

Multi-objective Differential Evolution Algorithms for Optimization of Heterogeneous LWR Assemblies

Alan J. Charles, Geoffrey T. Parks

University of Cambridge Department of Engineering, Trumpington Street, Cambridge, CB2 1PZ, ajc289@cam.ac.uk

Abstract - *The performance of Differential Evolution (DE) algorithms was investigated for two different Mixed Oxide (MOX) fuel assembly design problems using the WIMS lattice physics code. Two different new multi-objective DE algorithms were developed and tested. The first problem involved a performance comparison between the DE algorithms and a Genetic Algorithm (GA) from the literature in optimizing a CORAIL MOX fuel assembly with the objectives of maximizing the plutonium content and minimizing the Beginning of Life (BoL) Power Peaking Factor (PPF). Statistical analysis of these results indicated that DE exhibited superior performance to that of the GA. For the second problem, a multi-objective DE algorithm was used to optimize a Japanese MOX fuel assembly containing gadolinia poison rods to maximize the plutonium content and minimize the BoL PPF. Subsequent burnup analysis of a Pareto-front solution showed comparable internal assembly PPF performance against burnup to expert solutions found in the literature. These two studies demonstrate the applicability of DE in solving different nuclear engineering design optimization problems and form the basis for a more comprehensive investigation.*

I. INTRODUCTION

For some nuclear fuel cycles, particularly those involving Mixed Oxide (MOX) fuels, there are benefits in varying the properties of the fuel radially on a pin-by-pin basis and axially along the assembly. Heterogeneity in fuel design inevitably increases fuel fabrication costs, so a capability to explore rigorously and systematically the trade-off between in-core fuel performance and fabrication cost of heterogeneous assembly designs would be a helpful aid to decision-making. New reactor designs often feature more complex fuels for increased performance, and the design of these fuels is too difficult to optimize using conventional engineering judgement alone. This is mainly due to the high number of possible design variables and their non-linear interaction. A lack of suitable optimization methods has a negative impact on the design process, leading to higher fabrication and infrastructure costs as well as deteriorated performance.

There is a large variety of optimization methods on offer, with many studies applying different methods to different problems, with varying results. Although newer methods are constantly being developed, there exists little work demonstrating their applicability to specific engineering problems.

This study seeks to determine the suitability of a relatively new evolutionary metaheuristic algorithm called Differential Evolution (DE) to optimizing a typical nuclear fuel assembly design problem. Performance will be evaluated by comparing results obtained to those found using a Genetic Algorithm (GA), a more conventional method often used as a benchmark in previous studies [1, 2].

1. Genetic Algorithms

GAs [3] are one of the more dominant methods used in nuclear engineering design optimization. These evolutionary algorithms mimic the processes of natural selection in an attempt to create successive populations of solutions which converge on an optimal solution. For this work, the Multi-objective Alliance Algorithm (MOAA) [4] was chosen as it has previously demonstrated its effectiveness in nuclear engineering fuel assembly design problems, producing solutions which are superior to previous ‘expert designs’ and performing better than other GAs [5], thus making it a good algorithm for comparing performance.

2. Differential Evolution

In contrast, DE algorithms [6] are a relatively newer type of evolutionary algorithm that work in a similar fashion to GAs but feature key differences in the way the new population is generated. In addition, the selection process is generally more stringent than with GAs (where inferior solutions have a probability of remaining in the population). This allows DE algorithms to offer potentially faster convergence (which is useful for computationally expensive problems such as those faced in nuclear engineering) at the risk of premature convergence on a non-optimal solution [7]. DE algorithms have previously been successfully applied to core design optimization problems [8]. However, they do not yet appear to have been applied to nuclear fuel assembly optimization problems, thus making this investigation both novel and a useful step in examining DE’s applicability to solving such problems.

For this work, new multi-objective forms of the DE algorithms JADE [9] and μ JADE [10] are developed and their relative performance against the MOAA will be evaluated on a typical nuclear engineering fuel assembly design problem. JADE was chosen as it had been shown in

the literature to exhibit superior performance over classic DE algorithms through its use of a variation on the classic DE mutation strategy, and the inclusion of an archive and self-adapting control parameters. Similarly, μ JADE was chosen as it was shown to be effective when working on multimodal problems, but uses a significantly smaller population size. These features should help the algorithm perform well in a multi-objective environment, where diversity and convergence rates are important.

II. METHODOLOGY

1. Creating Multi-Objective JADE (MOJADE) and μ JADE (MO μ JADE)

MOJADE and MO μ JADE were created using C++ and are based on the JADE and μ JADE algorithms as described by the authors in [9, 10], with the following modifications implemented to allow them to operate in a multi-objective environment.

