



**HAL**  
open science

## Surrogate-assisted asynchronous multiobjective algorithm for nuclear power plant operations

V. Drouet, Sébastien Verel, J.-M. Do

► **To cite this version:**

V. Drouet, Sébastien Verel, J.-M. Do. Surrogate-assisted asynchronous multiobjective algorithm for nuclear power plant operations. 2020 Genetic and Evolutionary Computation Conference (GECCO '20), Jul 2020, Cancún, Mexico. pp.1073-1081, 10.1145/3377930.3390206 . hal-03136995

**HAL Id: hal-03136995**

**<https://hal.science/hal-03136995>**

Submitted on 9 Sep 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Surrogate-Assisted Asynchronous Multiobjective Algorithm for Nuclear Power Plant Operations

V. Drouet

DEN - Service d'études des réacteurs  
et de mathématiques appliquées  
(SERMA)  
CEA, Université Paris-Saclayn  
Gif sur Yvette, France  
valentin.drouet@cea.fr

S. Verel

Univ. Littoral Côte d'Opale, UR 4491,  
LISIC, Laboratoire d'Informatique  
Signal et Image de la côte d'Opale  
F-62100 Calais, France  
verel@univ-littoral.fr

J.-M. Do

DEN - Service d'études des réacteurs  
et de mathématiques appliquées  
(SERMA)  
CEA, Université Paris-Saclayn  
Gif sur Yvette, France  
jean-michel.do@cea.fr

## ABSTRACT

In the context of the introduction of renewable energies in France, Nuclear Power Plant Operations (NPPO) are a key component for the compensation of the intermittent production of solar and wind power. In this work, we focus on the optimization of the operation cost and stability of power of a real-life power transient, while maintaining safety standards. From an optimization point of view, the NPPO problem is a typical example of a discrete constrained bi-objective problem based on time expensive computation simulation. We propose a massive asynchronous parallel master/workers MOEA/D assisted by a surrogate models. The algorithm design components are discussed and argued in this work. We show that our proposed surrogate assistance is able to improve algorithm performance and reliability, allowing us to extend our approach to a large range of strategic future real-life operations.

## CCS CONCEPTS

• **Applied computing** → **Engineering**; • **Computing methodologies** → *Massively parallel algorithms; Machine learning; Search methodologies;*

## KEYWORDS

Local search, Surrogate model/fitness approximation, Combinatorial optimization, Empirical study.

## 1 INTRODUCTION

Because of the increase of intermittent renewable energies combined with the shutting down of highly carbonated power plants, French agency RTE describes a future electricity market with very reduced margins, and strongly dependant on nuclear power plants [22]. French nuclear power plants (NPP) will therefore need to compensate for an important part of the intermittence, and the necessity of load-follow power transients will be increased.

New reactor designs such as the EPR [19] take into account this information and are designed to perform well in load follow operations. But most current French NPP are older designs, and even though they can and do perform load follow operations, they were not optimized for such transients. In this work, we show that it is possible to slightly alter the operating parameters of the NPP, in a way that both reduces the cost of the operation and increases power stability, while maintaining safety standards.

Optimizing reactor operations is not a new idea, be it on the French EPR [14] or on other reactors [3, 16, 24]. However, the common points of these works is the use of advanced control methods, designed with the help of analytical models of the reactor. In this work, the control method of the reactor is not altered, only the operating parameters changes, so that the change is completely transparent for the NPP operators.

This Nuclear Power Plant Operations (NPPO) problem is a typical example of a constrained bi-objective problem based on time expensive computer simulation. The search space is discrete, with around  $10^{12}$  possible solutions and a computation time of around 40 minutes per candidate solution. A large scale asynchronous master/workers MOEA/D (AMW-MOEA/D) was proposed in [9] as a first approach of the problem. Though this algorithm produced interesting solutions, its high variance and computation cost are a concern.

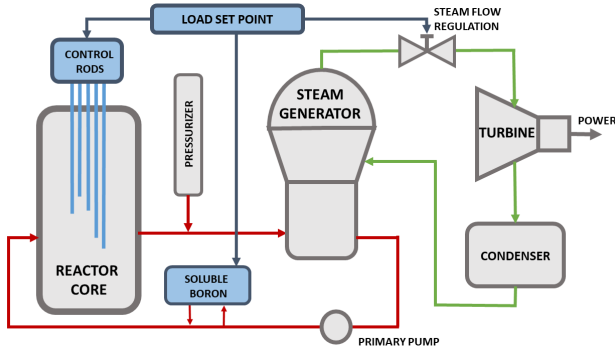
Beside parallel algorithms, surrogate models can also be used to face the issue of an expensive computation time of objectives evaluation [12]. In this approach, a cheaper surrogate model is learned during the optimization process, and replaces the expensive objective evaluation to guide more efficiently the search toward promising solutions. Such approach is a standard technique for continuous optimization problems such as the well-known Efficient Global Optimization (EGO) based on Gaussian Process [13]. A large number of surrogate models have been proposed for single or multiobjective continuous optimization (support vector regression, random forest, neural networks, radial basis functions, etc.) [2], but only few recent works deal with discrete optimization problems [1, 15, 25]. In multiobjective continuous optimization, all class of algorithms can be assisted by surrogate models (See. [6] for a review): Pareto based, indicator based, and more recently decomposition based approaches [4]. However, it is still an open issue to design efficient assisted surrogate model for discrete and multiobjective problems [23]. To our best knowledge, this work is the first attempt to design a surrogate assisted multiobjective algorithm for discrete problems which is parallel, asynchronous and able to scale up to thousands of computing units.

The first section of this paper describes the physics of the NPPO problem. The second section describes the proposed parallel asynchronous surrogate-assisted AMW-MOEA/D algorithm. The performance of this algorithm is discussed in the third section.

