Research Reactor Loading Pattern Optimization Using Estimation of Distribution Algorithms

Shan Jiang[1], Kemal Ziver[1*], Jonathan N. Carter[1], Christopher C. Pain[1], Matthew D.Eaton[1], Antony J. H. Goddard[1], Simon J. Franklin[2] and Heather J. Phillips[2]

[1] *Department of Earth Science and Engineering, Applied Modeling and Computation Group (AMCG), Imperial College London, SW7 2AZ, United Kingdom*
[2]*Imperial College Reactor Centre, Silwood Park, Buckhurst Road, Ascot, Berkshire, SL5 7TE, United Kingdom*

Abstract

A new evolutionary search based approach for solving the nuclear reactor loading pattern optimization problems is presented based on the Estimation of Distribution Algorithms. The optimization technique developed is then applied to the maximization of the effective multiplication factor (K_{eff}) of the Imperial College CONSORT research reactor (the last remaining civilian research reactor in the United Kingdom). A new elitism-guided searching strategy has been developed and applied to improve the local convergence together with some problem-dependent information based on the 'stand-alone K_{eff}' with fuel coupling calculations. A comparison study between the EDAs and a Genetic Algorithm with Heuristic Tie Breaking Crossover operator has shown that the new algorithm is efficient and robust.

KEYWORDS: Estimation of Distribution Algorithms, Nuclear reactor loading pattern optimization, Stand-alone K_{eff} with fuel coupling, Imperial College CONSORT Reactor, Artificial Neural Networks

1. Introduction

The main goal of Nuclear Reactor Loading Pattern Optimization (NRLPO) problem in commercial nuclear reactors is to search for 'profitable' Loading Patterns (or LPs) of fuel assemblies which maximize the performance of the reactor subject to a number of safety and operational constraints. The optimization schemes depend on the type of reactor being modeled, operational and safety rules, fuel inventory, economic factors etc. At the Imperial College Reactor Center (ICRC), the main motivation is to extend the operable life time of the reactor (the CONSORT reactor) with the existing fuel inventory, available from the 1960s and 70s, to carry out research [1] and provide services for industry [2]. In this work, an optimization study has been carried out to maximize the excess reactivity of the CONSORT reactor core with the fuel inventory available at present (year 2006) using the Estimation of Distribution Algorithms (EDAs) [3]. An innovative application of heuristic information for NRLPO problems based on the stand-alone K_{eff} with fuel coupling has been developed and applied in the optimization study and the results are compared against the previously published Genetic Algorithm (GA) with Heuristic Tie Breaking Crossover (HTBX) operator [4].

[*]Affiliated to AMCG group, RM Consultants Abingdon, UK, kemal.ziver@rm-consultants.co.uk

A brief description of this study which maximizes the K_{eff} of the CONSORT reactor of ICRC is presented, as well as the calculation of problem-dependent heuristic information, the stand-alone K_{eff} with fuel coupling. A fast K_{eff} predictor using Artificial Neural Networks (ANNs) has been constructed and is presented. The use of ANNs has enabled the optimization to be carried out very rapidly (in terms of CPU time), replacing the EVENT [5] code used for core physics calculations, which also provided accurate predictions. The framework and implementation of the EDAs as applied to the K_{eff} maximization study are presented followed by the results and conclusions.

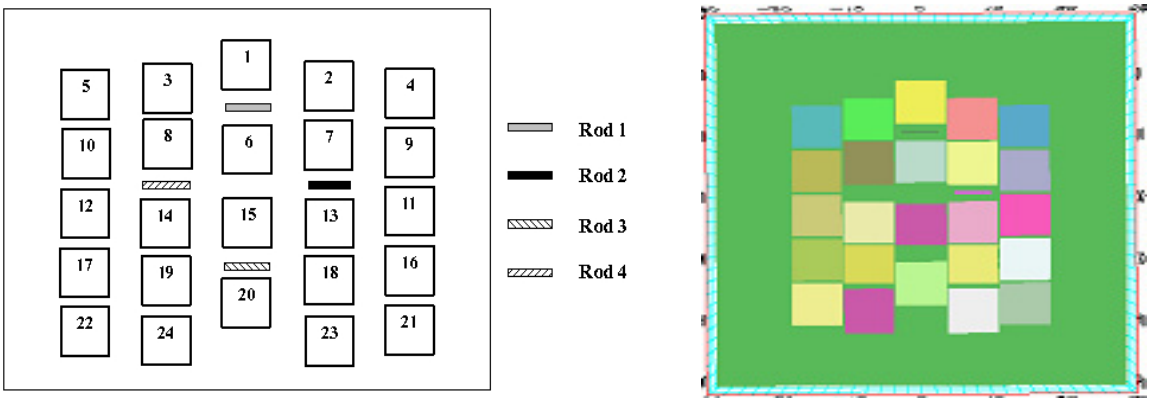
2. A Test Case: Imperial College CONSORT Research Reactor

