Matheuristics to optimize maintenance scheduling and refueling of nuclear power plants

Nicolas Dupin, El-Ghazali Talbi

December 21, 2018

Abstract

Scheduling the maintenances of nuclear power plants is a complex optimization problem, formulated in 2-stage stochastic programming for the EURO/ROADEF 2010 challenge. The first level optimizes the maintenance dates and refueling decisions. The second level optimizes the production to fulfill the power demands and to ensure feasibility and costs of the first stage decisions. This paper solves a deterministic version of the problem, studying Mixed Integer Programming (MIP) formulations and matheuristics. Relaxing only two sets of constraints of the ROADEF challenge, a MIP formulation can be written using only binary variables for the maintenance dates. The MIP formulations are used to design constructive matheuristics and a Variable Neighborhood Descent (VND) local search. These matheuristics produce very high quality solutions. Some intermediate results explains results of the Challenge: the relaxation of constraints CT6 are justified and neighborhood analyses with MIP-VND justifies the choice of neighborhoods to implement for the problem. Lastly, an extension with stability costs for monthly reoptimization is considered, with efficient bi-objective matheuristics.

Keywords : Matheuristics ; Mixed Integer Programming ; Multi Neighborhood Search ; EURO/ROADEF 2010 Challenge ; Maintenance scheduling; Optimization in Energy

1 Introduction

The 2010 ROADEF/EURO Challenge was specified by the French utility company (EDF) in [25] to address the complex industrial optimization problem to schedule nuclear power plant outages to process maintenances and refueling. Scheduling the maintenances of nuclear power plants is a complex industrial problem: the impact of the outages is calculated with the expected production costs to fulfill the power demands using the available power plants. Furthermore, the feasibility of a high-level maintenance planing must be ensured for low-level production constraints. The optimization problem was formulated using 2-stage stochastic programming for the challenge. Power demands, production capacities and costs were stochastic, discrete scenarios model this uncertainty. The first stage optimization problem concerns the maintenance decisions for nuclear power plants and the refueling quantities. The second stage optimization problem computes production plans implied by the first stage decisions for all stochastic scenarios, to minimize the average production cost.

This ROADEF Challenge is a complex industrial optimization problem with varied constraints and types of variables. The best results were mainly obtained in [13] with an aggressive local search approach. Several approaches in competition were based on exact methods like ([18, 22, 27]), this required heuristic reductions, constraint relaxations and repairing procedure to compute final solutions. The large size of the instances was a bottleneck for the exact methods. With the short development times of the Challenge, the competition focused on having efficient primal heuristics for the whole problem. Intermediate questions were opened for a better understanding of the problem structure. In this paper, we focus on a deterministic version of the problem with few relaxations/simplifications. This work is both motivated to design an efficient algorithm for an industrial application, and to understand better some structures of the problem and some results of the ROADEF Challenge.

This paper is organized as follows. In section 2, we give an overview of the problem. In section 3, we discuss related state-of-the-art elements to appreciate our contributions. In section 4, we present a new MIP compact formulation. In section 5, constructive matheuristics are derived from the previous MIP formulation. In section 6, a Variable Neighborhood Descent with MIP neighborhoods provides local improvements once feasible solutions are known. In section 7, the computational results are discussed on real world instances. In section 8, our contributions are summarized, discussing also future directions of research.

2 Problem statement

We consider in this paper some simplifications of the constraint of the 2010 ROADEF/EURO Challenge, as specified in [25], and an extension to optimize stability of monthly reoptimization. In the Challenge, there were two types of time steps, weekly discretization for the outage dates and a hourly discretization for production time steps. In this study, we consider only weekly time steps, production and demand levels are aggregated with their average value in each week. In the Challenge, discrete scenarios modeled the uncertainty of power demands, production capacities and costs. In this study, we consider only one scenario, aggregating the uncertain parameters with their average value over all the scenarios of the Challenge. Lastly, we relax in this study constraints CT6 (”stretch” decreasing profile) and CT12 (”modulation” constraints).

