard, and R. S. Overton, "Multi-Cycle Optimisation of Advanced Gas Cooled Reactor Loading Patterns using Genetic Algorithms," Paper submitted to *Nuclear Technology* for Publication (2002).

3. D. Barrable, J. H. Kershaw, R. S. Overton, K. Brearley and D. R. Gray, "Increasing fuel irradiations in advanced gas-cooled reactors in the UK," *Nuclear Energy*, **37**, p. 43 (1998).

4. P. Bryce, A. Goddard, M. Knight (editors), "PANTHER User's Guide for Release 5.1," Issue 1, *British Energy Report*, December (1998).

5. P. K. Hutt, N. Gaines, M. J. Halsall, M. McEllin and R. J. White, "The UK core performance code package," *Nuclear Energy*, 30, p 291, (1991).

6. H. C. Lee, H. J. Shim and C. H. Kim, "Parallel Computing Adaptive Simulated Annealing Scheme for Fuel Assembly Loading Pattern Optimisation in PWRS," *Nuclear Technology*, **135**, 39, (2000).

7. H. G. Kim, S. H. Chang and B. H. Lee, "Pressurised Water Reactor Core Parameter Prediction Using an Artificial Neural Network," *Nucl Sci. Engg*., **113**, 70, (1993).

8. H. G. Kim, S. H. Chang, and B. H. Lee, "Optimal Fuel Loading Pattern Design Using an Artificial Neural Network and Fuzzy Rule-Based System," *Nucl. Sci. Engg*., **115**, 152, (1993).

9. C. S. Jang and C. H. Kim, "Application of a Neural Network for Prediction of Two-dimensional Power Distribution in PWRs," *Trans. Am. Nucl. Soc*., **76**, 156, (1997).

10. C. S. Jang and C. H. Kim, "Application of a Neural Network for In-Core Fuel Management Optimisation Computation in PWRs," *Proc. Joint. Int. Conf. Mathematical Methods and Super Computing in Nuclear Applications, Saratoga, NY, USA*, **1**, p.782 (1997).

11. S. Haykin, " Neural Networks," *2nd Edition, Prentice Hall*, (1999).

12. P. D. Wasserman, "Neural Computing," *Van Nostrand Reinhold*, NY (1989).

13. A. Zell et. al. "SNNS Stuttgart Neural Network Simulator User Manual Version 4.1", *Report No 6/95, Institute for Parallel and Distributed High Performance System (IPVR)*, (1995).

14. S. A. Haddock, G. T. Parks, "AGR fuel management using PANTHER," *Proceedings of the international conference organised by the British Energy Society*, Edinburgh, 20-22 March (1995).

Table I. The Safety and Operational Constraints Applied by GAOPT

Description of Constraints	Limits
Minimum irradiation limit for refuelling	As high as possible
Minimum irradiation limit for shuffling	15GWd/tU
Maximum number of radial shuffles	4 (Normally)
Number of Fuel Assembly types to be inserted per cycle	Reactor Dependent (8 to 24)
Maximum radial form factor (RFF) allowed.	1.37
Control Rod Balance Constraints	Reactor Dependent For Reactor B mri≥0.6858 , qval≤1.25
Average control rod insertion limit to terminate the cycle EOC (burnup).	Reactor Dependent For Reactor B 0.26(*)

(*) A description of these limits can be found in reference [14]

Table II. Summary of Execution Times Obtained from Running Single and Multi Cycle Test Cases **Without Using ANNs**, giving Typical CPU times.

Single Cycle:

Case No	**Reactor (AGR)**	**LP's (+) Evaluated**	**LPs(-) Feasible**	**Reactor Cycles**	**Population Size**	**CPU Times (h) (866MHz PC)**
1	Reactor B	7787	50	1	5	10
2	Reactor B	9781	500	1	50	12
3	Reactor B	14904	500	1	75	19

Multi-Cycle:

Case No	**Reactor Type**	**LP's Evaluated(+)**	**LP's(-) Feasible**	**Reactor Cycles**	**Population Size**	**CPU Times (h) (866MHz PC)**
1	Reactor A	63898	40	4	50	88
2	Reactor B	75034	400	8	50	166

(+) Total number of LPs investigated by PANTHER

(-) LPs satisfy all constraints given in Table I.