## 2 NUCLEAR POWER PLANT OPERATIONS

### 2.1 Description of the French 1300MW Pressurized Water Reactor

The reactor considered in this work is the French 1300 MW pressurized water reactor (PWR). Among the French nuclear fleet, it is both one of the most common, after the 900MW PWR, and the one that presents the best potential of improvement for load follow operations, due to its operating mode: the G mode. The results of this work could easily be extended to the 900MW PWR that are operated in the G mode.



**Figure 1: Simplified schematic view of the 1300MW pressurized water reactor. The red lines represent the primary loop, the green lines the secondary loop, and the blue lines the control of the reactor.**

Fig. 1 shows a simplified view of the reactor. The nuclear power is produced by a chain reaction of neutron induced fissions in the reactor core. A first liquid water loop (in red on Fig. 1), pressurized at 155 bar, evacuates the power produced in the core to the steam generator, where it heats a secondary loop (in green on Fig. 1). The steam produced in the secondary loop is then used to actuate the turbine and alternator to produce power. The remaining steam is then cooled by the condenser, and fed back to the steam generator. The input of this reactor is the load, which means the electric power produced at the output of the alternator.

For small power variations, the physics of the neutrons in the PWR makes the system self regulated, and no operation is needed. But for more important variations, action is required to ensure that the reactor core will remain in its desired temperature state. Moreover, the operators have to control the power distribution in the core. The core is a heterogeneous 4 meter high and 3 meter wide cylinder, in which the power production is not homogeneous, both because of the presence of control rods, and neutron poisons that are produced by fission reactions. The operator has to make sure the power distribution stays close to the initial distribution. The control value for the distribution of power is called the axial offset and is defined as follows:

$$AO = \frac{P_{upper} - P_{lower}}{P_{upper} + P_{lower}} \quad (1)$$

where  $P_{upper}$  is the power produced in the upper part of the core, and  $P_{lower}$  is the power produced in the lower part of the core.

The plant operators have two ways of controlling the reactor:

- Control rods, made of neutron absorbant materials, are inserted in the core to slow down the chain reaction. There are two types of rods, depending on the importance of their effect of neutrons: grey rods, that are less absorbant, and black rods, that are more absorbant;
- Soluble boron is inserted in the water of the primary loop, to absorb neutrons and slow down the chain reaction.

The effect of the boron is homogeneous in the core, whereas the rods are inserted from the top of the core, and therefore disturb the power distribution in the core.

In the G mode, the rods are separated in two sets:

- The Power Shimming Rods (PSR), made of 4 groups (G1, G2, N1 and N2). The first two groups are made of grey rods, and the last ones of black rods. They have to compensate the reactivity effects of a change in power. They are used in an open loop indexed on the electric power generated by the plant.
- The Temperature Regulation Rods (TRR), made of black rods, are used to further regulate temperature. They compensate for defaults in the calibration of the PSR, and the apparition of neutron poisons such as xenon. They move in a limited maneuvering band  $m_{TRR}$  in the upper part of the core.

The different groups of the PSR move with constant overlaps. At first, G1 moves alone. Then after a certain amount of steps called the overlap between G1 and G2, G2 will start to move and the two rods will move together until G1 is completely inserted. The movement of the PSR are thus defined by three overlaps:  $o_{G1/G2}$ ,  $o_{G2/N1}$  and  $o_{N1/N2}$ . The rods each have a maximum speed  $s$ . The movement of the rods are measured in steps which are roughly equal to 1.6 cm.

The rods are moved automatically depending on the power generated by the plant and the gap between the temperature of the core and its set value. However, these rods cannot compensate for neutron poisons during a power transient. Therefore the operator has to inject soluble boron in the primary loop to help control the core temperature. Furthermore, the operators use the boron to compensate power distribution imbalances due to the movement of the rods, in order to maintain the axial offset close to its set value.

### 2.2 Power transient

The considered power transient is a day/night load follow operation (Fig 2). The power is reduced to 30% of the nominal power for 6 hours, then goes back to the nominal power. The maximum authorized slope of 5% of nominal power per minute is used, making it a very difficult transient for the operators.

### 2.3 Model of the reactor

The simulator used for this project is a multi-physics PWR model encompassing both the primary and secondary loops [8][18]. The computations are performed thanks to APOLLO3®[10], the multi purpose deterministic transport code developed at CEA institute. The neutronics model is based on a static 3D flux calculations based

on the diffusion approximation for the power distribution calculation, and a point kinetics model for the core power calculation. This model takes into account Doppler and moderator feedback effects through the use of a simplified thermal-hydraulics model computed in APOLLO3®. The evolution of actinides and several poisons, including xenon, iodine and samarium is computed at each time step. This neutronics model is coupled to a 0D model of the steam generator, so that the simulator can take into account the imbalance between the extracted steam flow and the produced steam flow during load-follow operations. A simplified model of the G mode is used in order to reproduce rod movements and boron concentration evolutions[8]. The mean computing time for the transient is around 9.5 minutes, with a standard deviation of about one minute.

## 2.4 Values of interest and input parameters

Industrial concerns related to load-follow operations are cost, stability and safety. Evaluating all costs related to power transients is a difficult tasks. It includes direct costs of operations, such as the volume of effluents. This volume is the volume of water effluents from the primary loop that were generated in order to change the boron concentration in the core. These effluents are costly to process because they contain activated nucleides. The costs also include indirect costs, such as the wear of the rods, or accelerated aging of the materials. Finally, some others costs are induced by the reduced load factor of the plant. In this study, and because of the time scale considered, we focus on direct costs.