First, selection and ranking are no longer done based on one objective and the ‘best’ solutions are now a list of non-dominated solutions which represents the current Pareto-front. These are determined from the current population. Next, the archive was changed to accept solutions from that population that have been dominated by new solutions, and a second archive was added to accept new solutions that are Pareto-equivalent to the existing population. The pseudocode for MOJADE and MO μ JADE can be seen in Appendix A. Control parameters used for MOAA and MOJADE / MO μ JADE are given below in Table I and Table II, respectively. The values for these were the same as those used in the original literature. All algorithms feature self-adapting parameters which in theory reduces the necessity for tuning multiple control parameters.

Table I. MOAA control parameters

Number of tribes	6
Probability 1 for the creation of tribes	0.5
Probability 2 for the creation of tribes	0.2
Initial standard deviation	0.3
Final standard deviation	0.01
Probability 3 for the creation of alliances	2 / variables
Alliance standard deviation	0.1
Total number of Pareto-optimal solutions	100
Factor for evaluation neighbourhood	10

Table II. MOJADE and MO μ JADE control parameters

Parameter	MOJADE	MO μ JADE
Rate of parameter adaptation	0.1	0.05
Greediness of mutation strategy	0.05	3 / population
Population	32	8
Generations	50	200

MOJADE and MO μ JADE were subsequently tested on the ZDT-1 problem [11] and their results were compared to those obtained using the NSGA-II algorithm [12], using a test case of 41 dimensions. The test confirmed that both DE algorithms were able to find the Pareto-front with performance comparable to the NSGA-II algorithm and confirmed that they can successfully operate in a multi-objective environment.

2. Multi-objective optimization of nuclear engineering fuel assembly design problems

The first problem investigated was that described in [5]: to optimize a 2D nuclear fuel assembly featuring two fuel types for minimum Power Peaking Factor (PPF) and maximum Pu content at Beginning of Life (BoL). This problem concerns optimization of a so-called ‘CORAIL’ assembly, which contains both Low-Enriched Uranium (LEU) and MOX pins in the same assembly. The presence of both Pu and LEU creates a wide range of neutron energies with the fuel types having different reaction probabilities, resulting in uneven neutron flux, power distribution variations and subsequently fuel temperature problems. It is therefore necessary to optimize the distribution of Pu pins in order to mitigate these issues. By varying the position of MOX pins and the amount of Pu contained within the pins, it is possible to reduce the PPF and improve reactor thermal margins, whilst (as [5] demonstrated) increasing the overall Pu content above that of the standard ‘CORAIL’ expert design. Constraints affecting this problem include a requirement that the minimum number of LEU pins must be half the total, and a maximum %Pu within the MOX pins (20%).

LEU enrichment is fixed at 5% U235 and reactor grade Pu composition is assumed. A standard CORAIL assembly contains 264 fuel pins, which when divided using octant symmetry, giving 39 unique fuel pin positions. These pins are labelled as fuel types 1, 2 and 3 (MOX type 1, MOX type 2, and LEU, respectively). The quantities N_1 , N_2 , and N_3 are therefore the total number of each respective pin type, where $N_1 + N_2 + N_3 = 39$. Some pins are weighted by 0.5 due to octant symmetry in the assembly. Two %Pu MOX pin types are allowed (W_1 , W_2). This gives a total of 39 integer variables, 2 continuous variables, with the constraints of $N_3 \geq 16.5$ (264/8) and $0 \leq W_1, W_2 \leq 20$. The Pu content is therefore $MOXT = W_1 \cdot N_1 + W_2 \cdot N_2$. Both objectives to be minimized are therefore PPF and $-MOXT$. PPF is obtained using the reactor physics code WIMS10a [13] to solve the neutron transport equation using the method of characteristics to calculate pin power and hence the PPF.

In this work, the MOAA algorithm was used as a benchmark to evaluate the performance of MOJADE and MO μ JADE. MOAA was implemented using the same default parameters used in [5]. Algorithms were run on the ‘Ray’ computer cluster used by the University of

Cambridge's Department of Engineering, with specifications shown in Table III.

Table III. Ray computer cluster specifications

Processor	Dual Intel Xeon Processor E5-2650 (8-core hyper threading, 2.6 GHz Turbo with 20 MB cache)
RAM	64 GB (8x8 GB) 1866 MHz DDR3
Video Card	512 MB AMD FirePro 2270
Hard Drive	1TB 7,200 rpm

The second problem was a preliminary investigation on the use of multi-objective optimization algorithms in optimizing MOX fuel assemblies containing gadolinia (Gd_2O_3) pins (e.g. Japanese MOX assemblies [14]). Employment of gadolinia pins in these assemblies potentially reduces the need to use Burnable Poison Rods (BPRs) in the guide tubes, normally employed to compensate for higher PPF values caused by higher levels of Pu content compared to other designs. By optimizing the design using Gd pins, the PPF can be reduced without using BPRs and can even allow for increased Pu content in the assembly.

