Novel Approaches on Reactor Core Design Optimization Problem

Novas Abordagens no Problema de Otimização do Projeto do Núcleo do Reator

Marco A. A. Kappel[1,†], Ivan N. Bastos[2], Caio L. S. Souza[2], Gustavo B. Libotte[2], Roberto P. Domingos[2], Gustavo M. Platt[3]

[1] *Centro Federal de Educação Tecnológica Celso Suckow da Fonseca – CEFET/RJ – Campus Nova Friburgo, Nova Friburgo, RJ, Brasil*
[2] *Instituto Politécnico, Universidade do Estado do Rio de Janeiro, Nova Friburgo, Brasil*
[3] *Escola de Química e Alimentos, Universidade Federal do Rio Grande, Santo Antônio da Patrulha, Brasil*
[†] **Corresponding author:** marco.kappel@cefet-rj.br

Abstract

Nuclear reactor core design is an optimization problem concerning the pertinent choice of a series of parameters that must obey some technical and physical constraints. Several methods have been applied in literature in order to obtain the optimal solution for this problem. The present work aims to provide a comparative analysis of two optimization methodologies in the reactor core design, as follows: Invasive Weed Optimization and Many-Objective Evolutionary Algorithm.

Keywords

Nuclear reactor core design • Optimization • Differential Evolution • Invasive Weed Optimization • Many-Objective Evolutionary Algorithm

Resumo

O projeto do núcleo de um reator nuclear é um problema que consiste na escolha pertinente de uma série de parâmetros que devem obedecer a algumas restrições técnicas e físicas. Diversos métodos têm sido aplicados na literatura especializada de modo a obter-se soluções ótimas para este problema. O presente trabalho tem como objetivo apresentar um análise comparativa de duas metodologias de otimização, quais sejam: *Invasive Weed Optimization* e Many-Objective Evolutionary Algorithm.

Palavras-chave

Projeto do núcleo do reator • Otimização • Evolução Diferencial • Invasive Weed Optimization • Many-Objective Evolutionary Algorithm

1 Introduction

The design of a nuclear reactor core is a quite difficult task where the designer is faced with many restrictions and many nonlinear relations on variables of the problem. The elements that comprise a nuclear reactor core must simultaneously meet both operating limits constraints and reactor configuration limits. The choice of a pertinent set of parameters is of fundamental importance to design of a nuclear core; for instance, the type of material employed [1]. Such complex task can be seen as an optimization process on a multimodal high dimensional space. Not so long ago, traditional gradient-based optimization techniques using linear-programming and perturbation analysis have been applied [2]. However, because of the high complexity, nonlinearity, multimodality and the lack of knowledge about the search domain, the use of more robust and appropriate techniques such as Simulated Annealing [3], genetic algorithms [4] and Differential Evolution [5] have been also proposed.

Metaheuristics based methods demand the appropriate choice of execution parameters which has considerable impact on its performance, in fact a set of parameters can lead to a reasonable solution in a timely manner or complete failure of the search process.

A newly proposed metaheuristic called Many-Objective Evolutionary Algorithm (MOEA) was investigated. This optimization method aims to reduce the common problem of diversity loss during the search process. A second method was also applied to the problem, a numerical stochastic optimization algorithm inspired on colonizing weeds, Invasive Weed Optimization (IWO). This metaheuristic simulates the adaptative behavior of weeds, undesired kind of plants that interfere negatively in agriculture because of their ease of dispersion, rapid growth and continuous production. Essentially, the algorithm comprises three steps: reproduction, spatial dispersal and competitive exclusion of data structures representing solutions to the problem, the plants. The key concept of this populational algorithm is based on grouping fitter plants and elimination of inappropriate ones, highlighting the invasive habits of growth and evolution of the species.

The solutions obtained from the two methods were compared, a comparison with previous works was also performed.

This work is organized as follows: section 2 explains the nuclear reactor core design problem and presents all parameters to be estimated. Section 3 shows the applied optimization methods details and algorithms. In section 4, the results for each method execution are presented and compared. In section 5, the conclusions for the present work are presented.

2 The nuclear reator core design problem

We consider a simplified cylindrical three-enrichment-zone pressurized water reactor with a typical cell composed of moderator (light water), cladding, and fuel. Fig. 1 illustrates such a reactor and the dimensions are in centimeters. The design parameters, which may be adjusted in the optimization process, as well as their variation ranges are shown in Table 1.

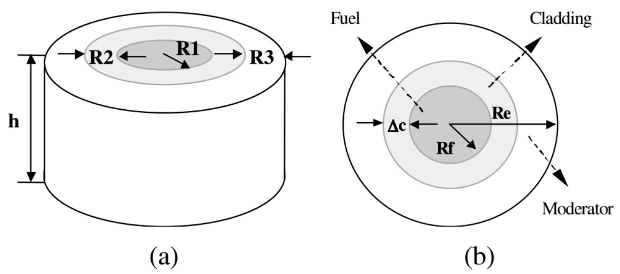


Figure 1: The reactor (a) and its typical cell (b).

The objective of the optimization problem is to minimize the average power peak factor f_p of the proposed reactor for a given average thermal flux f_0, considering as constraints the criticality ($k_{\text{eff}} = 1.00 \pm 0.01$) and sub-moderation. So, the optimization problem can be written as a minimization of the function f_p (R_{fuel}, R_{clad}, R_{eq}, Enr_1, Enr_2, Enr_3, M_{fuel}, M_{clad}) subject to:

$$\Phi\left(R_{fuel}, R_{clad},\ R_{eq},\ Enr_1,\ Enr_2,\ Enr_3,\ M_{fuel},\ M_{clad}\right) = \Phi_0 \tag{1}$$

$$0.99 \le k_{\text{eff}}\left(R_{fuel},\ R_{clad},\ R_{eq},\ Enr_1,\ Enr_2,\ Enr_3,\ M_{fuel},\ M_{clad}\right) \le 1.01 \tag{2}$$

$$\frac{dk_{\text{eff}}}{dV_{\text{m}}} > 0 \tag{3}$$

$$R_{fuel,\min} \le R_{fuel} \le R_{fuel,\max} \tag{4}$$

$$R_{clad,\min} \le R_{clad} \le R_{clad,\max} \tag{5}$$

$$R_{eq,\min} \leq R_{eq} \leq R_{eq,\max} \tag{6}$$

$$Enr_{1,\min} \leq Enr_1 \leq Enr_{1,\max} \tag{7}$$

$$Enr_{2,\min} \leq Enr_2 \leq Enr_{2,\max} \tag{8}$$

$$Enr_{3,\min} \leq Enr_3 \leq Enr_{3,\max} \tag{9}$$

$$M_{fuel} = (UO_2,\ \text{U-metal}) \tag{10}$$

$$M_{clad} = (\text{Zircaloy-2},\ \text{aluminum},\ \text{304 stainless steel}) \tag{11}$$

where V_m is the moderator volume and the min and max subscripts refer to the lower and upper limits of the parameter ranges. The nature and ranges of each parameter are shown in Table 1.

Table 1: Parameter Ranges.