Stability of power in the reactor is a concern during power transients because of neutron poisons. Xenon and Iodine appear in the reactor during a power transient, and can cause spatial oscillations of the flux. These variations are a safety concern, and can prevent the reactor from undergoing a second transient for a few hours or even days after the first one. Several values of interests can be computed during a transient to measure instability, in this study the gap between the set axial offset and effective axial offset, integrated over time, is considered.

Finally, there are safety concerns associated to power transients. When a lot of rods are inserted in the core, the worry is that the safety margin for an emergency stop, during which all rods are dropped in the core, is not enough. Another concern, because of

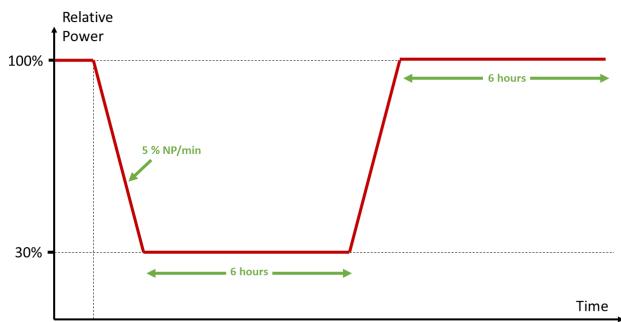


Figure 2: Power transient.

the disruptions of the flux, is that locally, the power will be too high and go over the authorized limit, causing an alarm, and reactor shutdown. A random sampling of the search space showed these two constraints were rare, with less than 1 % occurrence. Therefore, no repair or penalty strategies are implemented to deal with constraints, the solutions are simply rejected when they don't respect one or the other.

We therefore consider 4 values of interests, two objectives and two constraints. The volume of effluents  $v$ , the axial offset integral  $i$ , the safety margin, and maximum local power.

The physical behaviour of the reactor changes during its exploitation cycle. French 1300MW PWR are operated with the GEMMES management, which means their fuel is reloaded every 18 months. During these 18 months, the depletion of fuel causes changes in the flux, and therefore in the objectives. The depletion of the flux is measured by the burnup ( $bu$ ), which is the total energy received by a ton of fuel. A random sampling of the search space showed that there were no strong correlations between the different moments of the cycle for the considered objectives. It is therefore necessary to consider multiple burnup points during the optimization process. Four burnup points were considered: beginning of the cycle (BOC), one quarter of the cycle (QOC), middle of the cycle (MOC) and three quarters of the cycle (TQOC). The end of the cycle is not considered, since the fuel is too depleted to allow load follow operations.

For each calculation, the volume of effluents and axial offset integral are computed for the four burnup points. If for any burnup, one constraints is not respected, the candidate solution is rejected. The values computed are then used to compute the two values of interest for the complete cycle:

$$I = \max_{bu} \{i(bu)\} \quad (2)$$

$$V = \frac{1}{\sum_{bu} 1/v_{ref}(bu)} \sum_{bu} v(bu)/v_{ref}(bu) \quad (3)$$

$I$  is computed with the values of the axial offset integrals. The maximum imbalance is considered, since imbalance is a safety concern.  $V$  is computed as the mean value of the volumes of effluent at each burnup. The weights used are the reference volume computed with the current rod management. This is motivated by the fact that the volume of effluents increases with the burnup, because the base concentration of boron decreases. It is therefore more costly to do a power transient near the end of the cycle, meaning that the operator will favor the beginning of the cycle. Without the weights, the TQOC volume of effluents would have too strong an impact on  $V$ . The multiple-burnup problem is therefore a two objective problem, with two constraints. Since 4 calculations are needed to compute one candidate solution, the average computing time is 40 minutes. The objectives are normalized using standard normalization, using the mean and standard deviation computed during the fitness landscape analysis [8].

The goal of this study is to slightly modify the plant to improve performance without disrupting it so much that the safety demonstration would be completely different. Therefore, the G mode will not be modified in itself. Only the rod movement parameters will be modified: the overlaps between the PSR groups, their maximum

speed, and the maneuvering band of the TRR<sup>1</sup>. Their range of possible variation is determined by technological constraints. They can only take discrete values in steps because of technological constraints. The parameters and their ranges of variations are summed up in table 1.

**Table 1: List of all inputs**

Input	Min	Max	Current	Unit
$o_{G1/G2}$	0	255	185	steps
$o_{G2/N1}$	0	255	175	steps
$o_{N1/N2}$	0	255	160	steps
$s_{G1}$	20	110	60	steps per minute
$s_{G2}$	20	110	60	steps per minute
$m_{TRR}$	8	117	27	steps

Therefore, the search space is a discrete 6 dimension space with a size of around  $10^{12}$ . In summary, the real-world NPPO problem is an expensive discrete biobjective optimization problem with constraints.

### 3 SURROGATE-ASSISTED AMW-MOEA/D

#### 3.1 Definition

A multiobjective problem (MOP) aims at optimizing jointly multiple values of interest. These objectives often being antagonist, the answer is a set of trade-offs solutions between values of interests. Let  $\Omega$  be the search space of dimension  $d$  and  $f : \Omega \rightarrow \mathbb{R}^m$  the function to optimize, with  $m$  scalar objectives. A solution  $x \in \Omega$  is said to dominate a solution  $x'$  if for all  $i \in \{1, \dots, m\}$ ,  $f_i(x) \leq f_i(x')$  and there exists  $j \in \{1, \dots, m\}$  such that  $f_j(x) < f_j(x')$ . A solution  $x \in \Omega$  is non dominated if there does not exist a solution in  $\Omega$  that dominates  $x$ . The goal of a multiobjective optimization process is to find all non dominated solutions. This set is called the Pareto set, and its image by  $f$  in the objective space is the Pareto front.