In [14], one assembly was optimized for minimum PPF over the life of the assembly, using a fixed %Gd content, fixed pin types and changing %Pu contents for two types of UPu MOX pin. Using multi-objective algorithms, it is possible to further explore the search space for this problem. The variables were changed to include all five originally proposed assembly layouts, allowing %Gd and %Pu to change, and allowing all non-Gd pins to be either type of Pu MOX pin. The different assembly layouts used can be seen below in Fig. 1.

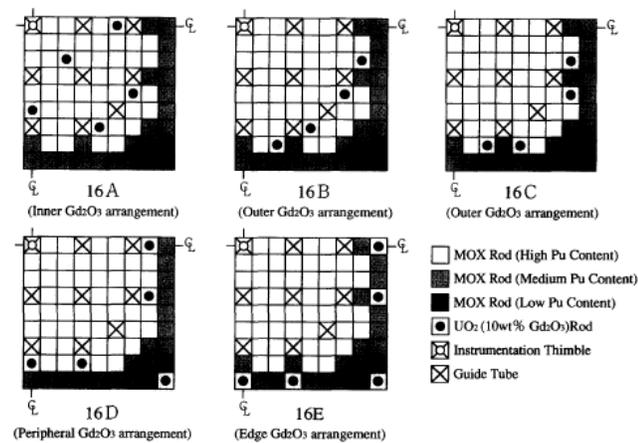


Fig. 1. Japanese UPu MOX 1/4 assembly layouts used in the second problem (taken from [14]).

Constraints of 10% Gd in gadolinia pins and 20% Pu for the MOX pins were used. MOJADE was used to optimize the problem with the objectives of maximizing the

Pu content of the assembly and minimizing the PPF for a BoL assembly. Following this, depletion of a randomly chosen solution from the resulting Pareto-front was then performed to see how the PPF changed over the life, with results compared to [14]. MOJADE was used with the same values of control parameters as used for the first problem to see how well the algorithm would perform without custom tuning of any control parameters. Again the reactor physics code WIMS10a [13] was used to calculate values for PPF and to perform depletion calculations.

III. RESULTS

1. Results on first problem

Each algorithm was run 30 times, with a limit of 1600 solution evaluations per run. Analysis of the results involved comparing Pareto-fronts found by the three algorithms using a number of indicators and statistical tests to determine the performance of each algorithm. The indicators used were the epsilon and hypervolume indicators. The epsilon indicator [15] represents the minimum distance required to translate all the points of the found Pareto-front from a given algorithm to weakly dominate the reference set (which is the overall Pareto-front constructed from all solutions found by all algorithms). The hypervolume indicator [16] is the difference between the hypervolume of the objective space dominated by the Pareto-front found by a particular algorithm and the hypervolume of the objective space dominated by the reference set, using the least-optimal solution found from all three algorithms as a reference point. In both cases smaller values indicate better performance. The Kruskal-Wallis test [17] was then used to determine the statistical significance of these values. For this work, the Kruskal-Wallis test results represent the probability that the given indicator values are not a true representation of the algorithm's relative performance against another, and are instead the result of random chance.

A graph showing the results from all 30 runs for each of the three algorithms can be seen in Fig. 2. Filtering these to just show the Pareto-Optimal (PO) solutions for each algorithm produces Fig. 3. Both figures plot solutions in PPF - (-MOXT) space, so that both objectives are to be minimized and thus the bottom-left corner represents an ideal solution. It can be seen from these two graphs that MOJADE and MOμJADE are clearly performing comparably to MOAA, and both significantly contribute to the highlighted overall Pareto-front shown in Fig. 3. MOAA appears to dominate the Pareto-front at both the extremes of plutonium content, with MOJADE and MOμJADE being most effective around the middle of the Pareto-front. A small amount of clustering is also present in the PO solutions of MOAA, with clear gaps in the Pareto-front that are populated by MOJADE and MOμJADE solutions.