Table III. Application of ANNs to Single Cycle Optimisation

Population Size/ Evaluations (+)	Using ANN as a filter LPs	CPU(s)	Without ANN LPs	CPU(s)
5/50	3918	17987	7787	57663
50/500	6669	32308	9781	47384
75/500	6595	30200	14904	68250

(+) Number of LPs that satisfy all the constraints

Table IV. Acceptable LPs Predicted by the Optimiser for Reactor A

Population Size	ANN as a Filter ?	RFF	Discharge Irradiation	Profit	Mri(+)	Qval(+)
5	Yes	1.3518	24.21	0.193344	0.7129	1.1095
5	No	1.3676	25.31	0.193730	0.7072	1.1615
50	Yes	1.3343	25.36	0.203050	0.7083	1.2348
50	No	1.3379	25.98	0.202311	0.7229	1.1403
75	**Yes**	**1.3557**	**26.11**	**0.199306**	**0.6999**	**1.1383**
75	**No**	**1.3758**	**26.07**	**0.207411**	**0.7533**	**1.1213**

(+) These are control rod balance constraints mri is the mean rod insertion and qval is defined below.

$$qval = \frac{max(mri)_j}{min(mri)_j}$$

Where, j represents grey rods in each quadrant ($j = 1,2,3,4$).

Here profit is normalised function taking into account the operational and fuel cost requirements for each reactor.

Table V. Comparison of CPU Times (s) to Show the Use of ANNs in Fuel Management Optimisation of AGRs. Figures in Brackets are the Speedup Factors

Run No	Reactor (AGR)	Without ANNs(*)	ANNs as a Filter	ANNs as Surrogate Models	ANNs as a Restart
1	Reactor A	706 [0.0]	550 [1.28]	45 [15.69]	120 [5.88]
2	Reactor B	101749 [0.0]	2687 [37.87]	49 [2076.5]	12225 [8.3]

(*) This is the base case (Note all cases have been run on 866 MHz PC).

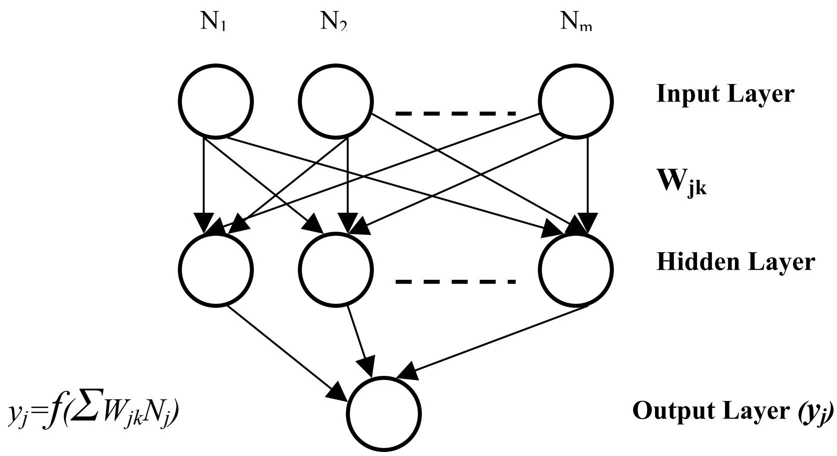

Fig. 1. The three-layered feedforward multi-layer perceptron. Here f is the sigmoidal threshold function [12] and [13].