MOP strategies based on decomposition consist in splitting the  $m$ -objective problem in a number of mono-objective sub-problems, that will be solved separately or cooperatively. In the case of MOEA/D [26], the objective space is split in  $\lambda$  directions using weight vectors  $w \in \mathbb{R}^m$ . A scalarizing function  $g(\cdot|w, z^*)$  is then used to transform the objectives  $f$  into a scalar value. Common scalarizations [7] include the Chebyshev function:

$$g(z|w, z^*) = \max(w_1 \cdot |z_1 - z_1^*|, w_2 \cdot |z_2 - z_2^*|) \quad (4)$$

or the weighted sum:

$$g(z|w, z^*) = w_1 \cdot |z_1 - z_1^*| + w_2 \cdot |z_2 - z_2^*| \quad (5)$$

where  $z^*$  is a reference point towards which the algorithm is guided to progress.

Each direction solves its mono-objective problem using an (1+1)-Evolution Strategy without cross-over. For each direction, the algorithm keeps in memory the best solution. This solution is mutated as described in the next section, in order to explore the search space and find a better solution.

<sup>1</sup>The maximum speeds of the N1 and N2 rods are not considered, since they are almost never reached.

However, directions does not work alone. We define a neighbourhood  $\mathcal{N}(D)$  for each direction  $D$ . When a solution is proposed for the direction  $D$ , it is also sent to all directions of the neighbourhood, the idea being that if a solution is good for one direction, it is likely to be interesting too on neighbourhood directions, if the weight vectors are distributed regularly. Notice that there is no cross-over in this version of MOEA/D following previous analysis [18]. For the 2-objective case, we split the objective space with constant angles:

$$w_i = \left( \cos\left(\frac{i\pi}{2\lambda}\right), \sin\left(\frac{i\pi}{2\lambda}\right) \right) \quad \text{for } i = 0, \dots, \lambda - 1 \quad (6)$$

The neighbourhood is defined by the parameter  $T$ . For a direction  $D_i$ , the neighbourhood is then:

$$\mathcal{N}(D_i) = \{D_{i-T}, \dots, D_{i+T}\} \cap \{D_0, \dots, D_{\lambda-1}\} \quad (7)$$

#### 3.2 Mutation operator

The mutation operator is based on two parameters:

- the mutation rate vector  $p \in [0, 1]^d$  which defines, for each input, the probability that it will be mutated ;
- the mutation range vector  $r \in [0, 1]^d$  which defines the range in which the inputs will be mutated.

First, following standard mutation of EA, the operator selects the inputs that will be mutated, according to each input's rate mutation  $p_i$ : each input  $i$  is mutated following Bernoulli law of parameter  $p_i$ . The vector  $p$  is chosen such that the expected value of the number of mutated inputs will be 1. If  $p$  is uniform, then the number of mutated inputs follows a Binomial distribution, with an expected value  $p_0 d$ , therefore  $p_i$  is set to  $1/d$ .

For each selected input, the new value is selected according to a uniform probability law in the set:

$$\{x_i - \lfloor r_i (M_i - m_i) \rfloor, \dots, x_i + \lfloor r_i (M_i - m_i) \rfloor\} \cap \{m_i, \dots, M_i\} \setminus \{x_i\} \quad (8)$$

where  $i$  is the selected input,  $x_i$  its current value,  $m_i$  and  $M_i$  respectively its lower and upper bound.

A hashing table is used so that no solution is computed twice. The mutation operator is therefore applied until it produces a solution that is not present in the table.

#### 3.3 Surrogate strategy

In order to improve the search, the proposed algorithm is assisted with a surrogate model (See Alg. 1). Two surrogate functions are considered, for the objectives  $V$  and  $I$ . Let  $M$  be the surrogate model that generates the estimated objective vector. For one direction  $D$ , the fitness of solution  $x$  on direction  $D$  can then be estimated as  $g(M(x)|w_D, z^*)$ . The surrogate model is trained at each iteration of the AMW-MOEA/D algorithm, using all previously computed solutions.

The selection of a new solution uses a  $(1+\lambda_M)$ -Evolution Strategy (ES) with a population of  $\lambda_M$  solutions, that is run for  $G$  generations (See Alg. 2). Starting with the current best solution for the direction considered,  $\lambda_M$  mutants are iteratively generated, evaluated with the surrogate and selected.

In order to acquire enough points so that the prediction of the surrogate model is accurate, the surrogate assisted selection is only

used after a certain amount of points,  $N_{start}$ , have been evaluated. Before that, the next point is simply a mutant of the current best.

If  $G = 1$ , this is a filter [4], that does a first screening of the mutants using the surrogate before choosing the next point. As the number of evaluated solutions  $G\lambda_M$  increases towards the size of the search space, this is a full internal optimization process based on the surrogate. In [4], it was suggested that this type of setup performs better than a simple filter.

A detailed analysis of several surrogate models for the NPPO problem is performed in Sec. 4.2.

### 3.4 Parallel design

The algorithm is designed for massively parallel use (several thousands of computing units), with a master/workers architecture. It allows for a centralization data, and input/output operations, without overloading the master node, because of the long computing time. With a thousand computing units and an average computing time of 40 minutes, the master node has an average time of 2 seconds to treat an incoming message, which seems large enough for the simple operations of adding a solution to the results file, updating the best solutions of the neighbourhood directions, and selecting a new solution for evaluation.

Because of the important variance of the computing time and the large number of computing units, the algorithm works in a completely asynchronous manner [11]. This means that when a calculation ends on one processor, its results is directly treated by the master node, that sends a new solution for evaluation without waiting for the full generation to be computed.

The pseudo-code of the AMW-MOEA/D is given in Alg. 1, Alg. 2, and Alg. 3. The initial population is generated using Sobol sequences. At each iteration, the reference point  $z^*$  is updated so that its coordinates correspond to the minimum known value for each objective. The stopping condition is a determined computation time. From previous works [18], the Pareto front seems to be globally balance between objectives, so the directions are equally distributed among computing units at initialization, and are not changed during the execution.