2.1 Core Plan and Fuel Store

In order to demonstrate the application of EDAs to NRLPO problems, a test case was set up to maximize the K_{eff} of the CONSORT reactor given the store of fuel elements available at the ICRC. A new core state representing the reactivates of the fuel elements approximately extrapolated to present time (2006), has been constructed based on the fissile mass estimations. The reactor physics code WIMS8A[6] was used to obtain the multi-group constants which are then used in the 3D homogenized EVENT computational model. A hypothetical core state was previously studied to test the optimization algorithms developed which can be found in [7]. For detailed reactor information, contact reactor.centre@imperial.ac.uk.

The core plan showing the EVENT model and fuel channel numbering of the CONSORT reactor is shown in Fig.1. There are 24 fuel channels in the core. Each box in Fig.1 (left) represents a fuel channel, indexed by the number inside. There are four control rods, rod no.1, 2, 3, and 4, in the core as shown in Fig.1 (left). Note that the rod no.3 and no. 4 are not included in the EVENT model because they act as shut down rods to ensure the emergency safety. Meanwhile, rod no. 1 and 2 are fully inserted in EVENT model.

Figure 1: A plan view of the Imperial College CONSORT reactor showing the element positions (left) and the computer model developed using EVENT (right).



The fuel inventory consists of 35 fuel elements in this study. The fuel elements are ranked by their infinite multiplication factors, K_∞, and the ranking number is then used as their IDs in the algorithm, see Table 1. An important hard constraint is that Fuel Element10 and Fuel Element.11 (see Table 1) must always be inserted into channel 6 and channel 15, shown in Fig. 1, due to safety and operational constraints. Given the core plan, the fuel store and the constraints, the size of the search space can be evaluated, and approximately is 2.2×10^{29}. It should be noted that the fuel

elements orientation is not included in the EVENT model for K_{eff} calculation and therefore is eliminated from this optimization study.

2.2 The stand-alone K_{eff} with fuel coupling

The K_∞ is a very important reactor physics parameter in ranking the fuel elements in terms of their reactivates for fuel management optimization calculations. However, K_∞ does not give any spatial information about where a fuel element should be inserted into the reactor core. For optimization purposes, the 'coupling' of a fuel element at different positions in the reactor core with other fuel elements is very useful and can be used as heuristic information. A new method is proposed to address the spatial effects based on the calculation of the stand-alone K_{eff} with fuel coupling. The method involves insertion of a fuel element j in channel i, and filling all other channels with a 'generic' fuel m. Performing this for each fuel element, an LP is then created and examined by the simulation software EVENT to obtain its K_{eff}. This result is recorded as the stand-alone K_{eff} with fuel coupling information for fuel j when it is inserted in channel i. This calculation can be repeated for all the fuel elements and all the channels, the results can be presented in a 24x35 matrix. Each entry [i, j] of this matrix represents the 'spatial contribution' of assigning fuel j to channel i in a more 'realistic' context, compared to using K_∞ information only. The use of the stand-alone K_{eff} with fuel coupling in optimization will be described in the following sections.

Table 1: The K_∞'s calculated using WIMS for the fuel elements present in the store.