2.1 Set and index

Two kinds of power plants are modeled. On one hand, Type-2 (or T2) power plants indexed with $i \in \mathcal{I}$, correspond to nuclear power plants. T2 power plants have to be shut down for refueling and maintenance regularly. On the

Table 1: Definitions and notations for the imput parameters

Sets and indexes

$w \in \mathcal{W} = [\![1, W]\!]$	Weekly time steps.
$j \in \mathcal{J} = [\![1, J]\!]$	Flexible (Type 1, T1) power plants.
$i \in \mathcal{I} = [\![1, I]\!]$	Nuclear power plants (Type 2, T2).
$k \in \mathcal{K} = [\![0, K]\!]$	Cycles related to T2 units, $k = 0$ for initial conditions.
$m \in \mathcal{M}_{i,k} = [\![1, \mathbf{Np}_{i,k}]\!]$	Points of the CT6 profile for the cycle (i, k).

Temporal notations

$\mathbf{Dem}^w$	Power demands at time step w.
$\mathbf{F}$	Conversion factor between power and fuel.

Notations for T1 units j

$\mathbf{C}_{j,w}$	Production Costs proportional to the generated power at w.
$\underline{\mathbf{P}}_{j,w}$	Minimal power to generate at time step w.
$\overline{\mathbf{P}}_{j,w}$	Maximal generated power at time step w.

Notations for T2 units i

$\overline{\mathbf{A}}_{i,k}$	Maximal fuel level remaining to process outage $k+1$.
$\mathbf{Bo}_{i,k}$	Fuel level "Bore O" of cycle k
$\mathbf{C}_i$	Proportional cost to the final remaining fuel levels at W.
$\mathbf{c}_{i,k,m}$	Loss coefficient of generated power at mode m and cycle k.
$\mathbf{C}^{pen}_{i,k,w}$	Additional (stability) cost to schedule outage k at week w.
$W^0_{i,k}$	Initial week where outage k was scheduled
$\mathbf{Da}_{i,k}$	Outage duration for maintenance and refueling at cycle k.
$\mathbf{To}_{i,k}$	First possible outage week for cycle k of T2 plant i.
$\mathbf{Ta}_{i,k}$	Last possible beginning week for outage k of T2 plant i.
$\mathbf{f}_{i,k,m}$	Fuel levels for the stretch decreasing profile at mode m and cycle k.
$\overline{\mathbf{P}}^w_i$	Maximal generated power at time step w .
$\mathbf{Q}_{i,k}$	Proportion of fuel that can be kept during reload in cycle k at plant i
$\underline{\mathbf{R}}_{i,k}$	Minimal refueling at outage k.
$\overline{\mathbf{R}}_{i,k}$	Maximal refueling at outage k.
$\overline{\mathbf{S}}_{i,k}$	Maximal fuel level of T2 plant i at production cycle k.
$\mathbf{Xi}_i$	Initial fuel stock of T2 plant i.

other hand, Type-1 (or T1) power plants are indexed with $j \in \mathcal{J}$, model other power plants with more flexibility in the production. Outages and $p$roduction campaigns are indexed with the cycles $k \in \mathcal{K}$ for all T2 plants i. By convention, a cycle begins with the outage period for maintenance and refueling, before the $p$roduction campaign. The outage and production decisions are discretized weekly and indexed with $w \in \mathcal{W} = [\![1; W]\!]$.

2.2 Objective function

The objective function of the Challenge minimizes the power generation cost while satisfying the customer load for all time steps. Production costs of T1 units j are $\mathbf{C}_{j,w}$ proportional to the production levels at a given time step w. Production costs of T2 units are calculated proportionally to the fuel consumption. This makes an aggregated cost function with weights defined to optimize the global financial cost.