The main parameters of the algorithm, excluding the surrogate parameters, are therefore:

- the number of directions  $\lambda$ ;
- the number of computing units  $N_{proc}$ ;
- the scalarization function;
- the normalization function for the objectives;
- the neighbourhood size  $T$ .

Previous work by Muniglia [18] found that  $\lambda = 200$  directions are enough to capture this problem, and that the Chevishev scalarization is better suited to ensure a variety of solutions.

## 4 EXPERIMENTAL ANALYSIS

### 4.1 Experimental design

Both the surrogate assisted AMW-MOEA/D and the non assisted algorithm were tested on 10 runs of 10 hours each, on 1008 CPU on the TGCC (SKL Irene on Intel Xeon 8168 2.7GHz CPU units) Two sets of parameters were considered for the surrogate assisted algorithm: one in which a simple filter is used ( $G = 1, \lambda_M = 150$ ) and

---

### Algorithm 1: AMW-MOEA/D: master process

---

```

1 for  $i \leftarrow 1$  to  $N_{proc}$  do
2    $D_i \leftarrow$  Attribute direction;
3    $x_i \leftarrow$  Initialize using Sobol numbers;
4   Send  $x_i$  to process  $i$ ; // Non blocking
   communication
5 end
6  $z^* \leftarrow$  Initialize reference point;
7  $f_D^* \leftarrow$  Initialize best fit for each direction  $D$ ;
8  $x_D^* \leftarrow$  Initialize best solution for each direction  $D$ ;
9  $S \leftarrow \emptyset$ ; // Set of evaluated solutions
10 while time left do
11   Receive Msg from process  $i$  w.r.t. direction  $D_i$ ;
12    $(f, x) \leftarrow$  Msg; // Receive obj. vector and
   solution
13   Normalize  $f$ ; // See 2.4
14    $S \leftarrow S \cup \{(x, f)\}$ ;
15   for  $k \leftarrow 1$  to  $d$  do
16     if  $f_k < z_k^*$  then
17        $z_k^* \leftarrow f_k$ ; // Update reference point
18     end
19   end
20   for  $D \in \mathcal{N}(D_i)$  do
21     if  $g(f|_{w_D}, z^*) < g(f_D^*|_{w_D}, z^*)$  then
22        $x_D^* \leftarrow x$ ; // Update best known solution
23        $f_D^* \leftarrow f$ ;
24     end
25   end
26   Train surrogate models  $M$  with  $S$ ;
27   if  $\#S < N_{start}$  then
28      $x' \leftarrow$  Mutate  $x_{D_i}^*$ ;
29   else
30     Select  $x'$  using surrogate models  $M$ ; // See Alg.
   2
31   end
32   Send  $x'$  to process  $i$ ; // Non blocking
   communication
33 end

```

---

one with a complete evolutionary strategy ( $G = 10, \lambda_M = 15$ ). In both cases, 150 solutions are evaluated with the surrogate model at each step, which is roughly the total amount of solutions evaluated per direction by the non assisted algorithm.

### 4.2 Surrogate models

Different surrogate models can be used for discrete optimization problems. In Gaussian Process [25], the fitness of solutions is supposed to follow a normal distribution characterized by a co-variance matrix between solutions. Although this model is able to estimate an error of the model, for a massive master/workers parallel system, the computation time on the master process is too long to scale up to thousand units. Another alternative is Polynomial Chaos (PC)

---

**Algorithm 2:** ES:  $(1 + \lambda_M)$ -Evolution Strategy to select a candidate solution for direction  $D$  using surrogate model  $M$  model

---

```

1  $x^* \leftarrow x_D^*$ ;
2  $g^* \leftarrow +\infty$ ;
3  $j \leftarrow 1$ ;
4 while  $j \leq G$  do
5    $\{y_k\} \leftarrow$  Generate  $\lambda_M$  mutants of  $x^*$ ;
6   Evaluate all solutions  $y_k$  with  $g(M(\cdot)|w_D, z^*)$ ;
7    $y^* \leftarrow \operatorname{argmin}_{\{y_i\}} g(M(\cdot)|w_D, z^*)$ ;
8   if  $g(M(y^*)|w_D, z^*) < g^*$  then
9      $x^* \leftarrow y^*$ ;
10     $g^* \leftarrow g(M(y^*)|w_D, z^*)$ ;
11  end
12   $j \leftarrow j + 1$ ;
13 end
14 return  $x$ ;

```

---

**Algorithm 3:** AMW-MOEA/D: worker process

---

```

1 while time left do
2   Receive Msg from master process;      // Blocking
   communication
3    $x \leftarrow$  Msg;
4    $f \leftarrow$  Evaluate  $x$ ;
5   Send  $f$  to master process;          // Blocking
   communication
6 end

```

---

approach. The model is an orthogonal basis of polynomial functions as such Legendre basis of polynomials [27]. The coefficients are estimated with sparse linear regression methods (LASSO in this work). Random Forest (RF) can also be used as surrogate model for discrete or mixed optimization problems (discrete and continuous variables) [21]. RF is an efficient ensemble surrogate based on decision tree weak learners. Each tree is learned at the same time with bagging technique of sample and random sub-sampling of predictors. Gradient Boosting (GB) [5] is also another ensemble technique based on weak learners (also decision trees in our case). On the contrary of RF, in gradient boosting the weak learners are trained sequentially based the error of previous weak learners. We also analyze the Support Vector Regression (SVR) model [17] which is generalization of linear model using the kernel trick to face non-linearity.