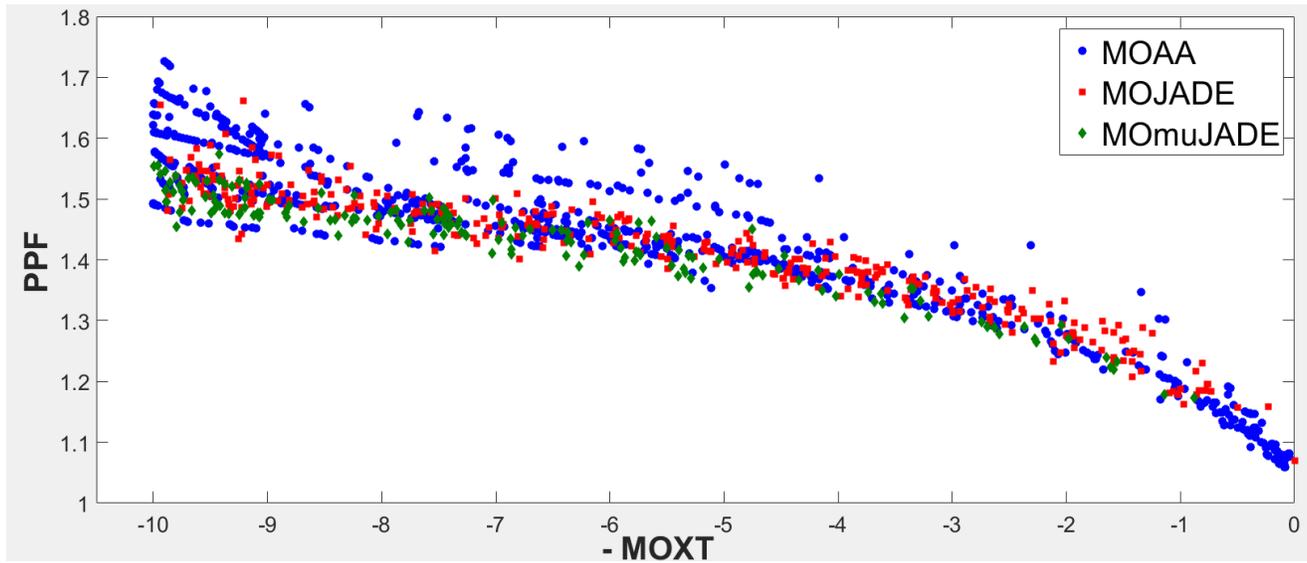


Fig. 2. Results of MOAA, MOJADE and MO μ JADE optimization of MOX fuel assemblies.

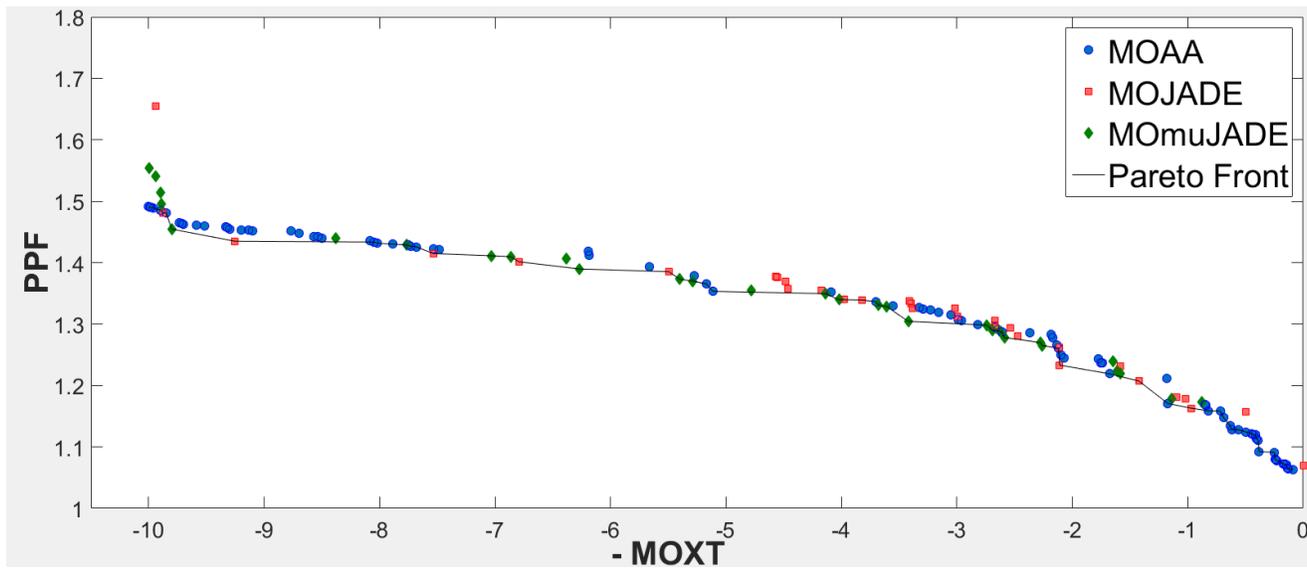


Fig. 3. Comparison of non-dominated solutions found using MOAA, MOJADE and MO μ JADE algorithms to optimize MOX fuel assemblies.

The clustering of MOAA results can be explained by looking at the patterns themselves. Each MOAA run tends to converge on a single pin pattern with one or two changes in the pins, and the non-dominated solutions therefore show the effect of increasing or decreasing the values of W_1 and/or W_2 within the same pin pattern. This results in a number of solutions that have very similar values for $MOXT$ and PPF . In contrast, both MOJADE and MO μ JADE produce many different patterns within each run and thus arguably better explore the search space of different pin arrangements.

Results from each of the 30 runs for each algorithm were used to compute the mean and standard deviation of the hypervolume and epsilon indicators along with their corresponding p-values from the Kruskal-Wallis test. These can be seen in Tables IV – VI respectively.