K_∞	1.5222	1.5249	1.5418	1.5443	1.5257	1.5159	1.5313
Rank	22	20	13	12	18	31	15
K_∞	1.5285	1.5259	1.5329	1.5183	1.5175	1.5162	1.5167
Rank	16	17	14	23	25	30	28
K_∞	1.5151	1.5167	1.5131	1.5169	1.5176	1.5175	1.5247
Rank	32	29	33	27	24	26	21
K_∞	1.5251	1.1312	0.7741	1.6038	1.6085	1.6666	1.6651
Rank	19	34	35	11	10	6	9
K_∞	1.6660	1.6653	1.6786	1.6785	1.6738	1.6756	1.6777
Rank	7	8	1	2	5	4	3

2.3 An ANN for fast K_{eff} evaluation

A three-layer feed forward ANN is used to provide fast K_{eff} predictions which was then used to mimic EVENT results (K_{eff}s). Each EVENT 3D calculations in six-energy groups took about 20 minutes CPU time, but this step was performed using the ANN in a fraction of a second. ANNs can be used provided that the network is properly trained and tested which of course require additional analyst's effort [9]. The ANN simulator used here has 22 input nodes representing the 22 fuel channels, since the Channel 6 and Channel 15 have to be filled only with Fuel Element 10 and 11, respectively. The input nodes are fully connected with 44 hidden nodes. There is 1 output node for the predicted K_{eff}. The number of hidden nodes was chosen by the 'trial and error' experiments [8]. Stuttgart Neural Network Simulator (SNNS) [8] is used for constructing and training the proposed ANN.

A total of 2025 different LPs are generated randomly and their K_{eff}'s are calculated by EVENT. A subset of 1620 of them was used for training, 202 for validation and 203 for testing of the ANN.

The input data for the ANN used for training the network are not the fuel elements ID but an exponential rescaling of their corresponding stand-alone K_{eff} with fuel coupling. For example, an LP representing the channels 1 to 3:

$$X = [15,7,9] \quad (1)$$

which means fuel channel 1, 2, and 3 are loaded with Fuel Element 15, 7, and 9, respectively. From the calculated stand-alone K_{eff} table we have that K_{eff} [15,1] = 1.00971, K_{eff} [7,2] = 0.99532 and K_{eff} [9,3] = 1.00870, applying an exponential scalar (for example, 20), the input is transformed to:

$$X' = [1.00971^{20}, 0.99532^{20}, 1.00870^{20}] = [1.21320, 0.91045, 1.18920] \quad (2)$$

Doing this, the variance is rescaled and this helps the ANN to recognize different LPs during the training. The input vector X' is then normalized between [0,1], and the corresponding K_{eff} is normalized between [0.3,0.7]. Using the trained ANN, the K_{eff}s of 1000 LPs can be predicted within a second with estimated errors of 0.128% of EVENT calculations on average. The same number of LP evaluations by EVENT take a CPU week on the same machine (Intel Xeon 2.8 GHz). The results from the testing of the ANN are given in Table 2.

Table 2: The results from the testing of the ANN to predict K_{eff}. The Error is represented by the absolute error between ANN and EVENT over EVENT calculation.

LPs Set	Total No. LPs	Error< 0.1%	Error<0.5%	Error<1%	Average Error
Training Set	1620	1535	1611	1620	0.13%
Unseen Test Set	203	194	203	203	0.13%

3. Application of EDAs to Reactor Loading Pattern Optimization

3.1 An Introduction to EDAs

3.1.1 Problem Representation

EDAs are a class of algorithms, which can be regarded as a subset of the Evolutionary Algorithms (EAs). In EAs, a dynamically changing population of solutions is maintained during the search. GAs are also a subset of EAs. In standard GAs, partial solutions are extracted from some known solutions and recombined using the crossover and mutation operators. In EDAs, a probability distribution model of 'promising solutions' is sampled to generate new solutions. This model is then dynamically updated by the newly sampled solutions.

Some of the key points in EDAs are the representation of the solutions, the probability model and the methods to update and sample this model. A straightforward method of LP representation is using the Permutation Representation (PR). The PR is an integer vector which contains a permutation from 1 to *n*, indicating the assignment of fuel elements to positions. This is used in the benchmark GA implemented for the comparison study. The complexity of dealing with PR is *O(n)*. Note that, the *n* positions include the in-core fuel channels and the positions to store the out-of-core fuel elements so that the full search space can be explored.

An alternative representation is a binary matrix encoding, which is used in our EDAs. An LP is represented by a binary matrix with 35 rows and 35 columns. Each row represents an in-core fuel channel or an out-of-core position to store a fuel element which is not presented in this LP. Each column of this matrix represents 1 of the total 35 different fuel elements. The entry [*i, j*] is set to 1 if and only if fuel channel *i* is loaded with fuel element *j*. Otherwise it is equal to 0. Note that

because one channel can only be occupied by one fuel assembly, each row, and each column can only have one '1' element at a time.