Parameter	Symbol	Ranges
Fuel Radius Size (cm)	R_{fuel}	0.508 to 1.27
Cladding thickness (cm)	R_{clad}	0.0254 to 0.254
Moderator thickness (cm)	R_{eq}	0.0254 to 0.762
Enrichment of zone 1 (%)	Enr_1	2.0 to 5.0
Enrichment of zone 2 (%)	Enr_2	2.0 to 5.0
Enrichment of zone 3 (%)	Enr_3	2.0 to 5.0
Fuel Material	M_{fuel}	U-metal or UO_2
Cladding Material	M_{clad}	Zircaloy-2, aluminum or 304 stainless steel

The HAMMER system [6] was used to solve cell and diffusion equations. It performs a multigroup calculation of the thermal and epithermal flux distribution from the integral transport theory in a unit cell of the lattice.

3 Optimization methods

3.1 Differential Evolution

In order to assess the complexity of the nuclear reactor design for any metaheuristic method confidence regions was obtained for each parameter of a solution found by a Differential Evolution (DE) execution. DE is a well known and popular metaheuristic already applied to this problem before [5]. It was developed by Storn & Price [7] and the main idea is to apply the so-called evolutionary operators to modify a population of vectors, each vector characterizes a possible solution of the problem. The applied operators are mutation, crossover, and selection. Differential Evolution shares several features with the basic cycle of an evolutionary algorithm as shown in Fig. 2.

The mutation operator is applied to create new individuals from a group of individuals of the previous generation. It consists of a weighted addition of the difference between two randomly selected vectors v_1, v_2 to a third vector v_3 which is also randomly selected. The weight, named F, is a configuration parameter of the algorithm, controlling the variation level of vector difference. To the modified vector v_{mut}, resulting from the previous operations, a crossover is applied. A fourth random vector v_4 is again selected from the population and its components are mixed to the modified vector v_{mut}, generating a test vector v_{test}. At last, the selection operator is used, by the application of the objective function to v_{test} and v_4 , in order to identify which one is the best solution candidate. Then, the chosen one is transferred to the next generation. This process repeats until the stopping criterion is reached, usually characterized by the minimization or maximization of the objective function.

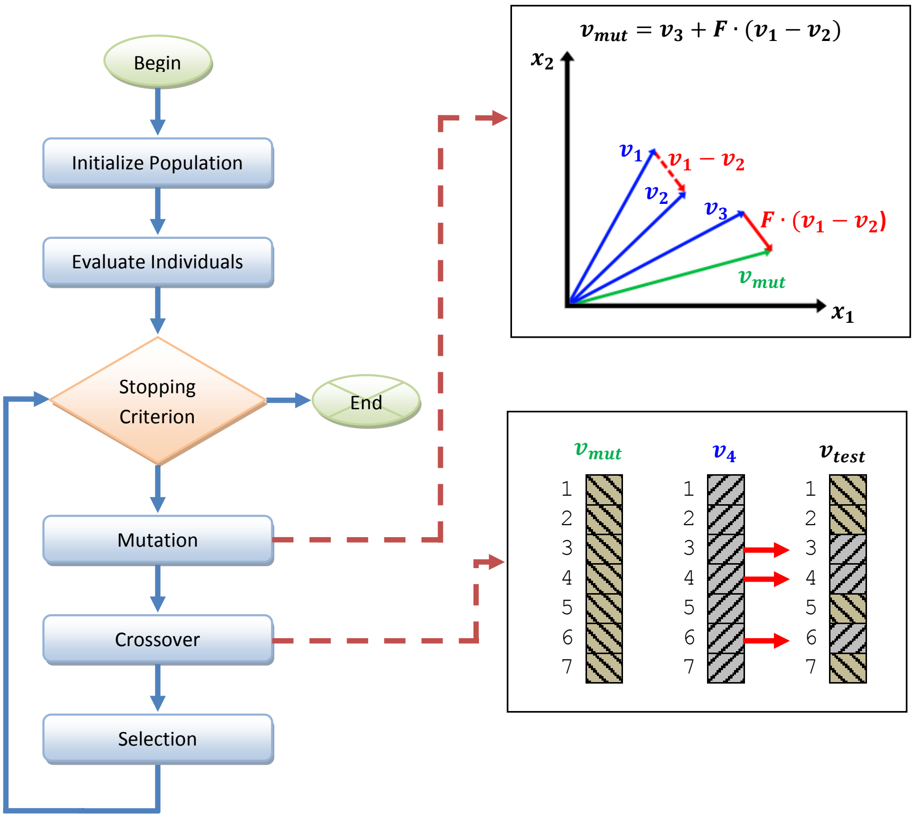


Figure 2: Flow Chart for Differential Evolution.

Differential Evolution is able to handle non-differentiable, nonlinear and multi-modal objective functions. These aspects are extremely beneficial in cases of complex physical problems, as is the case of the nuclear reactor core design. By the fact that stochastic perturbation can be made independently, the method allows parallelization and is considered efficient in dealing with objective functions of high computational cost. Differential Evolution is also easy to use, requiring an input of few configuration parameters, and it also has good convergence properties, i.e., good convergence to the global minimum in independent tests [7].

3.2 Invasive Weed Optimization

The key idea of the algorithm comes from evolutionary aspects of essential features of survival and evolution in nature. The method takes into account the behaviour of invasive weeds, which is a kind of plant with very peculiar features. Weeds are commonly recognized as plants that grows in undesirable locations and, because of its invasive behavior and intense capacity of reproduction, threatens agriculture. Considering that these plants demonstrate a great capacity of adaptation in different environments, their hability originated the computational algorithm, as described in details by [8]

Some specific parameters must be considered in order to successfully simulate the colonizing behavior of weeds: a limited search area is defined and a finite number of seeds is scattered over the explored field. Each of the seeds sprout a new plant, which in turn produce new seeds.The seeds will take its randomly defined location in the search field to cultivate new plants and this continuous process leads to increasingly fit weeds. A defined number of fitter members of the population naturally survive over time and produce new seeds, defining the cyclic process of survival and development of weeds. The algorithmic flow is depicted in Fig. 3.

The algorithm consists of four basic steps:

(1) Startup of the population:

A set of N_0 initial guesses, denoting the seeds, is randomly created and scattered over the d-dimensional search area, limited by the lower and upper bounds, represented by L_i and U_i, respectively, for $i = 1, \ \dots, \ d$. The