Table IV indicates that MOJADE is most consistent at producing results which dominate the entirety of the known search space. MO μ JADE performs slightly worse and also has a large standard deviation. This is likely due to the small population of MO μ JADE, making its performance highly influenced by the number of times the algorithm is run (more so than the other algorithms), in order to provide more data. Table V, however, suggests that, whilst MO μ JADE results do not give as much data as MOJADE as to the size of the search space, they are more likely to be closer to the ‘true’ Pareto-front.

Table VI gives the p-value results of the Kruskal-Wallis test for the hypervolume and epsilon indicators for both DE algorithms versus the GA, as well as against each other. Against MOAA, hypervolume test results show that MOJADE performs better due to methodology and not by chance. However, for the epsilon indicator, the result shows a statistical likelihood that differences in epsilon

performance between MOAA and MOJADE are by chance. For MO μ JADE, the results suggest that the DE algorithm outperforms the GA on both indicators due to the methodology. Finally, when comparing MO μ JADE against MOJADE, the Kruskal-Wallis test results indicate that the lower mean hypervolume indicator for MOJADE and the lower mean epsilon indicator for MO μ JADE are due to differences in their methodology and not by chance.

Table IV. Hypervolume indicator values

Algorithm	Mean	Standard Deviation
MOAA	1.6664	0.5169
MOJADE	0.7672	0.1047
MO μ JADE	1.1267	0.7723

Table V. Epsilon indicator values

Algorithm	Mean	Standard Deviation
MOAA	0.3897	0.1478
MOJADE	0.3941	0.1204
MO μ JADE	0.3320	0.1081

Table VI. Kruskal-Wallis test results

Algorithm	Hypervolume	Epsilon
MOJADE vs MOAA	3.879E-11	9.528E-01
MO μ JADE vs MOAA	8.513E-07	7.363E-02
MO μ JADE vs MOJADE	9.497E-05	5.650E-02

2. Results on second problem

MOJADE was run 20 times with a limit of 1600 solution evaluations per run. A graph showing the results of MOJADE, the Pareto-front, and the solution chosen for depletion can be seen below in Fig. 4.

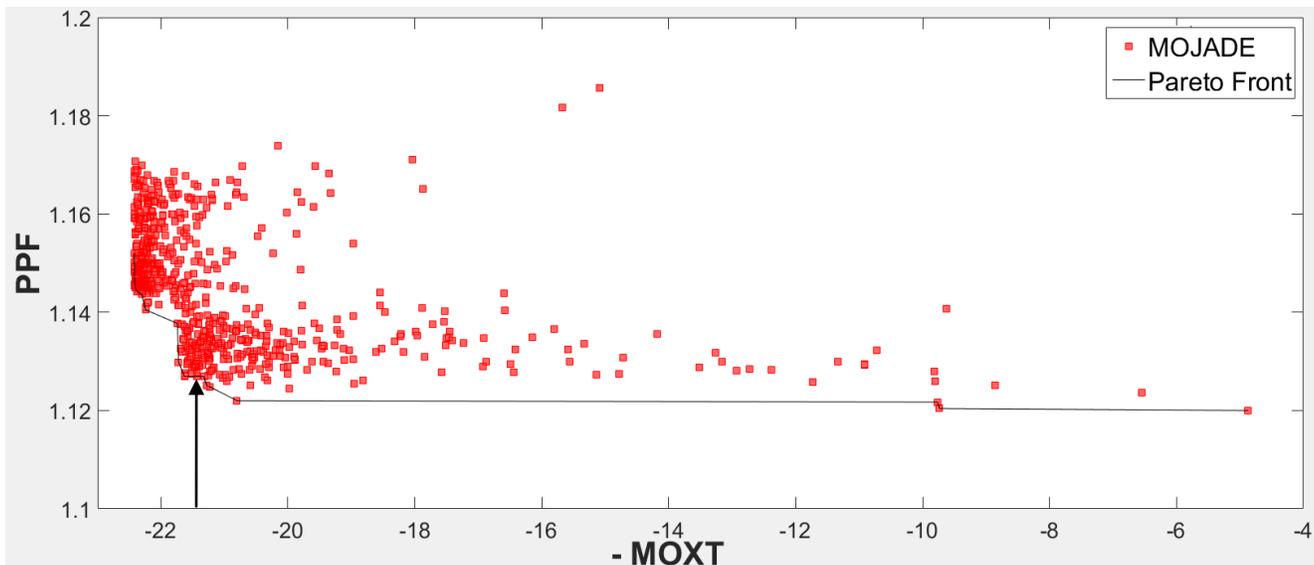


Fig. 4. Results of MOJADE optimization of MOX fuel assemblies with gadolinia pins. The arrow indicates the solution chosen for the burnup study.

The graph clearly shows that MOJADE tends to converge on solutions which contain high amounts of Pu, and the Pareto-front for solutions with lower values of MOXT is poorly populated. This may be due to some form of premature convergence in the population diversity (either through the crossover rate or mutation rate), where a point of local optima of solutions containing high amounts of Pu causes the algorithm to converge prematurely. The crossover and mutation rates are self-adapting control parameters which are in turn affected by the greediness and the rate of parameter adaption, as specified above in Table II. Due to time constraints, a sensitivity study of these control parameters was not possible but will be investigated in the future, as this test has shown that some problems still require some tuning of control parameters in order to fully explore the search space.