In this study, we extend the objective function to consider stability costs for the operational application. Indeed, the French Utility Company reoptimizes monthly the planning of maintenances. The baseline maintenance planning impacts the reoptimized one. The maintenance operations are outsourced and formalized/organized by contracts to external companies fixing the periods of operations. In the model, time window constraints CT13 can model such decisions with hard constraints. In the reality, maintenances can be slightly modified if the financial stakes worth the reorganization efforts, from a contractual and organizational point of view. To model this fact, we consider an objective of stability, defining with $\mathbf{C}^{pen}_{i,k,w}$ the stability cost to reschedule the maintenance (i,k) at week w. Denoting $W^0_{i,k}$ the initial beginning week of maintenance (i,k), $\mathbf{C}^{pen}_{i,k,W^0_{i,k}} = 0$. Any penalization with $\mathbf{C}^{pen}_{i,k,W^0_{i,k}} = 0$ can be considered A linear penalization can be written with $\mathbf{C}^{pen}_{i,k,w} = |w - W^0_{i,k}|$, or a quadratic function $\mathbf{C}^{pen}_{i,k,w} = (w - W^0_{i,k})^2$ are natural choices. Another penalization function can consider constant costs for modified maintenances, with $\mathbf{C}^{pen}_{i,k,w} = 1$ for all $w \neq W^0_{i,k}$. We note that the case of the Challenge is extended with $\mathbf{C}^{pen}_{i,k,w} = 0$.

2.3 Constraints description

Table 2 defines the constraints with their nomenclature from CT1 to CT21 in the challenge specification [25]. CT1 are the demand constraints to equalize productions and demands for each time step and each scenario. Constraints CT2 to CT6 and CT12 model production constraints: production bounds for T1 and T2 power plants and specific technical constraints of nuclear power plants. Constraints CT7 to CT11 model stock level constraints: bounds on stock levels and refueling and CT9 is the equation linking T2 production and fuel consumption. The remaining constraints (CT13 to CT21) are specific to T2 plants outage scheduling. CT13 defines time window constraints to begin the maintenances. Constraints CT14 to CT21 can be unified in a common format of resource constraints applying for the maintenance decisions, as defined in Table 2. These

Table 2: Constraints coupling productions, fuel stock levels and outage decisions

CT1, demand covering: for all time step $w \in \mathcal{W}$, the total production of T1 and T2 power plants must equalize the demands $\mathbf{Dem}_w$.

CT2, T1 production bounds for all $w \in \mathcal{W}$, the production domain of T1 plant $j \in \mathcal{J}$ describes exactly the continuous domain $[\underline{\mathbf{P}}_{j,w}, \overline{\mathbf{P}}_{j,w}]$

CT3-5, T2 production bounds The productions of offline T2 power plants are null during outages. Otherwise, the production domain of T1 plant $j \in \mathcal{J}$ describes exactly the continuous domain $[0, \overline{\mathbf{P}}_{i,w}]$ for all period $w \in \mathcal{W}$.

CT6, "stretch" constraints (relaxed fully or partially in this study) During every time step $w \in \mathcal{W}$ of the production campaign of cycle $k \in K_i$, if the current fuel stock of plant $i \in \mathcal{I}$ is inferior to the level $\mathbf{Bo}_{i,k}$, the production of i is deterministic, following a decreasing profile, piecewise linear function of the stock level, as in Figure 1.

CT7, refueling quantities: The refueling possibilities for outage k of T2 plant i describes exactly the continuous domain $[\underline{\mathbf{R}}_{i,k}, \overline{\mathbf{R}}_{i,k}]$

CT8, initial fuel stock: The initial fuel stock for T2 unit i is $\mathbf{Xi_i}$, known and common for all scenario s.

CT9, fuel stock variation during a production campaign: The fuel stock variation during a production campaign of a cycle (i,k) between t and $t+1$, is proportional to the power produced by i at time step t, with a proportional factor $-\mathbf{F}_t$.

CT10, fuel stock variation during an outage During an outage, the fuel stock variation is the sum of the decisional refueling bounded with CT7 and a certain amount of unspent fuel, calculated with a proportional loss with factor $\mathbf{Q}_{i,k} < 1$ to the residual fuel before refueling.

CT11, bounds on fuel stock The fuel level is in $[0, \mathbf{S}_{i,k}]$ for cycle k of T2 unit i. The fuel level must be lower than $\mathbf{A}_{i,k+1}$ to process outage $k+1$.