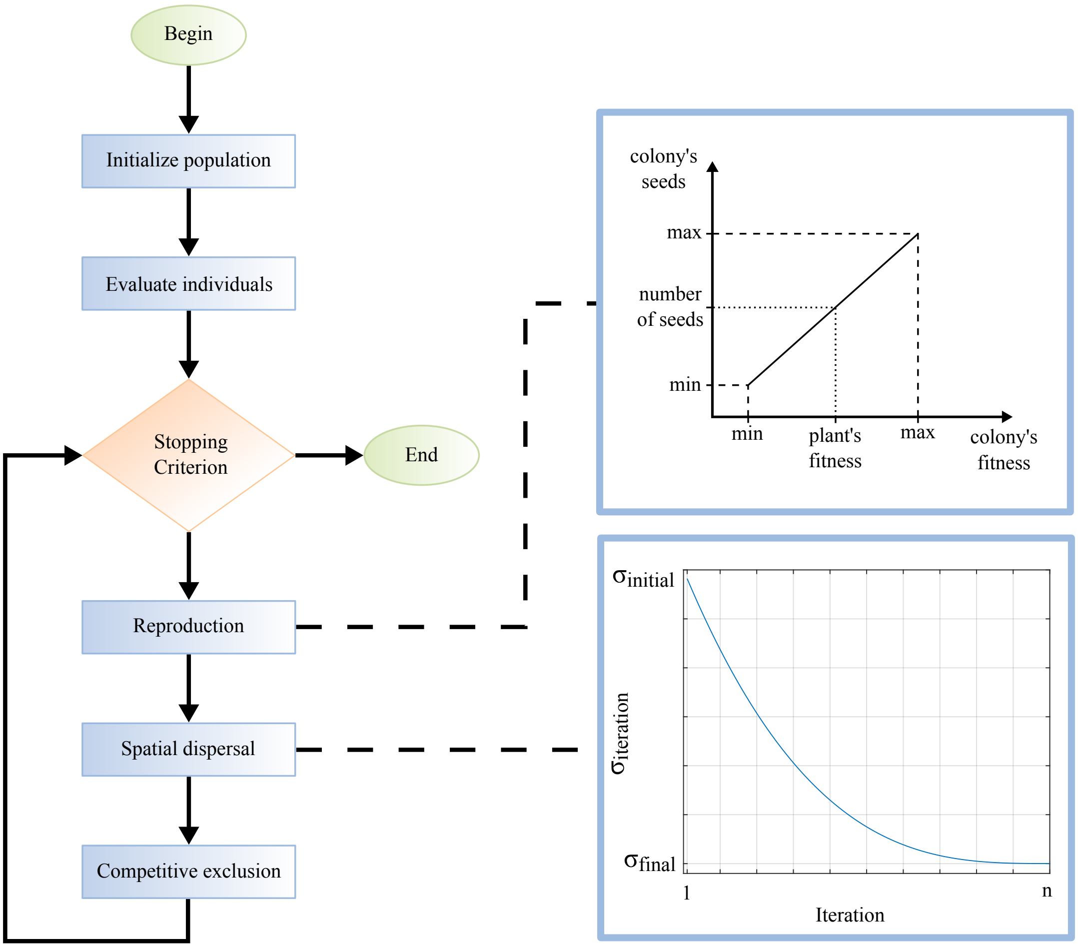


Figure 3: Flow Chart for Invasive Weed Optimization.

starting population of weeds is created by Eq. 12:

$$P_i^{(0)} = L_i + ((U_i - L_i)\, r_i) \tag{12}$$

where r_i is an uniformly distributed random number.

(2) Reproduction:

The objective function value of every element of the population is evaluated according to the objective function. This calculation points out the lowest and highest objective function value and defines the number of seeds each plant will be able to produce respecting the linear relationship among its own objective function value and the rest of the population. The best individuals in the population (the ones with more chance to survive) will produce more seeds and the number of new seeds will be limited.

This procedure provides the opportunity for less fitted individuals to produce new seeds, even in a smaller amount, the reason is that it is possible these individuals are located at an interesting position and they may contribute to the process of optimization.

(3) Spatial dispersal:

The new seeds must now be scattered. They must follow a normally distributed random dispersal over the d-dimensional search area with zero mean and variance calculated according to Eq. 13 which ensures the new seeds are going to be scattered close to the parent plant. Individuals with best objective function value will be able to produce more seeds and consequently they will have more new plants, sprout from its own seeds, in their neighborhood.

$$\sigma_{iteration} = \sigma_{\text{final}} + \frac{(it_{\max} - it)^{\eta}}{it_{\max}^{\eta}}(\sigma_{\text{initial}} - \sigma_{\text{final}}) \tag{13}$$

The nonlinear reduction of the variance according to nonlinear modulation index η, from σ_{initial} to σ_{final}, provides a good efficiency in spreading out the new seeds closer to the plants with higher objective function value, grouping fitter plants and isolating the inappropriate ones. This behavior favors the elimination of plants with low objective function value and performs a more refined search as the iterations increase and the plants get closer to the solution of the problem.

(4) Competitive exclusion:

After reproduction and spatial dispersal of new plants the colony must keep a maximum number of plants in the population, defined by the parameter p_{max}. A mechanism of elimination based on a kind of competition is necessary. When all produced seeds are located in their respective places they are ranked, plants and their offspring with better objective function value survive and gain a new opportunity to reproduce. Weeds with lower objective function value are eliminated.

3.3 Many-Objective Evolutionary Algorithm

Many-Objective evolutionary algorithm is a general name given to a class of optimization methods. The present work will explore one of this methods proposed by Zhenan He and Gary G. Yen [9], which is based on a strategy consisting of reducing the objective search space and improving the diversity of solutions. This new meta-heuristic optimization method is called by the authors as Many-Objective Evolutionary Algorithm based on objective search space Reduction and Diversity improvement (MaOEA-R&D) and it aims to overcome problems such as low diversity of solutions, a common problem of many multi-objective evolutionary algorithms [9].

As the name suggests, this algorithm has two stages and each stage has its own population, mutation and crossover rates. The mutation and crossover operators chosen by the authors to tackle the nuclear core design problem are the simulated binary crossover [10] and polynomial mutation [11].

The first stage (Fig. 4) reduces the objective search space by finding target points for each objective, those points are intended to be far from each other and achieve the extreme solution for each objective. In the process of finding those points, the algorithm classifies individuals (i.e. solution candidates for the problem) in subpopulations, generates offsprings for each subpopulation by randomly choosing two individuals and applying the crossover and mutation operators, then it ranks them by the Achievement Scalarizing Function (ASF). Finally, the target points are selected among all individuals based on the ASF for each objective.

The second stage (Fig. 5) improves diversity and updates the target points if any better solution is found. First a new population is generated around the target points then offsprings are generated from the population, these new individuals are classified as inside or outside some range acording to their objective function value. If there are enough individuals inside the range to update the whole population, the method sorts them by Pareto-dominance. Subsequently, individuals are selected from the non-dominated ones by a diversity operator. In case the algorithm does not get enough individuals, the population is completed with dominated ones with larger closest distance from non-dominated individuals. Also if there are not enough individuals inside the range the population is completed with individuals outside the range closest to the target points and their middle points. After the new population is selected, the target points and bounds are updated.

The diversity operator works by removing individuals that are closer to other individuals, it aims to keep the diversity and removes the redundancy of the population.

4 Execution

4.1 Differential Evolution

Differential Evolution was executed 10 times and, on each run, the same configuration parameters were used, as described in Table 2). Since it is established that the optimum value for the objective function is unknown, the stopping criterion was fixed as a maximum number of iterations.

These configuration parameters were established after several tests, and they differ from others previous use of Differential Evolution in the reactor core design problem, as showed in [5]. Table 3 shows the results for all 10 executions of the method. The parameters found are on the first columns, followed by their f_p and the number of necessary objective function (NFE) evaluations until the best result was obtained.