Following this, one solution on the found Pareto-front was chosen at random (shown in the arrow in Fig. 4) and depletion to 15 GWd/t was performed. The evolution of the PPF against burnup for an example assembly generated by MOJADE can be seen in Fig. 5 as the blue line, overlaid against the original results of [14].

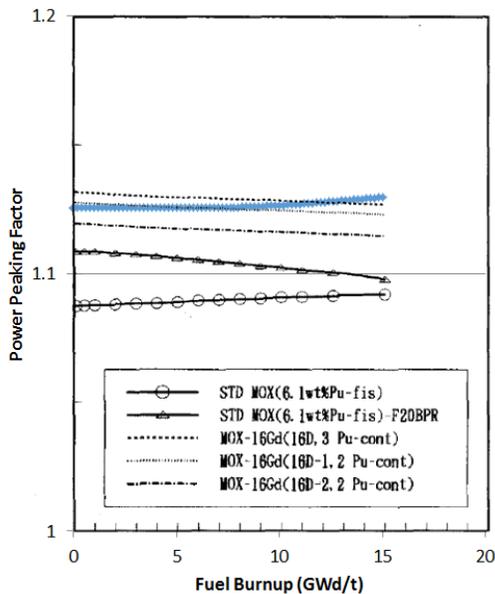


Fig. 5. PPF progression with burnup for gadolinia-MOX fuel assemblies (adapted from [14]).

The assemblies investigated in the original paper ranged from 5.7% to 6.4% average Pu enrichment. The chosen MOJADE solution had an average Pu enrichment of 19.8%. This test shows that MOJADE is able to produce solutions that contain more Pu and keep internal PPF performance over one cycle comparable to assemblies with much lower Pu contents. The assembly layout generated by MOJADE and investigated against burnup, and an example

expert solution assembly from the literature are shown below in Fig. 6.

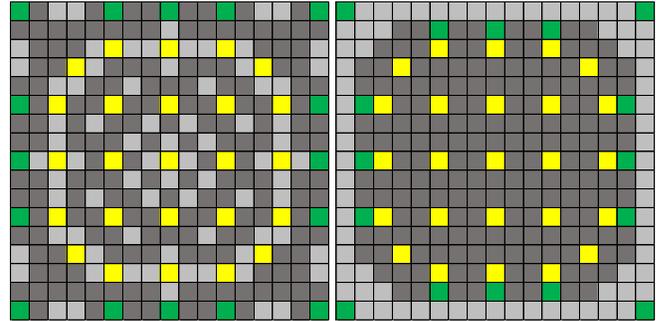


Fig. 6. UPu MOX assembly layouts with gadolinia poison rods produced using MOJADE (left) and from the literature (right). Light grey and dark grey indicate MOX pins (dark grey have higher %Pu contents), green indicates a poison rod, and yellow indicates guide tubes.

Further work in this area will require a more substantial burnup study. As this test demonstrated, the PPF is not necessarily highest at BoL and, as such, the algorithm should take the change of PPF with burnup into account during the optimization process. A more complete study will require looking at multiple cycles, assembly PPF performance in cluster geometries, as well as tests using the smaller population variant MO_μJADE. Therefore it is premature to draw any firm conclusions on the performance of DE algorithms in optimizing MOX problems involving the use of gadolinia poison pins. However, these first tests show promising results and demonstrate a proof of principle in the applicability of DE algorithms to solving more complex nuclear engineering design optimization problems.

IV. CONCLUSIONS

The results suggest that DE is able to find solutions comparable in quality to those found by MOAA and arguably provides solutions that are better spread than MOAA. Both DE algorithms show reasonable performance in these exploratory optimization problems, despite the algorithms originally being designed for single-objective optimization with a known global optimum.

A sensitivity study on each algorithm control parameter is recommended to further investigate its effect on performance, although it should be reiterated that all three algorithms feature self-adapting control measures which reduce the requirement for tuning an algorithm to a particular problem. Further testing should also include more complex problems, including the introduction of thermal-hydraulic feedback mechanics and the use of 3D models for axial optimization of fuel zoning (e.g. for use in BWRs).

Nevertheless, this study demonstrates that DE appears to be a promising new method for optimization of the design of nuclear fuel assemblies.