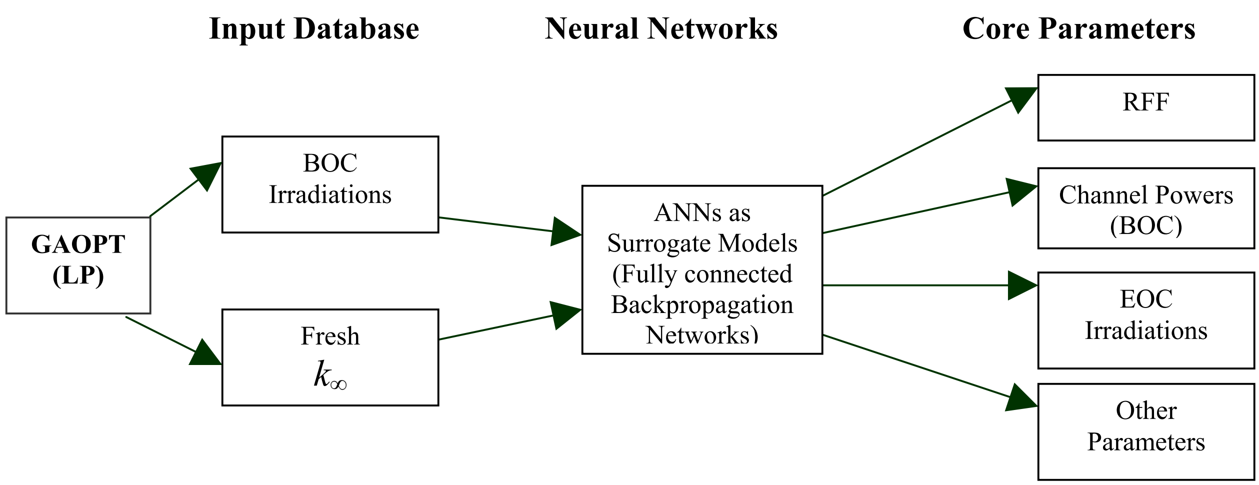

Fig. 2. Database Used and Core Parameters Predicted using ANNs for a single Loading Pattern generated by the Optimiser (GAOPT). (Note that BOC=Beginning of Cycle and EOC =End of Cycle.)