Analyzing Table 3, it is possible to see that good results were obtained for at least 70% of the runs, reaching an average objective function value as low as 1.2769. The best objective function value was obtained on two different runs for different parameters combinations which could indicate that the problem has several local minima on the possible solutions search region. In addition, it suggests that the problem has multiple global solutions, since the best calculated objective function value is exactly the same for runs 5 and 6. However, it will be shown that this is not the case.

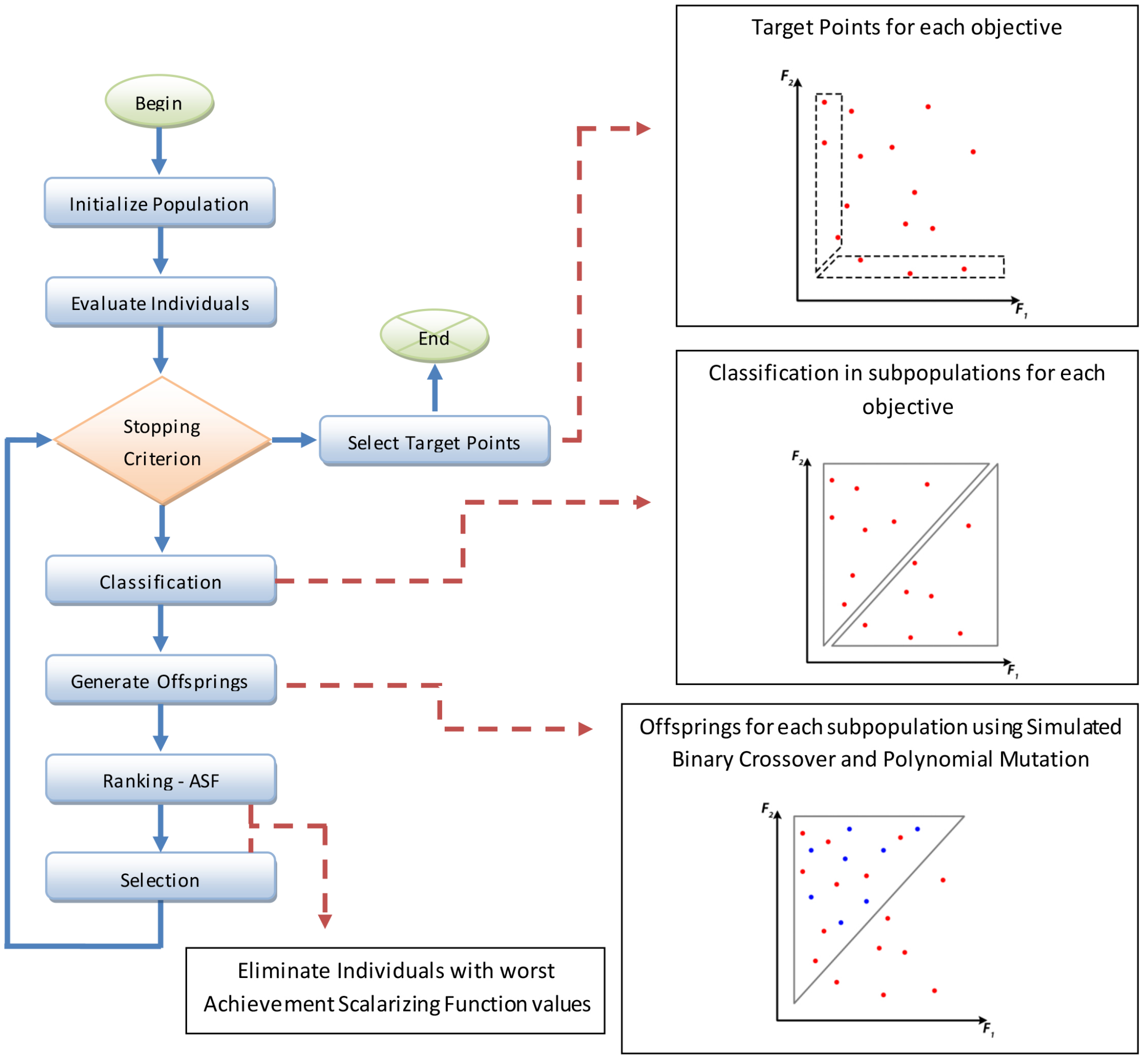


Figure 4: Flow chart for the first stage of MaOEA-R&D.

Table 2: Configuration parameters for Differential Evolution.

Parameter	Value
Population Size (NP)	3000
Mutation scaling factor (F)	0.5
Crossover rate (CR)	0.9
Maximum iterations	300

Using the methodology described on [12], it was possible to obtain confidence regions for the solution variables along the optimization process Fig. 6. The natural high number of objective function evaluations of Differencial Evolution can be used directly to generate regions around the estimated parameters, allowing a more detailed investigation of the search intervals. However, because the reactor design optimization does not correspond to a least squares minimization problem, it is impossible to consider statistically that these are confidence regions. It is only possible to say that the objective function varies by less than 2% inside the regions, turning all the encountered solutions very similar. It is possible to devise an idea of how the objective function behaves near a solution. The red dot in the Fig. 6 indicates the best solution found on the present work.

Analyzing the regions showed in Fig. 6, it can be noticed many correlations between some of the parameters. For