APPENDIX A: PSEUDOCODE OF MOJADE AND MO μ JADE

MOJADE

Begin

Set $\mu\text{CR} = 0.5$; $\mu\text{F} = 0.5$; $A1, A2 = 0$
 Create random initial population $\{x_{i,0} | i = 1, 2, \dots, \text{NP}\}$
 Evaluate and rank population, determine 100% best vectors

For g = 1 to G

SF = 0, SCR = 0

For i = 1 to NP

$\text{CR}_i = \text{rand}_i(\mu\text{CR}, 0.1)$, $F_i = \text{rand}_i(\mu\text{F}, 0.1)$
 Randomly choose x_{p_best} from 100%
 Randomly choose $x_{r1} \neq x_i$ from P
 Randomly choose $x_{r2} \neq x_{r1} \neq x_i$ from $P \cup A1 + A2$
 $v_i = x_i + F_i \cdot (x_{p_best} - x_i) + F_i \cdot (x_{r1} - x_{r2})$
 Generate jrand = randint(1, D)

For j = 1 to D

If j = jrand or $\text{rand}(0, 1) < \text{CR}_i$

$u_{i,j} = v_{i,j}$

Else

$u_{i,j} = x_{i,j}$

End If

End For

If $f(u_i)$ dominates $f(x_i)$

$x_i \rightarrow A1$ (replaces random member of A1 if A1 is

full)

$x_i = u_i$

$\text{CR}_i \rightarrow \text{SCR}$, $F_i \rightarrow \text{SF}$

Else

If $f(u_i)$ is Pareto-equivalent to $f(x_i)$

&& $f(u_i)$ is NOT dominated by $f(A2)$

Remove members of A2 that are dominated by

u_i

$u_i \rightarrow A2$

End If

End For

Rerank 100% best vectors

End For

$\mu\text{CR} = (1 - c) \cdot \mu\text{CR} + c \cdot \text{meanA}(\text{SCR})$

$\mu\text{F} = (1 - c) \cdot \mu\text{F} + c \cdot \text{meanL}(\text{SF})$

End For

End

MO μ JADE

Begin

Set $\mu\text{CR} = 0.5$; $\mu\text{F} = 0.5$; $A1, A2 = 0$

Create random initial population $\{x_{i,0} | i = 1, 2, \dots, \text{NP}\}$

Evaluate and rank population, determine 100% best vectors

For g = 1 to G

SF = 0, SCR = 0

For i = 1 to NP

$\text{CR}_i = \text{rand}_i(\mu\text{CR}, 0.1)$, $F_i = \text{rand}_i(\mu\text{F}, 0.1)$

Randomly choose $x_a \neq x_i$ from P

Randomly choose $x_b \neq x_a \neq x_i$ from P

Randomly choose $x_{p_best} \neq x_a$ from 100%

Randomly choose x_c from $P \cup A1 + A2$

$v_i = x_i + F_i \cdot (x_{p_best} - x_a) + F_i \cdot (x_b - x_c)$

Generate jrand = randint(1, D)

For j = 1 to D

If j = jrand or $\text{rand}(0, 1) < \text{CR}_i$

$u_{i,j} = v_{i,j}$, $b_{i,j} = 1$

Else

$u_{i,j} = x_{i,j}$, $b_{i,j} = 0$

End If

End For

For j = 1 to D

If $\text{rand}(0, 1) \leq 0.005$

$u_{i,j} = \text{low_lim} + \text{rand}(0, 1) \cdot (\text{up_lim} -$

low_lim)

$b_{i,j} = 0$

Else

$u_{i,j} = u_{i,j}$, $b_{i,j} = b_{i,j}$

End If

End For

$\text{CR}_i = \sum b / D$

If $f(u_i)$ dominates $f(x_i)$

$x_i \rightarrow A1$ (if A1 is full, replaces random member)

$x_i = u_i$

$\text{CR}_i \rightarrow \text{SCR}$, $F_i \rightarrow \text{SF}$

Else

If $f(u_i)$ is Pareto-equivalent to $f(x_i)$

&& $f(u_i)$ is NOT dominated by $f(A2)$

Remove members of A2 that u_i dominate

$u_i \rightarrow A2$

End If

End For

Rerank 100% best vectors

If $u_i \cup 100\%$ best vectors

$\text{BIR} = \text{BIR} + 1$

End If

End For

If $\text{mod}(g, \max(100, 10D)) = 0$

$\mu\text{CR} = (1 - c) \cdot \mu\text{CR} + c \cdot \text{meanA}(\text{SCR})$

$\mu\text{F} = (1 - c) \cdot \mu\text{F} + c \cdot \text{meanL}(\text{SF})$

End If

If $\text{mod}(g, \max(1000, 100D)) = 0$

If $\text{BIR} == 0$

Reinitialize pop, include random member of 100%

$\text{BIR} = 0$

End If

End If

End For

End

NOMENCLATURE

BPR = Burnable Poison Rod
DE = Differential Evolution
GA = Genetic Algorithm
LEU = Low Enriched Uranium
LWR = Light Water Reactor
MOAA = Multi-objective Alliance Algorithm
MOJADE = Multi-objective JADE
MO μ JADE = Multi-objective μ JADE
MOX = Mixed Oxide
PPF = Power Peaking Factor