Given the binary matrix representation, EDAs complexity has to be up to $O(n^2)$. However, for NRLPO problems, the reactor simulation for evaluating LPs is so expensive in term of computational time that the cost of using complicated searching algorithms can be ignored.

3.1.2 Probability Model of Promising Solutions

EDAs explore the search space systematically by introducing a probability distribution probability model, which records the estimated distribution of promising solutions, and therefore is used to sample more solutions.

The structure of the probability model is identical to the LP representation, which is a 35 by 35 matrix, but it contains real-value numbers between [0,1]. The entry [*i,j*] represents the probability of loading fuel element *j* to channel *i* for a promising (or acceptable) LP.

Using a distribution model one can easily control and guide the search process by looking into the model and perturbing it if necessary. The method to sample and update the probability model will be explained in sections 3.1.3 and 3.1.4, respectively.

3.1.3 Generating New Solutions by Sampling a Probability Model

To illustrate how to generate an LP from a distribution model we consider an example with four fuel elements. First, we randomly choose a fuel channel, and take the corresponding row.

$$P[i] = [0.1, 0.2, 0.3, 0.4] \tag{3}$$

Second, we calculate the corresponding cumulative vector, and generate a random number r in [0, 1] with uniform distribution (e.g. $r = 0.5$)

$$P[i]_c = [0.1, 0.3, 0.6, 1], \quad r = 0.5 \tag{4}$$

Finally, r is compared to each entry of $P[i]_c$ until $r<P[i,j]_c$. The resulting j is chosen as the fuel element ID to be loaded in channel i.

$$LP[i] = [0, 0, 1, 0] \tag{5}$$

This process is repeated until all the channels are properly loaded. Because each fuel element can only be used once, a candidate fuel elements list can be used in this procedure to ensure the validity of the generated LP.

3.1.4 Updating the Probability Model

In order to find more promising solutions, the probability model should be updated using the previously built model and some good individuals (solutions) in the current population. Let us assume there are three good LPs, *LP1*, *LP2* and *LP3*, and the previously built model $P^{(t)}$. We can update $P^{(t)}$ to $P^{(t+1)}$ channel by channel. First, channel i is chosen randomly, and $X[i]$ is the sum of $LP1[i]$, $LP2[i]$ and $LP3[i]$.

$$LP1[i] = [0, 0, 1, 0]; \quad LP2[i] = [0, 1, 0, 0]; \quad LP3[i] = [0, 0, 1, 0]; \quad X[i] = [0, 1, 2, 0] \tag{6}$$

Second, $X[i]$ is normalized so that all entries are in the range [0, 1] and sum to 1.

$$X[i]_n = [0, 0.33, 0.67, 0] \tag{7}$$

Finally, $P^{(t)}[i]$ and $X[i]_n$ is combined together to form the updated model $P^{(t+1)}[i]$. In this work, we used the method from the Population Based Incremental Learning algorithm [3].

$$P^{(t+1)}[i] = (1-\alpha)\cdot P^{(t)}[i] + \alpha \cdot X[i]_n \tag{8}$$

In which α is a scalar between [0, 1] . This process is repeated for each row of $P^{(t)}$ to generate a fully updated $P^{(t+1)}$.

3.1.5 The Algorithm

The main steps in a EDA are:

1. Initialize the probability distribution model with uniform distribution.
2. Sample the probability distribution model using the method described in section 3.1.3 to generate LPs population. A swap mutation operator [10] is then applied to the new population.
3. Select some LPs according to their K_{eff} using a 2-person tournament selection method.
4. Update the probability model using the selected individuals from the previous probability model, as described in section 3.1.4.
5. If maximum number of LP evaluations (100,000, for example) is not met, go back to 2, otherwise end the search.

3.2 EDA_G: A General EDA with a New Elitism Strategy

Genetic Algorithms perform very well in locating a population which contains promising solutions but often fail to find the local optimum. This is also a common problem of many other population-based algorithms. An EDA with a modified probability model, EDA_G, has been developed to improve the local convergence.

$$P^{(t+1)} = (1-\alpha)\cdot P^{(t)} + \alpha \cdot X_n + \eta \cdot X_b \tag{9}$$

In equation (9), X_b represents the best solution found during the search and η is a scalar dimensionless factor. This term perturbs the probability model in the direction of the best known solution. It drives the whole population to move towards an area of search space where the current best is at its centre. If this is not applied the search will be performed only in the local area of the current best. The value of η has to be tuned carefully so that the search is not trapped in local optima. A random perturbation operator, such as a conventional mutation, has been suggested to be applied [10] as it can also help the global search. The values of the parameters used in equation (9) are presented in Table 3, these are derived from numerical experiments carried out for the present study.