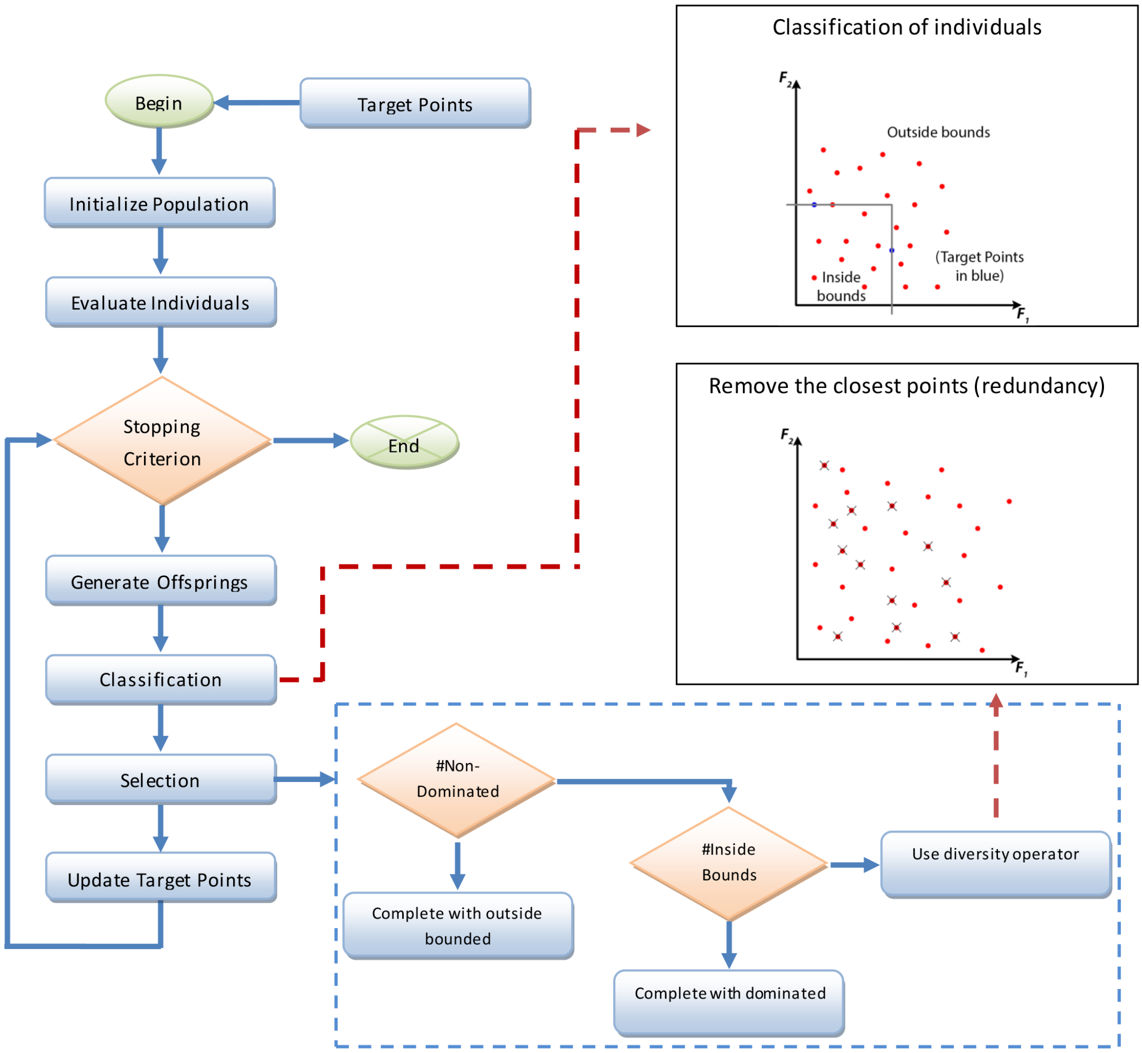


Figure 5: Flow chart for the second stage of MaOEA-R&D.

Table 3: Differential Evolution Results.

Run	$R_{fuel}(cm)$	$R_{clad}(cm)$	$R_{eq}(\%)$	$Enr_1(\%)$	$Enr_2(\%)$	$Enr_3(\%)$	M_{fuel}	M_{clad}	f_p	NFE
1	0.9525	0.1343	0.6963	2.2340	2.4325	4.4988	U-Metal	SS-304	1.2756	112.738
2	0.8418	0.0860	0.6378	2.0925	2.2720	4.1264	U-Metal	SS-304	1.2829	146.307
3	1.0081	0.1487	0.7450	2.3567	2.7746	4.7514	U-Metal	SS-304	1.2826	127.517
4	1.0628	0.1438	0.7620	2.3794	2.5731	5.000	U-Metal	SS-304	1.2748	151.039
5	0.8595	0.1108	0.6521	2.2094	2.3331	4.4609	U-Metal	SS-304	1.2724	164.161
6	1.0032	0.1122	0.7213	2.2248	2.3556	4.6287	U-Metal	SS-304	1.2724	297.914
7	0.9604	0.1140	0.6997	2.1614	2.3785	4.3237	U-Metal	SS-304	1.2839	165.827
8	1.1329	0.1006	0.7620	2.000	2.1397	4.1210	U-Metal	SS-304	1.2763	152.180
9	1.0102	0.1415	0.7377	2.3415	2.5104	4.8573	U-Metal	SS-304	1.2739	161.723
10	0.8690	0.1189	0.6534	2.1907	2.3217	4.3804	U-Metal	SS-304	1.2747	160.094
Average	0.9700	0.1211	0.7067	2.2190	2.4091	4.5149	U-Metal	SS-304	1.2769	153.950

example, a correct estimation of the parameters Enr_1, Enr_2, Enr_3 and R_{clad} highly depends on each other estimation. The parameter combinations showed in Fig. 6 have significantly high indications of correlations. However, R_{fuel} only correlate with R_{eq}. All other possible regions not shown in 6 present sparse shapes similar to Fig. 3h which means that they do not indicate the presence of correlation amongst the parameters. Applying the information from this correlation analysis may be very helpful for devising an optimization strategy or algorithm.

4.2 Invasive Weed Optimization

For the IWO 10 runs of the algorithm were executed so that a comparison of solutions was possible. All executions were carried out with the configuration parameters listed in the Table 4.

Table 4: Configuration parameters for Invasive Weed Optimization.

Parameter	Value
Number of initial population (N_0)	10
Maximum number of iterations ($it_{\max}$)	50
Maximum number of plant population ($p_{\max}$)	15
Maximum number of seeds ($s_{\max}$)	5
Minimum number of seeds ($s_{\min}$)	0
Nonlinear modulation index (η)	3
Initial value of standard deviation (σ_{initial})	1
Final value of standard deviation (σ_{final})	0.01
Initial search area (x_{ini})	Concerning to Table 1

The appropriate choice of the nonlinear modulation index (η) is essential in the convergence of the IWO. This coefficient makes de evolutionary behavior of the population of weeds changes through time. This is a desirable behavior, since it is expected that the algorithm carries out a wider search in the first few iterations and then the results are fine tunned as they aproach the maximum number of iterations ($it_{\max}$). Directly related to the choices of the initial and final value of standard deviation (σ_{initial} and σ_{final}, respectively), the nonlinear modulation index decreases the standard deviation of the iteration as smooth as possible. Its choice is based on the results presented in [8].

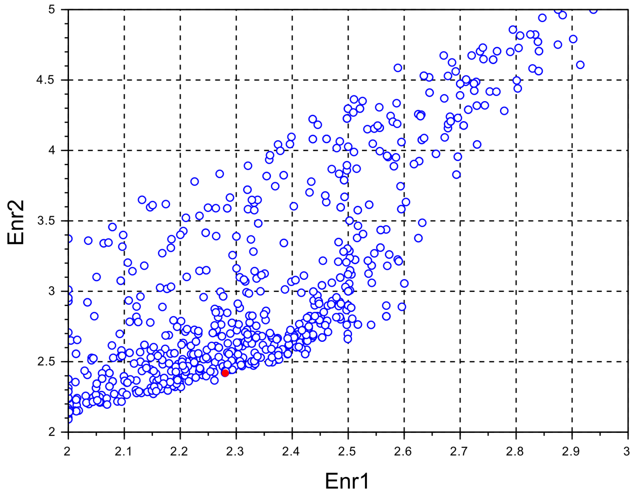
The number of initial population (N_0), as well as the minimum and maximum number of seeds (denoted by $s_{\min}$ and $s_{\max}$, respectively) are based on comparative tests for the problem and the results presented in [8]. A positive caracteristic of the algorithm is its rapid convergence, partly justified by the reduced number of individuals in the population and the limited rate of reproduction of new seeds. The increase in the value of these parameters must be evaluated together with the computational time due to the comparison between the individuals of the population in the competitive exclusion stage.

Due to the reduced range of each of the problem variables (as shown in Table 1), it was not necessary to conduct a very extensive exploratory search in the first few iterations, consequently the initial standard deviation (σ_{initial}) does not assume a very high value, which increases the chance of the population to stay in the limits for each of the variables of the model. This fact has accelerated the convergence of the algorithm given that less operations were necessary to generate new estimates for the new seeds of the population. The results for all 10 executions of the IWO are listed in Table 5.