To select a surrogate model for the NPPO problem, an offline analysis is performed, based on the data from the non assisted runs. All methods are implemented using the Python library Scikit-learn [20]. For each of these runs, at several points of the run, the surrogate models are built using the already computed points. The coefficient of determination of the regression is computed on the remaining points as a test set in order to reflect the problem: will the surrogate be accurate on the further evaluated points? Fig. 3 shows the results of this analysis for both objectives. Apart from

**Table 2: Mean hypervolume improvement between 5000 evaluations and the end of the run. The p-value is the result of the comparison with the non assisted configuration with a Welsch t-test.**

Configuration	Hypervolume Improvement	p-value
Non assisted	1.32	-
$(G = 1, \lambda_M = 150)$	2.07	0.19
$(G = 10, \lambda_M = 15)$	3.08	0.01

the linear regression, all methods converge towards a good value of the coefficient of determination  $R^2$ . However, it appears that random forest and in a smaller way gradient boosting perform better for a smaller amount of computed solutions. These are both methods based on decision trees, which might be more adapted to this problem because some of the inputs are thresholds, that generate slope discontinuities in the objectives. For all methods, the coefficient of determination remain lower for the volume of effluents than the axial imbalance, suggesting it is harder to model.

The random forest training time is under 0.4 seconds, even for large training sets, making it fast enough for the massively parallel configuration. It is therefore selected for the surrogate assisted AMW-MOEA/D. The  $N_{start}$  parameter is set to 5000 evaluated solutions so that both  $R^2$  coefficients are above 0.8 when the assistance starts.

### 4.3 Performance analysis

Fig. 4 and 5 show the comparison of the assisted and non assisted algorithm, with the two sets of parameters ( $G = 1, \lambda_M = 150$ ) and ( $G = 10, \lambda_M = 15$ ). The  $N_{start}$  parameter is set to 5000 for all surrogate assisted runs, therefore the average behaviour of the algorithm should be completely similar before 5000 evaluations, except from the variance due to the random process of the initialization phase and the non assisted phase. The difference of final hypervolume is not statistically significant, however the slope seems to be higher after  $n = 7000$  for the assisted runs.

In order to eliminate the bias of the variance of the algorithm during the initialization and non assisted phases (under 5000 evaluations), we consider the hypervolume improvement between  $n = 5000$  and the end of the run. The results of this analysis are presented in Tab. 2. For each configuration, a statistical comparison to the non assisted configuration is performed using a Welsch t-test.

The mean hypervolume improvement is higher for the assisted runs. For the ( $G = 10, \lambda_M = 15$ ), the hypervolume improvement is twice that of the non assisted algorithm, and the difference is statistically significant. The difference, however, is not statistically significant for the filter configuration ( $(G = 1, \lambda_M = 150)$ ), confirming the intuition in [4] that for the same number of evaluations, using several generations on a smaller population will yield better results.

The analysis of the Pareto fronts shows that both configurations of the algorithm are able to find solutions in the same regions of the objective space as the non assisted algorithm, showing that the surrogate assistance does not reduce the diversity of the solutions, therefore giving the NPP operator a variety of possible trade-offs between axial imbalance and cost.

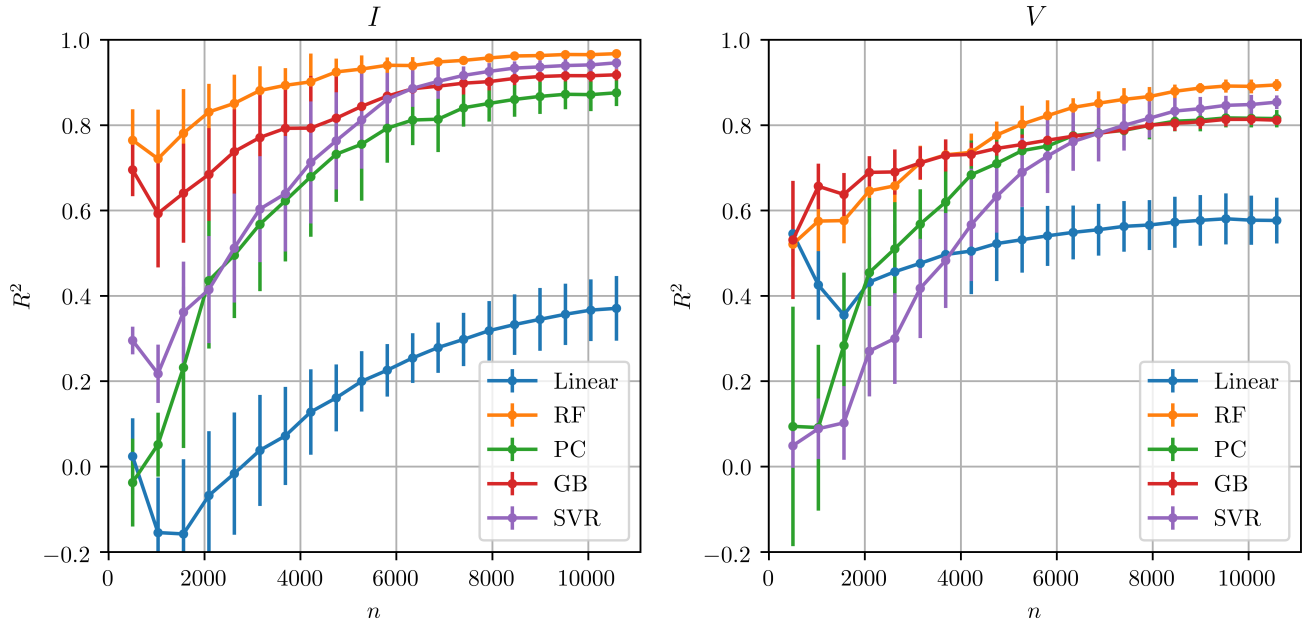


Figure 3: Coefficient of determination ( $R^2$ ) of different surrogate models evaluated on non assisted runs according to the number of evaluated solutions  $n$ .