CT13, time windows for maintenances The beginning dates of outage k of T2 unit i must be included in $\mathbf{To}_{i,k}, \mathbf{Ta}_{i,k}]$.

Implicit constraints: The maintenances follow the order of set $k \in \mathcal{K}$ without skipping maintenances: if outage $k+1$ is processed, it must follow the cycle k.

CT14-21, resource constraints in maintenance scheduling: For all constraint $c \in \mathcal{C}$, a subset of outages A_c (possibly $\mathcal{I} \times \mathcal{K}$) and a subset of time periods W_c (possibly $W_c\mathcal{W}$) is considered. For all outage $a \in A_c$ and for all time period $w \in W_c$, a resource consumption $R^c_{a,w}$ is defined for the outage a being scheduled at w, and the global resource comsumption at w must be lower than $\overline{R}^c_w$

constraints express minimal spacing/ maximal overlapping constraints among outages, and Minimum spacing constraint between coupling/decoupling dates, resource constraints for maintenances using specific tools or manpower skills and limitations of simultaneous outages with maximal number of simultaneous outages or maximal T2 power off-line.

2.4 Justification of the relaxation of constraints

In this section, we justify the reasons to relax fully modulation constraints CT12, and the stakes to consider partial decreasing profile constraints CT6.

2.4.1 Modulation constraints CT12

In this paper, modulation constraints, denoted CT6 in the Challenge description, are fully relaxed. In the Challenge, it is formulated as a maximal cumulated volume of energy generated when a unit i is not producing at the maximal power $\overline{\mathbf{P}}_{i,w}$ at time step w, the cumulated energy beeing counted for all cycle k when the fuel level is superior to $\mathbf{Bo}_{i,k}$. When the fuel level is inferior to $\mathbf{Bo}_{i,k}$, the production is at the maximal power defined by the CT6 decreasing profile.

A first reason to relax constraints CT12 (especially in [22]) is that modeling CT12 induces more difficulties, requiring additional binary variables for a MIP formulation. A second reason is the evaluation of the impact of such constraints. T2 production is indeed less expensive than T1 production, the cost optimization induces that T2 units are likely to produce at the maximal powers when they are online, so that the modulation capability is a marginally used. The case where modulations cannot be avoided is when the total power of online T2 units exceed the global demands. This phenomenon is local and predictible knowing the demand curves, it suggests to relax firstly constraints modulation CT12 before to repair locally the violated modulation constraints to tackle the problem of the Challenge.

A structural reason to relax constraints CT12 comes from the real technical constraints of nuclear power plants. The real constraints are that the duration where nuclear units do not produce at the maximal power is bounded, with a maximal duration in a intermediate power level similarly with [9]. This induced for tha large time periods of the Challenge to limit the global volume of modulations in cycles. Actually, it is possible to shut down completely a nuclear unit in a production cycle, without maximal duration on the offline period, with minimal durations for the offline and online periods, similarly with [26].

2.4.2 Decreasing profile for T2 units CT6

In the Challenge, the “stretch decreasing profile” CT6 are defined when the fuel stock level of a T2 unit $i \in \mathcal{I}$ is inferior to the level $\mathbf{Bo}_{i,k}$ in the cycle $k \in K_i$. The production of i is deterministically defined, following a decreasing profile, piecewise linear function of the stock level, as in Figure 1.

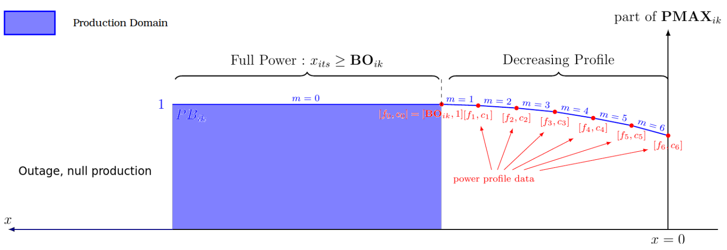

Figure 1: Illustration of the production domain for T2 power plants in the Challenge ROADEF