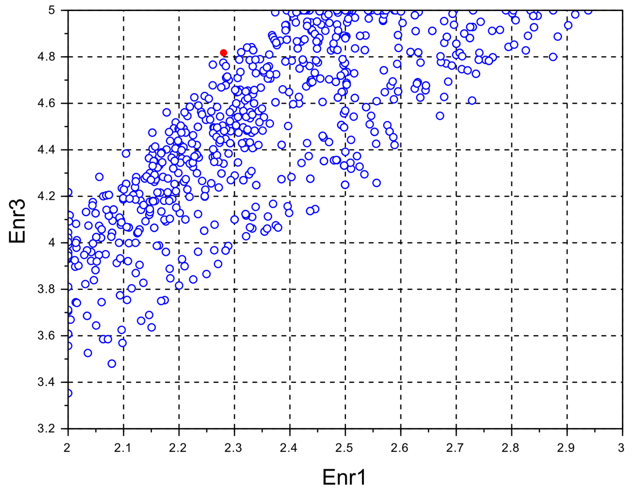
4.3 Many-Objective Evolutionary Algorithm

The parameters in table 6 were initialy set as equal for both phases of the algorithm. The size of the population and maximum iterations were chosen in order to keep the consistency with the IWO. The crossover rate (CR) and mutation rate (MR) were simple chosen as standard or "best practices" from the literature. The CR 1.0 means that every new solution generated contains part of two other solutions, and the MR 0.05 was the first choice for a fixed rate against the usually used $\frac{1}{NP}$. The mutation and crossover distribution index are the same used and suggested by the authors of the algorithm [9].

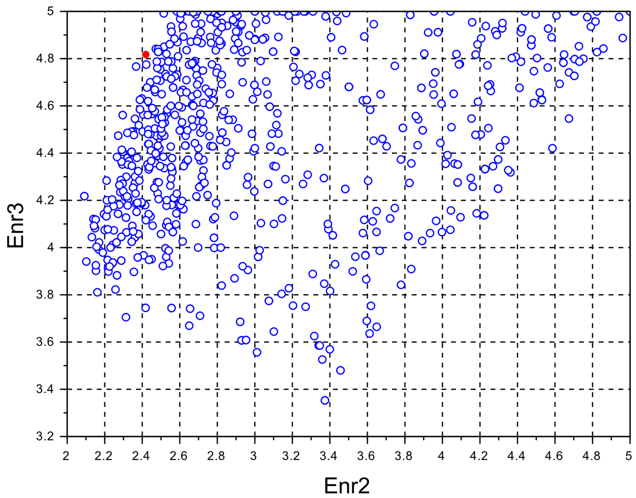
The algorithm has a modification in the diversity operator: an additional criterion for the removal of the individuals with closest distance. Since the distance between I_i to I_j is equal to the distance of I_j to I_i, any of those