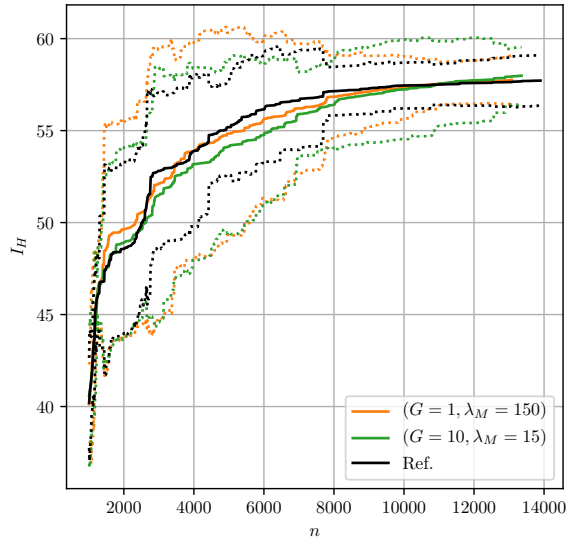


Figure 4: Hypervolume evolution of the surrogate assisted and non assisted algorithm. Full line is the average over the runs, dotted line are the confidence interval.

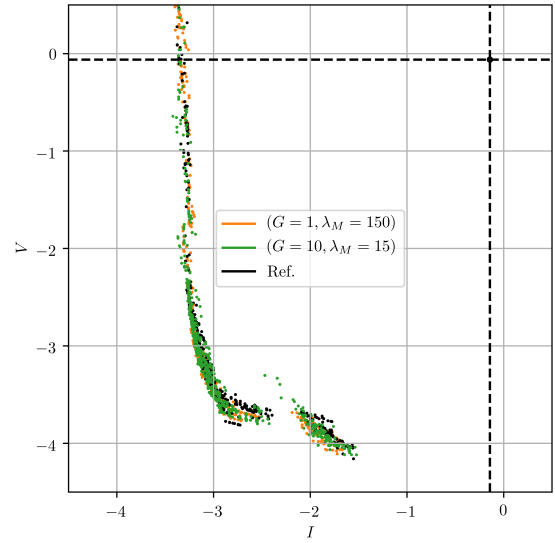


Figure 5: Final Pareto Fronts for the different runs. The dotted black line shows the current rod parameters.

#### 4.4 Discussion

Table 3 shows the confusion matrix of the surrogate assistance for the ( $G = 10, \lambda_M = 15$ ) runs (the matrix of the ( $G = 1, \lambda_M = 150$ ) runs is similar). "Improving solution" means that the solution improves the current best fit for at least one direction. "Predicted



**Table 3: Confusion Matrix (percentage) for the ( $G = 10, \lambda_M = 15$ ) Surrogate assisted AMW-MOEA/D runs**

	Improving	Not improving
Pred. improving	2.19	25.29
Pred. not improving	3.51	69.0

improving" means the solution was predicted to improve the current best for at least one direction, not only the one it was selected for. These two tests do not occur at the same time: between the selection of a solution and the end of its evaluation, because the algorithm is asynchronous, many solutions have been received, therefore the current bests may have changed. This can explain the high rate of false positive: solutions that were predicted improving but were not because the current best changed while they were being computed. The surrogate assistance is too slow in a massively parallel asynchronous framework. However, the rate of improving solutions is 2.28 % for the non assisted algorithm, where it is 5.70 % for the ( $G = 10, \lambda_M = 15$ ) runs, showing that the assistance is still able to find more improving solutions.

Only 27 % of the selected solutions were predicted to be improving, which might seem small: why not increase the number of solutions evaluated by the surrogate to improve that rate? Intuitively, with the asynchronicity and therefore the high rate of false positive, it is not likely that pushing the algorithm so that it always finds a solution that is predicted to be improving will improve performance. On the contrary it could reduce the diversity of the result. Such analysis on the surrogate model leads us to push toward further developments.

However, the increased rate of convergence of the assisted algorithm allows us to have an increased trust in its results, therefore reducing the necessary number of runs required to ensure a good final Pareto front. The solutions found by the assisted algorithm improve both the volume of effluents and imbalance of the plant during the power transients, by at least the same magnitude than the non assisted one (See Fig 5). Lastly, notice that the Pareto set approximation found by our algorithm improves by several orders the actual rod parameters (at the intersection of dotted lines in Fig. 5) which shows the relevance of the approach.

## 5 CONCLUSIONS AND FUTURE WORKS

Reactor operations optimization is a major challenge for the French nuclear industry. The methodology we propose, thanks to the surrogate assistance, is a robust way of reducing the cost of the transient, and reducing the axial instability of the power in order to introduce more renewable energies in energy mix. It provides a variety of trade-offs between these two objectives, with a reasonable computational cost. Moreover, it can easily be adapted to a different model of reactor, or a different power transient. The NPPO problem studied here focuses on one power transient (Fig 2). Though it is a very important transient today, new types of transient will emerge in the coming years. With the increased trust of the multiobjective optimization algorithm, thanks to the surrogate assistance, it is now possible to search for interesting rod managements for a wide variety of future transient with a reduced computational cost.

In addition, this work opens new research directions for surrogate-assisted parallel algorithm dedicated to discrete multiobjective problems. Indeed, for black-box discrete optimization problem, the probability to find an improving solution could be small. Therefore, the trade-off between exploration and exploitation (for instance parameters  $\lambda_M$ , and  $G$  in our algorithm) to select a new candidate solution with respect to surrogate model uncertainty is one of our next research step to enhance the robustness and the efficiency of our methodology.

## ACKNOWLEDGMENTS

The authors would like to thanks IDEX Paris-Saclay for their financial support. We gratefully acknowledge Framatome and EDF for their partnership and support. We thank the GENCI project for the access to the TGCC.