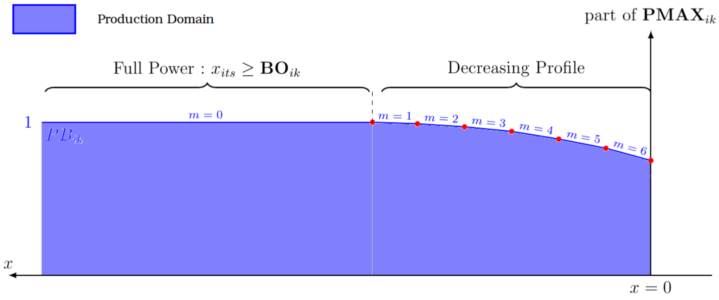

Figure 2: Illustration of the production domain for T2 power plants with light CT6 constraints

In this paper, we will fully relax constraints CT6 or we will consider only the upper bounds constraints as illustrated in Figure 2. A first reason to relax constraints CT6 is is that modeling CT6 induces more difficulties in the MIP modeling and in the MIP solving, requiring additional binary variables for a MIP formulation. A second reason is due to the impact of considering such constraints: T2 production has lower marginal costs than the T1 production. Hence, the optimization tends to have T2 units producing at their maximal power. It justifies the first relaxation with production domain as illustrated in Figure 2. The full relaxation of CT6 seems a priori interesting, to study if complete solutions can be repaired starting from solution of the model relaxing fully CT6 constraints. One of the applicative issue of this work is to analyze the impact for primal heuristics of such modeling variants in the real life solutions.

3 Related work

This section describes the state of the art of the solving approaches for the 2010 EURO/ROADEF Challenge, focusing on the related works both in the heuristic and mathematical programming approaches.

3.1 General facts

Although the teams of the challenge worked independently, similar ideas occurred. It seems strongly related to the specificity of the problem and the size of instances. A major difficulty comes with the large size of instances. Many approaches reduced the size of instances with different preprocessing strategies: reducing the time windows with exact or heuristic preprocessing, or aggregating production time steps to week or aggregating the stochastic scenarios to the average one. The heuristic preprocessing requires a repairing postprocessing, which made no difficulty to recover feasible solutions for the whole problem.

We distinguish three main types of approaches for the ROADEF challenge, matheuristics based on MIP exact approaches, decomposition heuristics following the 2-stage structure and straightforward local search approaches. We note only one attempt of population heuristics, with an ant colony optimization approach, with reported results using more memory than specified. An explanation comes from the memory limitations in [25] and the size of instances.

An open question after the Challenge was to prove dual bounds for the large size instances of the competition. Brandt et al. (2013) furnished the first dual bounds [7] using dual heuristics, the bounds proven in [11, 10] improved these last bounds.

3.2 Matheuristics based on MIP exact approaches

Two matheuristic approaches were designed for the ROADEF Challenge. A simplified MIP problem is solved, and the solution is repaired to ensure the feasibility for all the constraints and computing the cost in the full model. In both cases, the MIP model is solved heuristically using truncated exact methods, the simplified MIP being too difficult to be solved in one hour. We note that Joncour et al (2010) provided an exact and compact MIP formulation for the full problem, introducing binaries for all time steps, cycle and production mode $m \in \mathcal{M}$ [17]. This size of problem is not reasonable for a B&B search, even in a truncated mode, this work was a preliminary work before a Column Generation (CG) work similar with [27].

Rozenkopf et al. (2013) considered an exact formulation of CT6 constraints, in a CG approach dualizing coupling constraints among units [27], i.e. CT1 demands and CT14 to CT21 scheduling constraints. The MIP considered a unique scenario, the average one, with production time steps aggregated weekly. The CG approach is deployed to compute a LP relaxation relaxing scheduling constraints CT14-CT21, with subproblems solved by dynamic programming. The further CG heuristic incorporate these scheduling constraints in the integer resolution with the columns generated. The solution of this MIP gives the outage dates and week where the CT6 constraints are activated, the final cost and feasibility issues for the whole problem are given computing the production by Linear Programming (LP). This approach was one of the most effective for the challenge, up to 2% of the best solution known.