μ CR = adaptive crossover probability
 μ F = adaptive mutation probability
A1 = archive used for dominated solutions
A2 = archive used for Pareto-equivalent solutions
BIR = restart variable used if no improvement is made
c = rate of parameter adaptation
D = number of dimensions (variables)
G = number of generations
meanA = arithmetic mean
meanL = Lehmer mean
NP = last member of the population
p = greediness of the mutation strategy
P = population
randn = normal distribution
randc = Cauchy distribution
SCR = set of successful crossover factors
SF = set of successful mutation factors
up_lim / low_lim = limits set by the variable constraints
bi = crossover rate repair modifier following perturbation
vi = ith test vector following mutation
ui = ith test vector following crossover and perturbation
xi = ith member of the population

REFERENCES

1. A. CASTILLO, C. MARTIN-DEL-CAMPO, J.L. MONTES-TADEO, J.L. FRANCOIS, J.J. ORTIZ-SERVIN, R. PERUSQUIA-DEL-CUETO, "Comparison of Heuristic Optimization Techniques for the Enrichment and Gadolinia Distribution in BWR Fuel Lattices and Decision Analysis", *Ann. Nucl. Energy*, **63**, 556 (2014).
2. M.L. JAYALAL, S. RAMACHANDRAN, S. RATHAKRISHNAN, S.A.V. SATYA MURTY, M. SAI BABA, "Application of Genetic Algorithm Methodologies in Fuel Bundle Burnup Optimization of Pressurized Heavy Water Reactor", *Nucl. Eng. Des.*, **281**, 58 (2015).
3. J.H. HOLLAND, "Adaptation in Natural and Artificial Systems", MIT Press, Cambridge, Massachusetts (1992).
4. V. LATTARULO, G. PARKS, "A Preliminary Study of a new Multi-objective Optimization Algorithm", *Proc. IEEE Congress on Evolutionary Computation*, Brisbane, June 10–15, p. 1, IEEE (2012).
5. V. LATTARULO, B.A. LINDLEY, G. PARKS, "Application of the MOAA for the Optimization of CORAIL Assemblies for Nuclear Reactors", *Proc. IEEE Congress on Evolutionary Computation*, Beijing, July 6–11, p. 1413, IEEE (2014).
6. R. STORN, K. PRICE, "Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces", *J. Global Optim.*, **11**, 4, 341 (1997).
7. E. ZIO, G. VIADANA, "Optimization of the Inspection Intervals of a Safety System in a Nuclear Power Plant by Multi-Objective Differential Evolution (MODE)", *Reliab. Eng. Syst. Safe.*, **96**, 11, 1552 (2011).
8. W.F. SACCO, N. HENDERSON, A.C. RIOS-COELHO, M.M. ALI, C.M.N.A. PEREIRA, "Differential Evolution Algorithms Applied to Nuclear Reactor Core Design", *Ann. Nucl. Energy*, **36**, 8, 1093 (2009).
9. J. ZHANG, A.C. SANDERSON, "JADE: Adaptive Differential Evolution with Optional External Archive", *IEEE Trans. Evol. Comput.*, **13**, 5, 945 (2009).
10. C. BROWN, Y. JIN, M. LEACH, M. HODGSON, " μ JADE: Adaptive Differential Evolution with a Small Population". *Soft Comput.*, **1**, 1 (2013).
11. E. ZITZLER, K. DEB, L. THIELE, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results", *Evol. Comput.*, **8**, 2, 173 (2000).
12. K. DEB, A. PRATAP, S. AGARWAL, T. MEYARIVAN, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II", *IEEE Trans. Evol. Comput.*, **6**, 2, 182 (2002).
13. B.A. LINDLEY, T.D. NEWTON, J.G. HOSKING, P.N. SMITH, D.J. POWNEY, B. TOLLIT, P.J. SMITH, "Release of WIMS10: A Versatile Reactor Physics Code for Thermal and Fast Systems", *Proc. ICAPP 2015*, Nice, May 3–6, p. 1793, ANS (2015).
14. K. YAMATE, M. MORI, T. USHIO, M. KAWAMURA, "Design of a Gadolinia Bearing Mixed-Oxide Fuel Assembly for Pressurized Water Reactors", *Nucl. Eng. Des.*, **170**, 1-3, 35 (1997).
15. E. ZITZLER, L. THIELE, M. LAUMANN, C.M. FONSECA, V. GRUNERT DA FONSECA, "Performance Assessment of Multiobjective Optimizers: An Analysis and Review", *IEEE Trans. Evol. Comput.*, **7**, 2, 117 (2003).
16. J. KNOWLES, L. THIELE, E. ZITZLER, "A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers", TIK Report 214, Swiss Federal Institute of Technology (ETH) (2006).
17. W.H. KRUSKAL, W.A. WALLIS, "Use of Ranks in One-criterion Variance Analysis", *J. Am. Stat. Assoc.*, **47**, 260, 583 (1952).