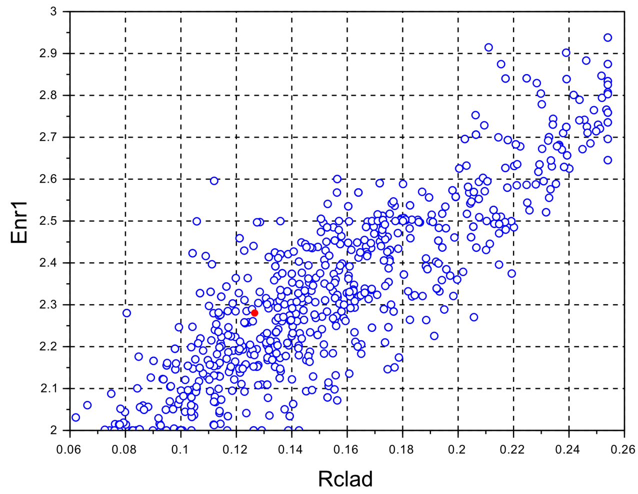
(a) Enr_1 xEnr_2



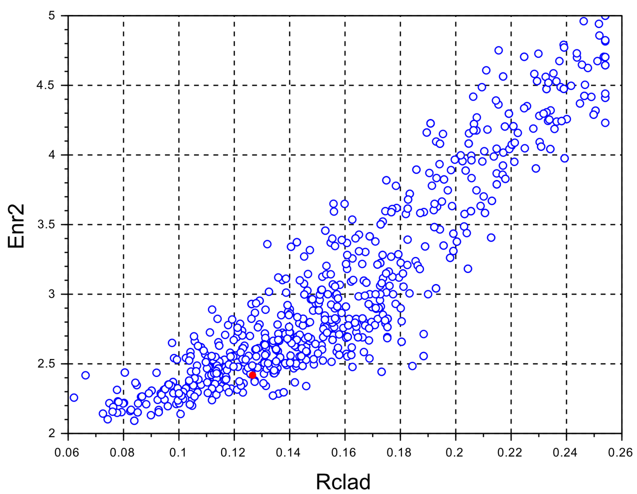
(b) $Enr_1 x Enr_3$



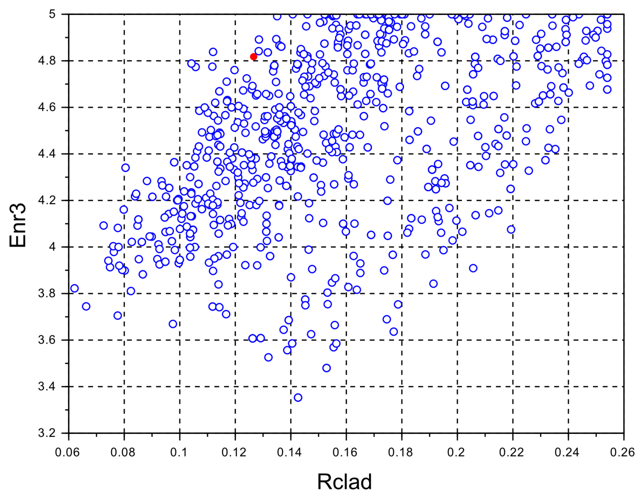
(c) Enr_2 x Enr_3



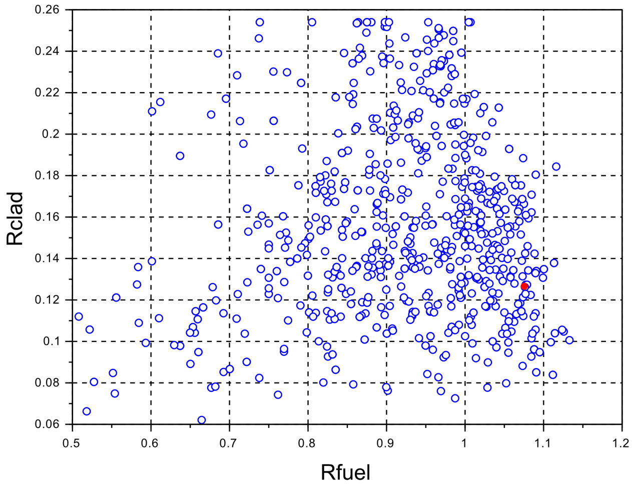
(d) R_{clad} x Enr_1



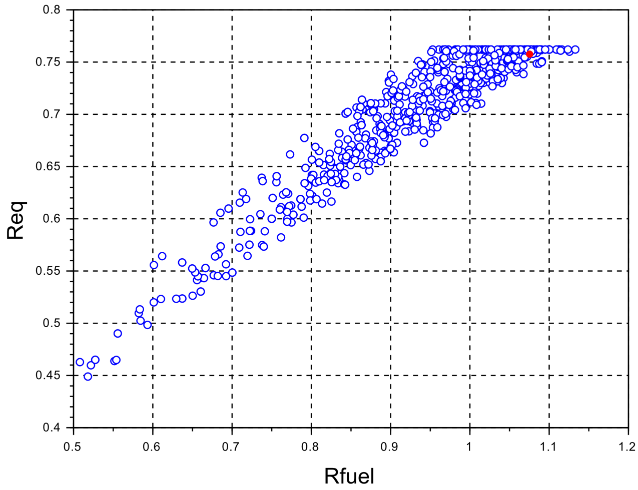
(e) $Rclad$ x Enr_2



(f) R_{clad} xEnr_3



(g) R_{fuel} x R_{clad}



(h) R_{fuel} x R_{eq}

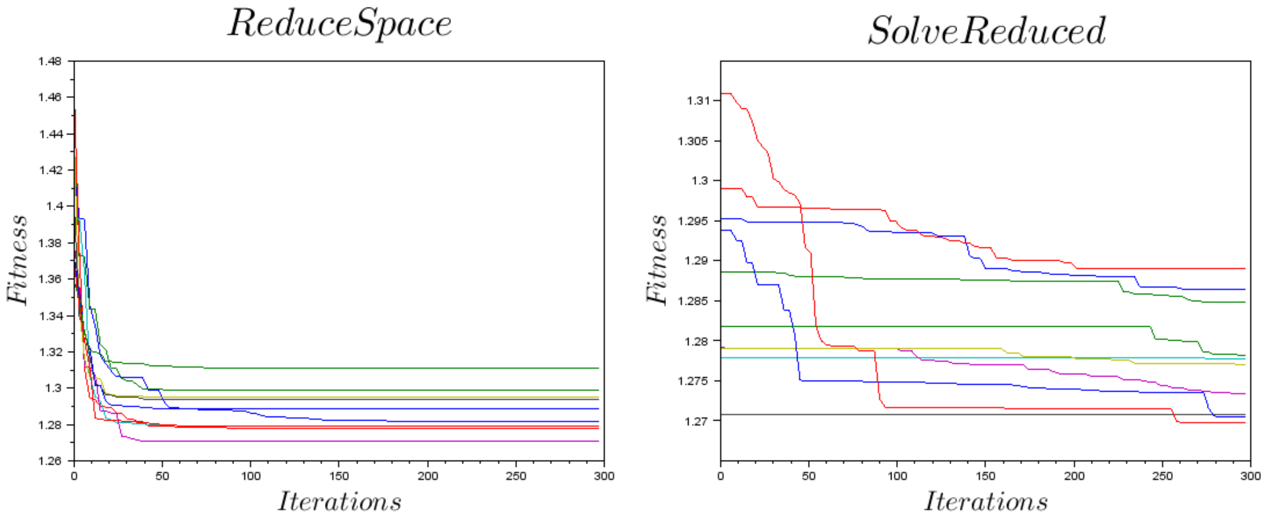
Figure 6: Regions of the problem space where the objective function varies by less than 2% around the best estimated parameters.

Table 5: Invasive Weed Optimization results.

Run	$R_{fuel}(cm)$	$R_{clad}(cm)$	$R_{eq}(\%)$	$Enr_1(\%)$	$Enr_2(\%)$	$Enr_3(\%)$	M_{fuel}	M_{clad}	f_p
1	0.694554	0.108361	0.569140	2.313048	2.602430	4.479661	U-Metal	SS-304	1.2845
2	0.901909	0.120486	0.670214	2.185427	2.359647	4.389187	U-Metal	SS-304	1.2745
3	0.883523	0.119866	0.671925	2.221883	2.623585	4.340446	U-Metal	SS-304	1.2885
4	0.588545	0.141780	0.504686	2.596281	2.816056	4.996377	U-Metal	SS-304	1.2770
5	1.077993	0.104849	0.743299	2.008470	2.312723	3.956182	U-Metal	SS-304	1.2876
6	0.950361	0.219712	0.734688	2.410124	4.121573	4.114011	U-Metal	SS-304	1.2879
7	0.959633	0.160277	0.705816	2.427750	2.706547	4.960727	U-Metal	SS-304	1.2774
8	0.941049	0.233515	0.727786	2.543370	4.214097	4.440953	U-Metal	SS-304	1.2894
9	1.072151	0.151212	0.758954	2.278674	2.502342	4.688076	U-Metal	SS-304	1.2739
10	0.795574	0.089516	0.614756	2.013403	2.348278	3.819981	U-Metal	SS-304	1.2860
Average	0.8865292	0.1449574	0.6701264	2.299843	2.8607278	4.4185601	U-Metal	SS-304	1.28267

Table 6: Configuration parameters for MOEA.

Parameter	Value
Population Size (NP)	3000
Mutation rate (MR)	0.05
Crossover rate (CR)	1.0
Mutation Distribution Index	20
Crossover Distribution Index	20
Maximum iterations	300



(a) Space Reduction Phase Convergence.

(b) Solve Reduced Phase Convergence.

Figure 7: Convergence of first and second phases of the Many-Objective Evolutionary Algorithm.

individuals can be removed. In the original algorithm the first is removed; here, the individual with worst objective function value is removed. This modification implies in keeping the best individuals in the population while the diversity is unchanged.

Table 7 shows the results of the first phase (reduce space) of the algorithm, the main objective of it is to find target points to the next phase. The highlighted line in Table 7 corresponds to the best result obtained in phase 1 of the method.

Table 7: First Phase - Many-Objective Evolutionary Algorithm Results.

Run	$R_{fuel}(cm)$	$R_{clad}(cm)$	$R_{eq}(\%)$	$Enr_1(\%)$	$Enr_2(\%)$	$Enr_3(\%)$	M_{fuel}	M_{clad}	f_p	Iteration
1	0.6955	0.1273	0.5906	2.1511	3.5983	3.5696	U-Metal	SS-304	1.2991	60
2	0.7377	0.1385	0.6068	2.3734	3.0462	4.4227	U-Metal	SS-304	1.2938	60
3	0.6009	0.1248	0.5308	2.5877	2.8790	4.9992	U-Metal	SS-304	1.2818	189
4	1.0454	0.2262	0.7335	2.4601	2.8601	4.9169	U-Metal	SS-304	1.3109	78
5	0.7913	0.1354	0.6214	2.2978	2.5802	4.4700	U-Metal	SS-304	1.2791	99
6	**1.0681**	**0.1619**	**0.7531**	**2.3826**	**2.5564**	**4.9998**	**U-Metal**	**SS-304**	**1.2708**	**87**
7	0.6429	0.2092	0.5712	2.8077	4.2341	4.8901	U-Metal	SS-304	1.2953	45
8	0.9422	0.2321	0.7543	2.6342	4.5862	4.5677	U-Metal	SS-304	1.2886	60
9	0.9002	0.1270	0.6886	2.3561	2.6111	4.7976	U-Metal	SS-304	1.2779	129
10	0.6620	0.1205	0.5589	2.4914	2.7247	4.8921	U-Metal	SS-304	1.2793	114
Average	0.8083	0.1607	0.6399	2.4503	3.1546	4.6460	U-Metal	SS-304	1.2882	94.5

Fig. 7a and Fig. 7b shows the progress of the best solution in each phase. In Fig. 7, each different color represents a different execution. For the first phase (reduce space) a number for maximum iterations between 50 and 100 was considered a good choice. On the other hand, for the second phase 300 iterations were not enough to get to a convergence. It indicates that more iterations can be used and probably better solutions can be found. The effort to improve diversity of solutions is the main reason the algorithm keeps evolving.