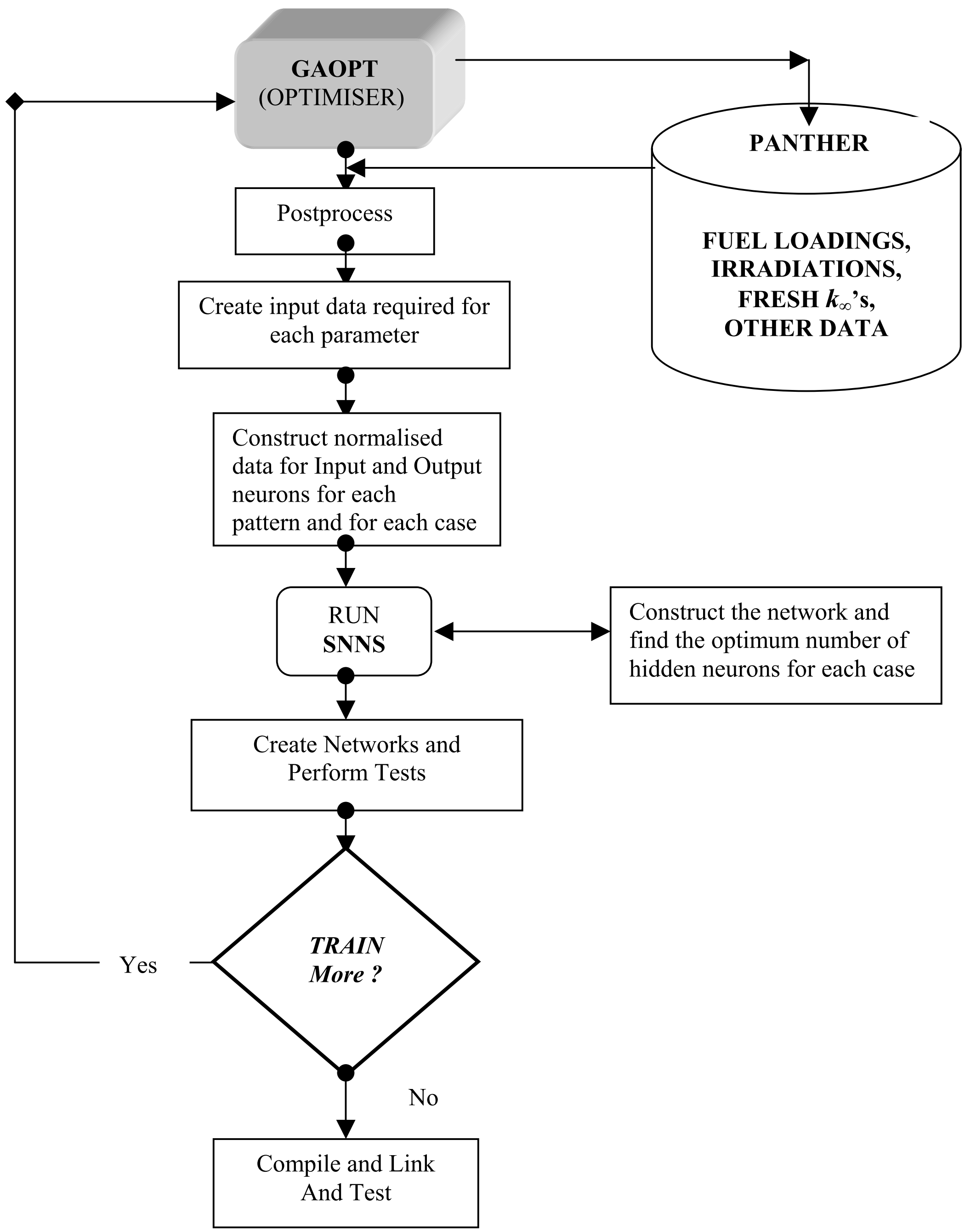

Fig. 3. Construction and training of Artificial Neural Networks

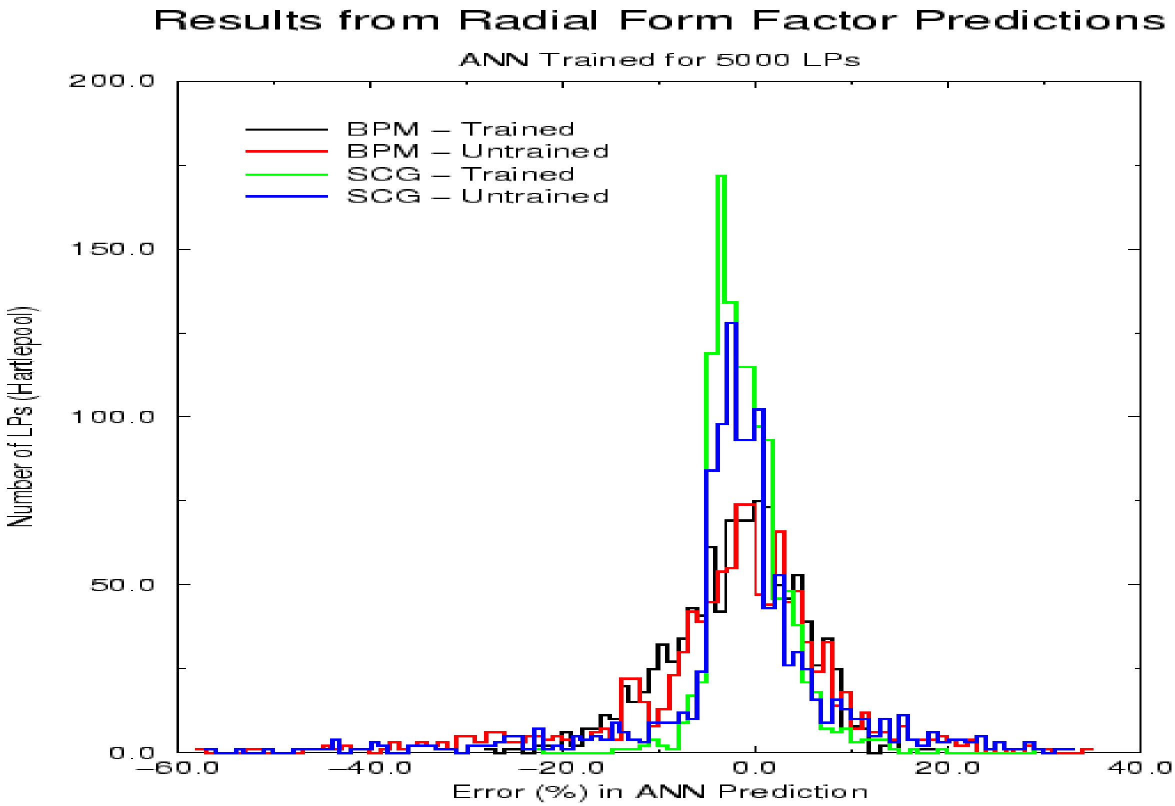

Fig. 4. Results from testing the ANNs for RFF prediction when two training functions, backpropagation with momentum (BPM) and scaled conjugate gradient (SCG) are used to test 1000 LPs trained (seen data) and untrained (unseen data).