3.3 The EDA_H: The EDA_G Algorithm Combined with Heuristics

In order to obtain better solutions and faster convergence, the EDA_G is combined with the stand-alone K_{eff} with fuel coupling data produced earlier. The resulting algorithm is referred to as EDA_H. The EDA_H algorithm is identical to EDA_G except that it samples the new population of LPs using the probability model, together with this heuristic information. The sampling probability is given by:

$$P'[i,j] = P[i,j]\cdot H[i,j]^{\beta} \tag{10}$$

Where in equation (10) H contains the heuristic information, β is an exponential scalar adjusting the weight between the population based learning and the heuristic information. H has identical structure to P. The entry $H[i, j]$ is the calculated stand-alone K_{eff} obtained from loading of fuel j to channel i with other channels filled with a generic fuel element as described in section 2.2 . A larger $H[i, j]$ value increases the probability of loading fuel j to channel i.

The inclusion of the heuristic information can be explained as follows. The probability of assigning a fuel element j to a fuel channel i depends on how often this assignment occurred in some known good LPs (positive), which is recorded in P, and its corresponding stand-alone K_{eff} with fuel coupling (positive). The stand-alone K_{eff} can be considered as a measurement of the 'contribution' of this assignment to the objective function, which is the maximization of K_{eff}. A larger $H[i, j]$ encourages the algorithm to generate more LPs with fuel j assigned to channel i.

The stand-alone K_{eff} with fuel coupling can be used directly in this way. In the case that the variance of different assignments of fuel elements to channels is too small even when a large β has been used, our suggestion is to use the ranked stand-alone K_{eff} matrix instead of the original one. The total $24 \times 35 = 840$ entries in the original stand-alone K_{eff} matrix are ranked from 1 to 840, which represent the relative contribution of all the possible assignments of fuel elements to fuel channels. We have used this method in this work. Unacceptable assignments (e.g. assigning Fuel Element 10 to any channel which is not channel 5) should not be included in this ranking as it will disturb the effect of the stand-alone with fuel coupling.

It should be noted that useful heuristic information like the stand-alone K_{eff} with fuel coupling is not always available and its values are highly reactor dependent and should be calculated a priori. So the calculation of the heuristics is problem-dependent, but the method of incorporating it into EDAs in the form described above can be generalized.

4. Results

GAs have been applied to NRLPO problems by many researchers and technologists [4] [7] [9] [11 to 14]. A very good summary addressing the NRLPO problem can be found in [15]. One of the successful algorithms is the GA with the HTBX operator, which has been applied to Pressurized Water Reactor (PWR) reloading optimization problem [4]. For this reason, we coded this algorithm (GA_HTBX) as the 'benchmark' algorithm for comparison with the results from the new EDA algorithms. The details of the HTBX operator can be found in [4], and are not to be presented here.

It should be noted that the performance of GAs and EDAs can be sensitive to the values of their control parameters. We have tuned the parameters used in the GAs and EDAs carefully to ensure the validity of the comparison. The parameters used are summarized in Table 3. Figure 2 shows the scalar flux contours calculated by 3D EVENT calculations for the fast and the thermal fluxes for a candidate LP investigated.

Numerical results after 100,000 LP evaluations are given in Table 4. For all EDAs and GAs, the best solution found in each generation was recorded, and their average values from 30 independent runs have been illustrated in Fig.3, which also shows the maximum and minimum objective function values (error bounds) found among 30 independent runs of EDA_H and GA_HTBX.

It has found that EDA_G and EDA_H algorithms both provided better solutions than GA_HTBX, as well as better averaged best solutions found over 30 independent runs. The standard deviations also suggest that both EDAs converge more quickly than GA_HTBX.