Lusby et al. (2013) relaxed fully the constraints CT6 and CT12, leading to a MIP formulation with binaries only for the outage decisions [22]. The production time steps were aggregated to weeks for size reasons. The stochastic scenarios were not aggregated, leading to 2-stage stochastic programming structure solved by Bender's decomposition. The master problem concerns the dates of outages and the refueling quantities. Independent sub-problems are defined for each stochastic scenarios with continuous variables for productions and fuel levels. The heuristic of [22] computes first the LP relaxation exactly with the Bender's decomposition algorithm. Then, a cut&branch approach repairs integrity, branching on binary variables without adding new Bender's cuts. The resulting heuristic approach was efficient for the small dataset of the qualification, difficulties and inefficiencies occur for the final instances of the competition.

3.3 Heuristics based on the 2-stage decomposition

A natural idea to solve the problem by decomposition is to follow the 2-stage structure of stochastic programming, distinguishing the high-level maintenance and refueling problem of T2 units from the lower level production problems, as in [2, 6, 7, 14, 15, 18].

The high-level problem fixes the maintenance dates and the refueling levels modifying slightly the current solution and respecting scheduling constraints CT7-CT11 and CT13-CT21. The previous maintenance planning of the last iteration (or the initial planning) is slightly modified in a defined neighborhood, where the feasibility of constraints CT13-CT21 can be ensured using MIP of Constraint Programming models.

The low-level subproblems compute independently for all scenario the production plans optimizing the production costs and fulfilling the constraints CT1-CT6 and CT12 having fuel levels and maintenance dates fixed. Such production problems can be solved using greedy strategies following increasing production costs or using LP problems.

We note that the operational approach in the French Utility Company was in this scope before the Challenge, we refer to [19]. The approaches in competition were not among the most efficient, which can be analyzed comparing with the solving characteristics of frontal local search approaches.

3.4 Local search approaches

The last category gathers *local search approaches*, tackling the problem straightforward without decomposition. In this category, we find the two best approaches of the Challenge in terms of solution quality, having both very similar results, a methodology similar with LocalSolver [13, 3] and an unpublished Simulated Annealing local search [24].

The crucial points are in this case the choices of the neighborhoods to have efficient moves with the possibility to calculate quickly the cost of a locally modified solution, to be able to explore aggressively a lot of solutions. The final neighborhoods chosen by Gardi et al (2011) are reported in [13]:

- k-MoveOutagesRandom: select k outages among T2 plants randomly and move them randomly in feasible time intervals, that is, ensuring the respect of earliest and latest starting dates (CT13 constraints) and maximum stocks before and after refueling (CT11 constraints);
- k-MoveOutagesConstrained: select T2 plants which are involved in combinatorial constraints related to outage scheduling (CT14-CT21) randomly, select k outages of these plants randomly, move these outages randomly in feasible time intervals;
- k-MoveOutagesConsecutive: select a T2 plant randomly, select k consecutive outages of these plant randomly, move these outages randomly in feasible time intervals.

To explain the better quality of primal solutions of frontal local search compared to heuristic decompositions, heuristic decomposition are slowed down by the use of MIP,LP or CP computations. On the contrary, frontal local search explored aggressively more solutions. An other explanation is due to the neighborhoods chosen for decomposition approaches, that were mostly very small (e.g allowing to move only one an outage from only one week). Such neighborhoods were used by [24] in the qualification phase, and lead to bad solutions. The effort to diversify and get larger neighborhoods is mentioned in [13] to have a neighborhood structure compatible to their aggressive exploration. We note that excellent results are reported in [13] with a time limit of few minutes, which is interesting for an industrial application.

Exact approaches are also slowed down by the use of MIP, LP or CP computations, with also memory space problematic, as in [22] to tackle large size instances with few approximations. Tractable exact optimization based approaches cannot avoid over-costs due to their approximation model, which explains the superiority of approaches [13, 24] in this case.

4 MIP formulation

In this section, we provide a MIP formulation for the problem, relaxing only constraints CT6 and CT12 as in [22]. It leads to a MIP formulation where the only binary variables are the decisions of outage beginning weeks .