## REFERENCES

- [1] Ricardo Baptista and Matthias Poloczek. 2018. Bayesian Optimization of Combinatorial Structures. In *International Conference on Machine Learning*. 462–471.
- [2] Thomas Bartz-Beielstein and Martin Zaefferer. 2017. Model-based methods for continuous and discrete global optimization. *Applied Soft Computing* 55 (2017), 154–167.
- [3] Adel Ben-Abdennour, Robert M. Edwards, and Kwang Y. Lee. 1992. LQG/LTR Robust Control of Nuclear Reactors with Improved Temperature Performance. *IEEE TRANSACTIONS ON NUCLEAR SCIENCE* 36, 6 (1992), 2286–2294.
- [4] Nicolas Berveglieri, Bilel Derbel, Arnaud Liefoghe, Hernán Aguirre, and Kiyoshi Tanaka. 2019. Surrogate-assisted multiobjective optimization based on decomposition: a comprehensive comparative analysis. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 507–515.
- [5] Neal Dawson-Elli, Seong Beom Lee, Manan Pathak, Kishalay Mitra, and Venkat R Subramanian. 2018. Data science approaches for electrochemical engineers: an introduction through surrogate model development for lithium-ion batteries. *Journal of The Electrochemical Society* 165, 2 (2018), A1–A15.
- [6] Kalyanmoy Deb, Rayan Hussein, Proteek Chandan Roy, and Gregorio Toscano-Pulido. 2018. A taxonomy for metamodeling frameworks for evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 23, 1 (2018), 104–116.
- [7] Bilel Derbel, Dimo Brockhoff, Arnaud Liefoghe, and Sébastien Verel. 2014. On the impact of multiobjective scalarizing functions. In *International Conference on Parallel Problem Solving from Nature*. Springer, 548–558.
- [8] Valentin Drouet, Jean-Michel Do, and Sébastien Verel. 2019. Design of a simulator oriented PWR model and optimization of load follow operations. In *Proceedings of ICAPP 2019*. Juan-les-Pins, France.
- [9] Valentin Drouet, Jean-Michel Do, and Sébastien Verel. 2020. Optimization of load-follow operations of a 1300MW pressurized water reactor using evolutionary algorithms. In *Proceedings of PHYSOR 2020*. Cambridge, UK.
- [10] H. Golfier et al. May 3-7 2009. APOLLO3: a common project of CEA, AREVA and EDF for the development of a new deterministic multi-purpose code for core physics analysis. In *Proceedings of M&C Saratoga Springs*.
- [11] T. Harada and K. Takadama. 2019. Analysis of semi-asynchronous multi-objective evolutionary algorithm with different asynchronies. *Soft Comput* (2019).
- [12] Yaochu Jin. 2011. Surrogate-assisted evolutionary computation: Recent advances and future challenges. *Swarm and Evolutionary Computation* 1, 2 (2011), 61–70.
- [13] Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* 13, 4 (1998), 455–492.
- [14] L. Lemazurier, M. Yagoubi, P. Chevrel, and A. Grossetête. 2017. Multi-objective  $H_2/H_\infty$  gain-scheduled nuclear control design. *IFAC PapersOnLine* 50 (2017).
- [15] Florian Leprière, Sébastien Verel, Cyril Fonlupt, and Virginie Marion. 2019. Walsh functions as surrogate model for pseudo-boolean optimization problems. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 303–311.
- [16] Cheng Liu, Jin-Feng Peng, Fu-Yu Zhao, and Chong Li. 2009. Design and optimization of fuzzy-PID controller for the nuclear reactor power control. *Nuclear Engineering and Design* 239 (2009), 2311–2316.
- [17] Ilya Loshchilov, Marc Schoenauer, and Michèle Sebag. 2010. A mono surrogate for multiobjective optimization. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*. 471–478.
- [18] M. Muniglia. 2017. *Optimisation du pilotage d'un Réacteur à Eau Pressurisée dans le cadre de la transition énergétique à l'aide d'algorithmes évolutionnaires*. Ph.D. Dissertation. Université Paris Saclay.

- [19] F. Pairot. 2008. EPR: high load variation performances with the TMODE core control. In *Proceedings of International Youth Nuclear Congress*. Interlaken.
- [20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
- [21] Herilalaina Rakotoarison, Marc Schoenauer, and Michèle Sebag. 2019. Automated Machine Learning with Monte-Carlo Tree Search. In *IJCAI-19 - 28th International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, Macau, China, 3296–3303. <https://doi.org/10.24963/ijcai.2019/457>
- [22] RTE. 2018. Bilan Prévisionnel de l'équilibre offre-demande d'électricité en France. France.
- [23] Jörg Stork, Martina Frieese, Martin Zaefferer, Thomas Bartz-Beielstein, Andreas Fischbach, Beate Breiderhoff, Boris Naujoks, and Tea Tušar. 2020. Open issues in surrogate-assisted optimization. In *High-Performance Simulation-Based Optimization*. Springer, 225–244.
- [24] Xiping Yang and Longzhou Fu. 1985. Optimal control of xenon oscillations in load follow of a large nuclear reactor, IFAC Control Applications of Nonlinear Programming and Optimization (Eds.).
- [25] Martin Zaefferer, Jörg Stork, Martina Frieese, Andreas Fischbach, Boris Naujoks, and Thomas Bartz-Beielstein. 2014. Efficient global optimization for combinatorial problems. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. 871–878.
- [26] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation* 11, 6 (2007), 712–731.
- [27] Mengdi Zheng, Xiaoliang Wan, and George Em Karniadakis. 2015. Adaptive multi-element polynomial chaos with discrete measure: Algorithms and application to SPDEs. *Applied Numerical Mathematics* 90 (2015), 91–110.