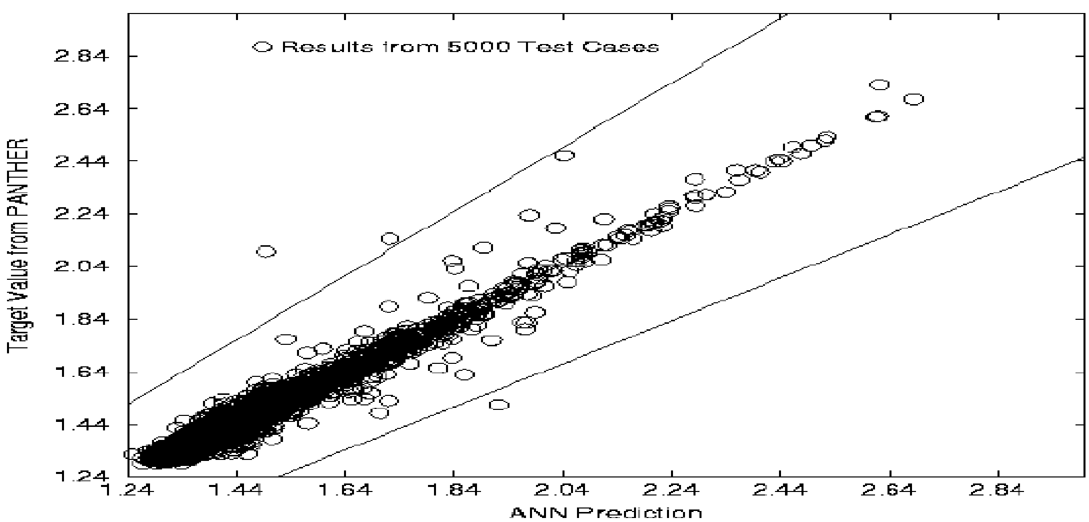

Fig. 5. ANN predictions vs the target value for RFF showing the predictions to be within ± 10% band.

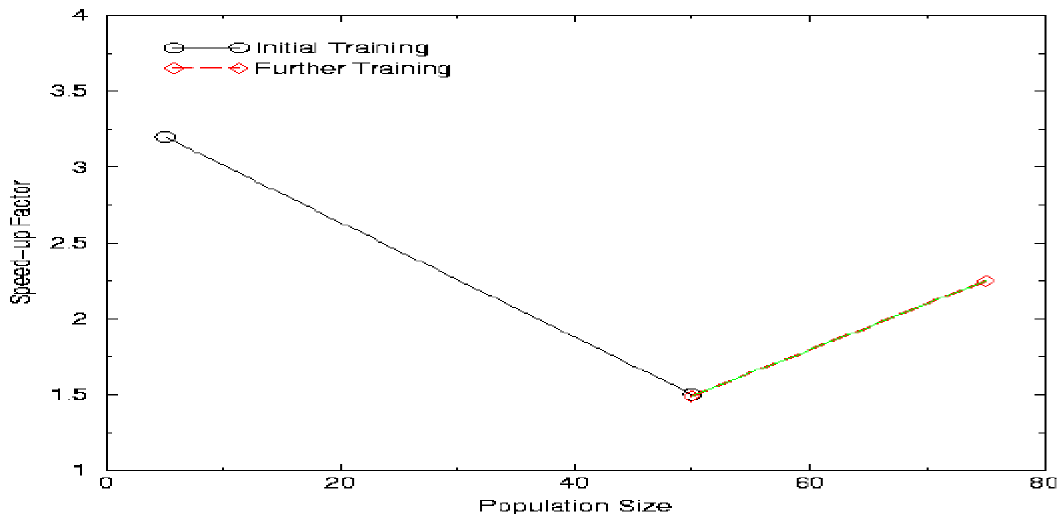

Fig. 6. Using ANN as a filter showing speedup factors obtained against population size. Initial training was performed for 2000 patterns, which was further increased to 5000.