Table 3: The well-tuned parameters settings used in EDAs and GAs (α, β and η are described in equations (9) and (10))

Algorithms	Population Size	Maximum Generations	α	β	η	Mutation Rate	Crossover Rate
EDA G	50	2000	0.01	N/A	0.01	0.05	N/A
EDA H	50	2000	0.01	2	0.01	0.05	N/A
GAs	50	2000	N/A	N/A	N/A	0.05	0.9

Figure 2: The 3D fast flux (left) and thermal flux (right) contours obtained from EVENT calculations

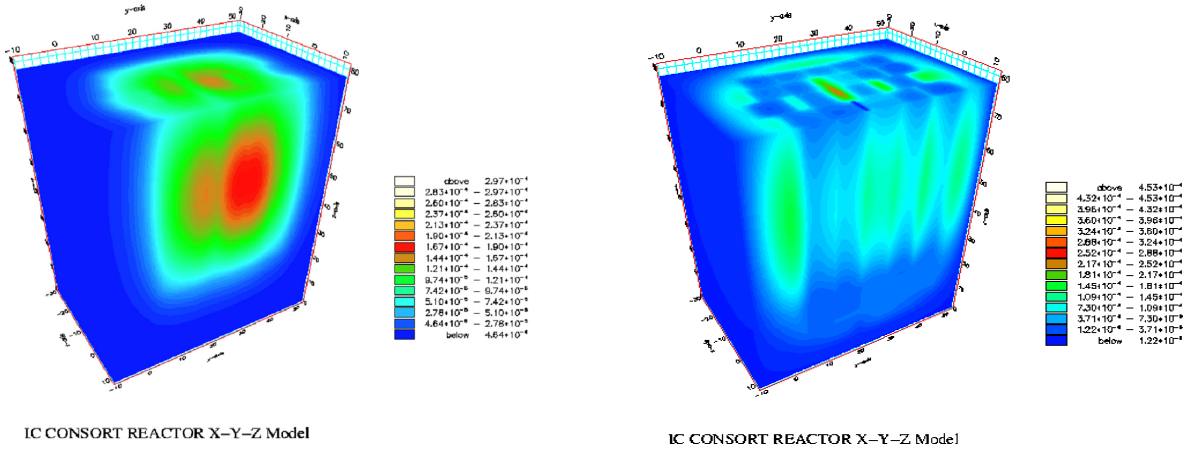
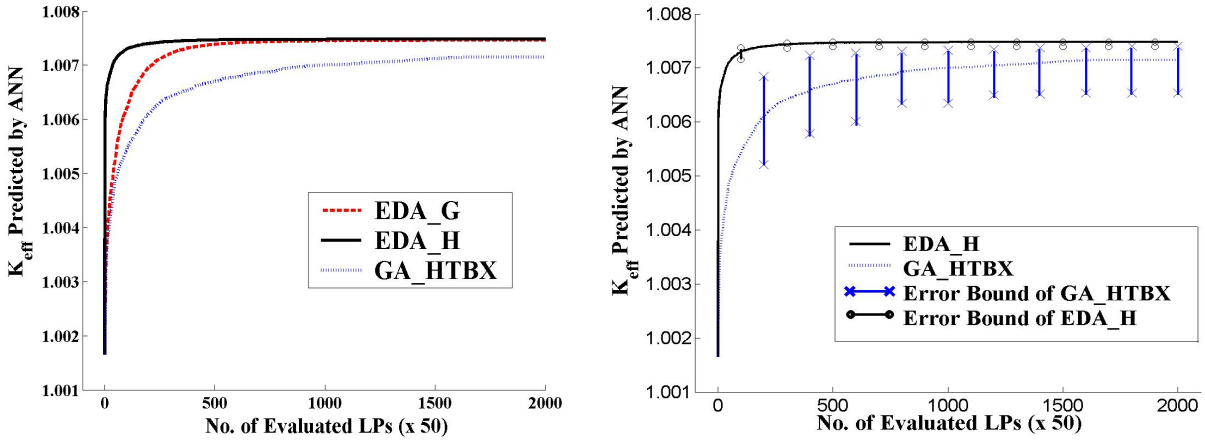


Table 4: The maximum K_{eff} found by EDAs and GAs from 30 independent runs and their corresponding averages and standard deviations

Algorithms	Best	Average	Standard Deviation
EDA_G	1.007480	1.007459	0.000039
EDA_H	1.007480	1.007477	0.000016
GA_HTBX	1.007400	1.007151	0.000221

Figure 3: Results from K_{eff} maximization of the CONSORT reactor comparing the performance of EDAs against the GA with HTBX operator.