Table 8 presents the result for the second phase. The variability of the solutions found is noticeable, with the difference between the best and worst solution being almost 0.02 . However, the three best solutions found have a very similar objective function value, with run 3 and 6 having their parameters very close and the run 2 differing in

Table 8: Second Phase - Many-Objective Evolutionary Algorithm Results.

Run	$R_{fuel}(cm)$	$R_{clad}(cm)$	$R_{eq}(\%)$	$Enr_1(\%)$	$Enr_2(\%)$	$Enr_3(\%)$	M_{fuel}	M_{clad}	f_p	Iteration
1	0.7927	0.2114	0.6600	2.5009	4.2088	4.2137	U-Metal	SS-304	1.2891	204
2	0.8877	0.1581	0.6725	2.4210	2.5757	4.9494	U-Metal	SS-304	1.2705	285
3	0.5936	0.1254	0.5034	2.4323	2.6556	4.6449	U-Metal	SS-304	1.2782	297
4	**1.0478**	**0.1642**	**0.7373**	**2.3718**	**2.5239**	**4.9610**	**U-Metal**	**SS-304**	**1.2698**	**264**
5	0.8335	0.1463	0.6411	2.3410	2.5979	4.6325	U-Metal	SS-304	1.2771	294
6	1.0681	0.1619	0.7531	2.3827	2.5564	4.9999	U-Metal	SS-304	1.2708	3
7	0.9010	0.2533	0.7290	2.7275	4.7045	4.7056	U-Metal	SS-304	1.2865	285
8	1.0020	0.1399	0.7147	2.1505	2.5997	4.1785	U-Metal	SS-304	1.2849	276
9	0.8950	0.1273	0.6890	2.3568	2.6104	4.7907	U-Metal	SS-304	1.2777	279
10	0.7491	0.1243	0.6071	2.4227	2.5553	4.8716	U-Metal	SS-304	1.2734	294
Average	0.8771	0.1612	0.6707	2.4107	2.9588	4.6948	U-Metal	SS-304	1.2778	246

various parameters. The highlighted line in Table 8 corresponds to the best result obtained in phase 2 of the method.

5 Conclusions

In the present work, two new metaheuristics were applied to the design of nuclear reactor core problem. The nuclear design problem was initially described along with an attempt of qualitative assessment of the complexity of the problem. For this, the attainment of a set of graphs describing the typical behaviour of the objective function when close to a solution was performed by using a classic metaheuristic algorithm. In the present case, Differential Evolution was used. Invasive Weed Optimization, an metaheuristic based on a biological metaphor for plant adaptation, was then applied to the problem. It is shown that Invasive Weed Optimization results is able to attain a solution with reasonable sucess. Although the solution found by the method is not the best found so far, the method should still be considered for the present problem in view of its fast convergence and simpler configuration. Many Objective Evolutionary Algorithm on the other hand, as the second metaheuristic here applied to the problem, was able to found a solution that outperform previous solutions found by different methods, even considering that we limited to 300 the number of iterations of its convergence phase.

Acknowledgement

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

[1] T. O. M. Muniz and F. B. S. Oliveira, "Modelagem computacional de uma blindagem de nêutrons utilizando métodos determinísticos," *VETOR - Revista de Ciências Exatas e Engenharias*, vol. 30, no. 1, p. 15–27, 2020. Available at: https://doi.org/10.14295/vetor.v30i1.12877

[2] D. Rozon and M. Beauet, "Canada deuterium uranium reactor design optimization using three-dimensional generalized perturbation theory," *Nuclear Science and Engineering*, vol. 111, pp. 1–12, 1992. Available at: https://doi.org/10.13182/NSE92-A23919

[3] D. J. Kropackzek and P. J. Turinsky, "In-core nuclear fuel management optimization for pressurized water reactors utilizing simulated annealing," *Nuclear Technology*, vol. 95, p. 9, 1991. Available at: https://doi.org/10.13182/NT95-1-9

[4] J. Chapot, F. C. Silva, and R. Schirru, "A new approach to the use of genetic algorithms to solve the pressurized water reactor's fuel management optimization problem," *Annals of Nuclear Energy*, vol. 26, no. 7, p. 641, 1999. Available at: https://doi.org/10.1016/S0306-4549(98)00078-4

[5] W. Sacco, N. Henderson, A. Rios-Coelho, M. Ali, and C. Pereira, “Differential evolution algorithms applied to nuclear reactor core design,” *Annals of Nuclear Energy*, vol. 36, no. 8, pp. 1093–1099, 2009. Available at: https://doi.org/10.1016/j.anucene.2009.05.007

[6] J. Suich and H. Honec, “The hammer system heterogeneous analysis by multigroup methods of exponentials and reactors,” *Savannah River Laboratory*, 1967. Available at: https://doi.org/10.2172/4440604

[7] R. Storn and K. Price, “Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997. Available at: https://doi.org/10.1023/A:1008202821328

[8] A. R. Mehrabian and C. Lucas, “A novel numerical optimization algorithm inspired from weed colonization,” *Ecological informatics*, vol. 1, no. 4, pp. 355–366, 2006. Available at: https://doi.org/10.1016/j.ecoinf.2006.07.003

[9] Z. He and G. Yen, “Many-objective evolutionary algorithm: Objective space reduction + diversity improvement,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 145–160, 2016. Available at: https://doi.org/10.1109/TEVC.2015.2433266

[10] K. Deb and R. B. Agrawal, “Simulated binary crossover for continuous search space,” *Complex Systems*, vol. 9, no. 2, pp. 115–148, 1995. Available at: https://www.complex-systems.com/abstracts/v09_i02_a02/

[11] M. M. Raghuwanshi and O. Kakde, “Survey on multiobjective evolutionary and real coded genetic algorithms,” *Proceedings of the 8th Asia Pacific symposium on intelligent and evolutionary systems*, pp. 150–161, 2004. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.83.2396&rep=rep1&type=pdf

[12] E. Beale, “Confidence-regions in non-linear estimation,” *Journal of the Royal Statistical Society B-Statistical Methodology*, vol. 22, no. 1, pp. 41–88, 1960. Available at: http://www.jstor.org/stable/2983877