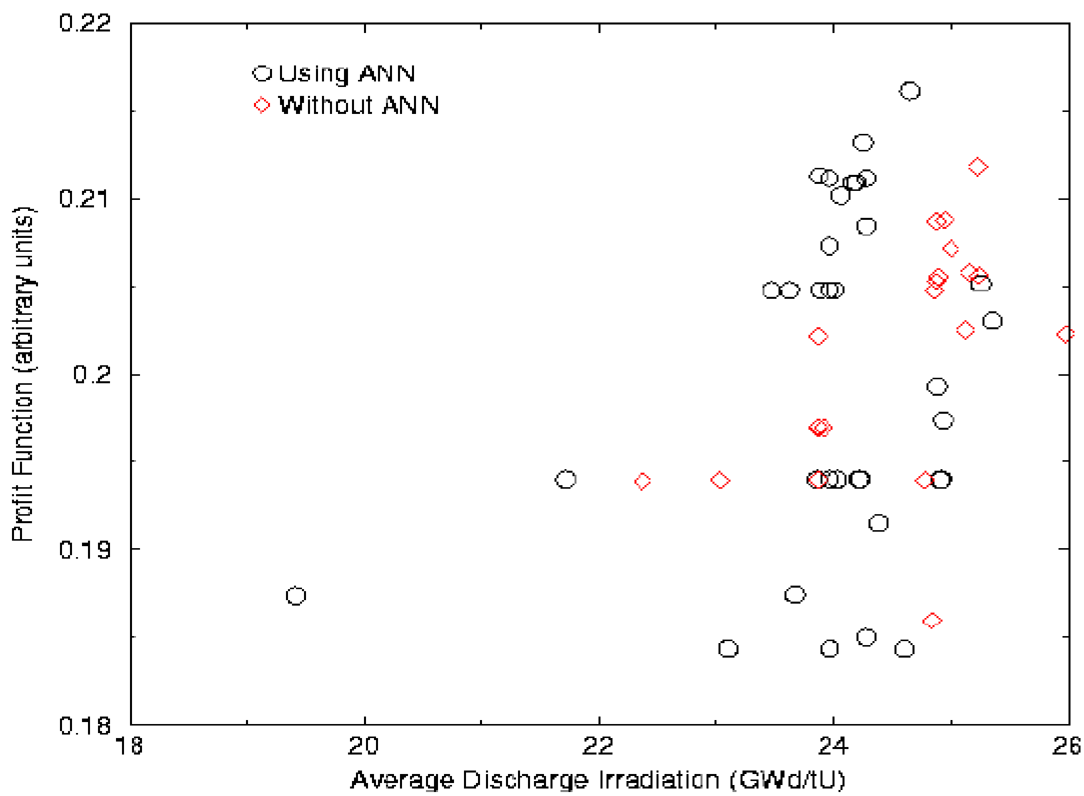

Fig. 7. Maximised Average Discharge Irradiation plotted against the profit function representing the final population in the GA after 6669 LP evaluations using ANN and 9781 LP investigations without ANN.