5. Conclusions

The use of ANN technology has provided extremely fast LP evaluations and enabled sufficient optimization experiments to be carried out on the ICRC CONSORT LP optimization. The results have shown that EDA based algorithms are very efficient, accurate and robust for the test problem. The newly developed stand-alone K_{eff} with fuel coupling has been found to provide very useful heuristic information for LP optimization. Both EDAs, with and without heuristic information, are regarded as promising approaches for the tested NRLPO problem and worth further investigations.

Acknowledgement

This work is funded by Imperial College Reactor Center as part of the Reactor Future Options project.

References

1) S. J. Franklin, A. J. H. Goddard and J. S. O'Connell, "Research Reactor Facilities & Recent Developments at Imperial College London," Research Reactor Fuel Management, European Nuclear Society, Bruges, Belgium (1998).
2) S. J. Franklin, D. Gardner, J. Mumford, R. Lea and J. Knight, "Business operations and decommissioning strategy for Imperial College London research reactor 'CONSORT' – A financial risk management approach," Research Reactor Fuel Management 2005, European Nuclear Society, Budapest, Hungary (2005).
3) H. Muhlenbein and G. Paab, "From recombination of genes to the estimation of distributions I. Binary parameters," Lecture Notes In Computer Science, **1141**, pp.178-187 (1996).
4) P. W. Poon and G. T. Parks, "Application of genetic algorithms to in-core nuclear fuel management optimization" Proceedings of Joint International Conference of Mathematical Methods and Super-Computing in Nuclear Applications, **2**, pp.777-786 (2003).
5) EVENT: "A multi-group finite element spherical-harmonics program for the solution of time-dependent and steady-state radiation transport problems in arbitrary geometries, A User's Guide," C. R. E de Oliveira, Applied Modeling and Computation Group, Department of Earth Science and Engineering, Imperial College London, December (2000).
6) WIMS8A: "User Guide for Version 8," AEA Technology Report, ANSWERS/WIMS, Vol. **9**, (1999)
7) S. Jiang, A. K. Ziver, J. N. Carter, C. C. Pain, A. J. H. Goddard, S. Franklin and H. Phillips, "Estimation of Distribution Algorithms for Nuclear Reactor Fuel Management Optimisation," Annals of Nuclear Energy, In press (2006).
8) A. Zell, G. Mauier, M. Vogt, N. Mache, "SNNS Stuttgart Neural Network Simulator User Manual Version 4.1," Report No 6/95, Institute of Parallel and Distributed High Performance System (IPVR) (1995).
9) A. K. Ziver, C. C. Pain, J. N. Carter, C. R. E. Oliveira, A. J. H. Goddard and R. S. Overton, "Genetic Algorithms and Artificial Neural Networks for Loading Pattern Optimization of Advanced Gas Cooled Reactors," Annals of Nuclear Energy, **31**, 431-457, (2004) .
10) I. M. Oliver, D. J. Smith and J. R. C. Holland, "A Study of Permutation Crossover Operators on the Traveling Salesman Problem," Proceeding of the 2nd International Conference of Genetic Algorithms and Their Applications, pp.224-230. (1987).
11) A. K. Ziver, C. C. Pain, J. N. Carter, C. R. E. Oliveira, A. J. H. Goddard and R. S. Overton, "Multi-Cycle Optimisation of Advanced Gas Cooled Reactor Loading Patterns using Genetic Algorithms," Nuclear Technology, Vol. **141**, pp122-141 (2003).
12) A. Erdogan and M. Geckinli, "A PWR reload optimisation code (Xcore) using artificial neural networks and genetic algorithms," Annals of Nuclear Energy, **30**, pp.35-53 (2003).
13) G. T. Parks, "Multi-objective Pressurized Water Reactor Reload Core Design by non-dominated Genetic Algorithm Search," Nuclear Science and Engineering, **124**, pp.178-187 (1996).
14) C. M. N. A. Pereira, and C. M. F. Lapa, "Coarse-grained Parallel Genetic Algorithm Applied to a Nuclear Reactor Core Design Optimization Problem," Annals of Nuclear Energy **30**, pp.555-565 (2003).
15) P. J. Turinsky, P. M. Keller, H. S. Abdel-Khalik, "Evolution of Nuclear Fuel Management

and Reactor Operational Tools", Nuclear Engineering and Technology, **37**, No.1, PP.79-90 (2005).