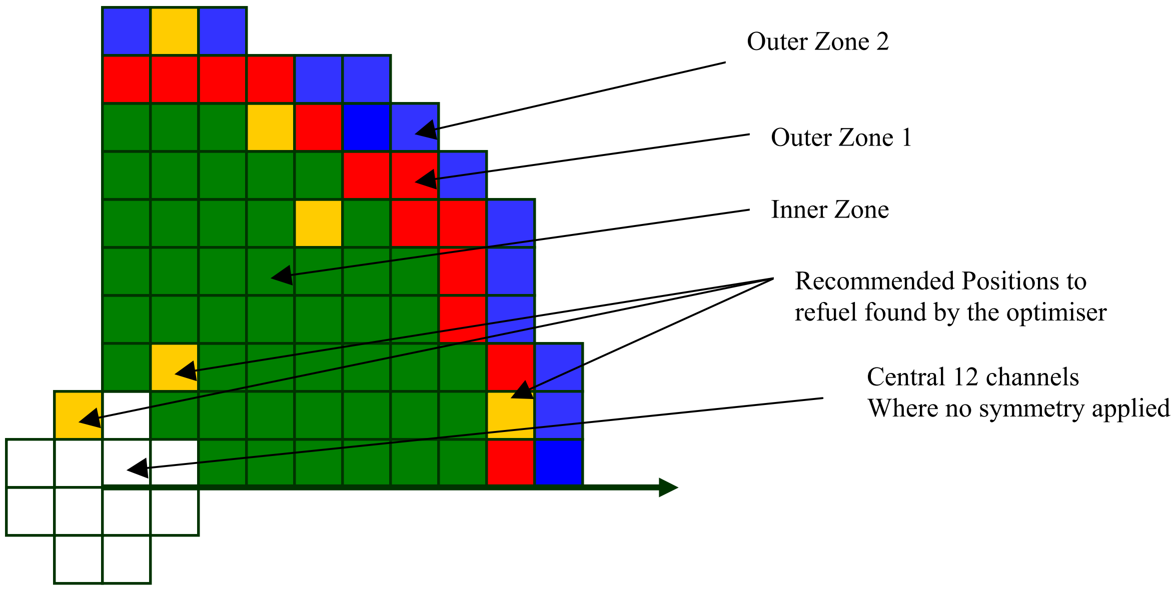

Fig. 8. Refuelling of LPs Predicted for a Single Cycle Maximisation of Discharge Irradiation when using ANN as a filter. Optimisation used 3 types of fuel assembly one for each zone.

4.96 (1.0)	4.42 (1.0)	6.00 (1.0)							
5.20 (1.0)	6.81 (1.0)	6.34 (1.0)	7.04 (1.0)	8.08 (1.0)	6.85 (1.0)				
4.50 (1.0)	5.09 (1.03)	5.00 (1.05)	6.18 (1.02)	9.09 (1.0)	7.60 (1.0)	7.92 (1.0)			
4.74 (1.0)	6.68 (1.0)	6.63 (1.01)	10.3 (1.0)	9.05 (1.0)	11.7 (1.0)	7.68 (1.0)	7.89 (1.0)		
5.55 (0.99)	5.76 (1.0)	8.37 (1.0)	11.1 (1.0)	13.4 (1.0)	11.5 (1.0)	10.7 (1.01	11.4 (1.0)	8.90 (1.0)	
6.35 (1.0)	6.49 (0.99)	6.30 (1.01)	10.7 (0.99)	12.7 (1.0)	13.1 (1.0)	11.0 (1.0)	7.52 (1.0)	8.33 (1.0)	
7.19 (0.98)	6.90 (1.02)	7.69 (0.99)	7.44 (0.98)	10.0 (1.04)	9.58 (1.0)	11.5 (0.99	12.9 (1.0)	10.6 (1.01)	
6.01 (0.96)	5.89 (0.99)	8.55 (1.0)	8.34 (1.04)	6.73 (1.0)	8.44 (1.01)	8.56 (1.0)	8.95 (1.0)	9.80 (1.0)	9.34 (1.0)
7.76 (1.00)	6.54 (1.0)	9.36 (1.0)	6.88 (0.97)	9.67 (0.99)	6.60 (1.0)	8.53 1.0	6.96 (0.98)	6.26 1.0	8.23 (1.0)
7.32 (1.01)	7.24 (0.99)	8.03 (1.0)	8.46 (1.02)	8.18 (1.0)	5.68 (1.02)	5.55 1.01	5.41 (1.0)	5.91 (1.0)	6.07 (1.0)

Key: MW (Ratio)

Ratio=PANTHER/ANN

Fig. 9. A Comparison of ANN Channel Power Predictions against PANTHER (Quarter Core shown). Note that LP for this case is rejected due to RFF violation.