4.1 Definition of the variables

A major modeling difference with [22] is in the binary variable definitions of $d_{i,k,w}$: we define $d_{i,k,w} = 1$ if and only if the outage beginning week for unit i's cycle k is before week w. Binary variables $x_{i,k,w}$ in [22] are equal to 1 if and only if outage beginning week for cycle (i, k) is exactly w. This choice allows to have efficient branching following results of [9]. We extend the notations with $d_{i,k,w} = 0$ for $k > K_i$,$d_{i,-1,w} = 1$ for $w < 0$, $d_{i,k,w} = 0$ for $w < 0$ and $k > -1$.

Other continuous variables must be introduced: refueling quantities $r_{i,k}$ for each outage (i,k), T2 power productions $p_{i,k,w}$ at cycle k, fuel stocks at the beginning of campaign (i,k) (resp at the end) $x^{init}_{i,k}, x^{fin}_{i,k}$, T1 power productions $p_{j,w}$, and fuel stock x^f_i at the end of the optimizing horizon. We note that T2 power productions $p_{i,k,w}$ are duplicated for all cycle k to have a linear model, $p_{i,k,w} = 0$ if week w is not included in the production cycle k.

$d_{i,k,w} \in \{0,1\}$	Beginning dates of outages of unit i at cycle k at w.
$r_{i,k} \geqslant 0$	Refueling levels of unit i at cycle k .
$p_{i,k,w} \geqslant 0$	Production of T2 unit i at cycle k at w.
$p_{j,w} \geqslant 0$	T1 production of unit j at week w .
$r_{i,k} \geqslant 0$	Refueling levels for unit i at cycle k.
$x^f_i \geqslant 0$	Fuel stock of T2 unit i at the end of the optimizing horizon.
$x^{init}_{i,k} \geqslant 0$	Fuel levels of T2 unit i at the beginning of cycle k.
$x^{fin}_{i,k} \geqslant 0$	Fuel levels of T2 unit i at the end of production cycle k.

Table 3: Set of variables

4.2 MIP formulation relaxing CT6 and CT12

Relaxing CT6 and CT12, it gives rise to the MIP formulation below. (2) is required with definition of variables d. (3) and (4) model CT13 time windows constraints: outage (i,k) is operated between weeks $\mathbf{To}_{i,k}$ and $\mathbf{Ta}_{i,k}$. (5) models CT1 demand constraints. (6) models CT2 bounds on T1 production. (7) models CT3, CT4 and CT5 bounds on T2 production. (8) models CT7 refueling bounds, with a null refueling when outage i,k is not operated, ie $d_{i,k,W} = 0$. (9) writes CT8 initial fuel stock. (10) writes CT9 fuel consumption constraints on stock variables of cycles k $x^{init}_{i,k}, x^{fin}_{i,k}$. (11) models CT10 fuel losses at refueling. (12) writes CT11 bounds on fuel stock levels only on variables $x^{init}_{i,k}$ which are the maximal stocks level over cycles k. thanks to (10). (13) models CT11 min fuel stock before refueling, these constraints are active for a cycle k only if the cycle is finished at the end of the optimizing horizon, ie if $d_{i,k+1,W} = 1$, which enforces to have disjonctive constraints where case $d_{i,k+1,W} = 0$ implies a trivial constraints thanks to (12). (14) linearizes the constraints to enforce x^f_i to be the fuel stock at the end of the time horizon. x^f_i is indeed the $x^{fin}_{i,k}$ such that $d_{i,k,W} = 1$ and $d_{i,k+1,W} = 0$, for the disjonctive constraints (14) that write a trivial constraints in the other cases thanks to (12), we define $\overline{S}_i = \max_k \mathbf{S}_{i,k}$. (15) is a common framework for scheduling constraints from CT14 to CT21, which was noticed independently in [22, 18].

$$
\begin{aligned}
v_0 = \min \quad & \sum_{i,k} \mathbf{C}^r_{i,k} r_{i,k} + \sum_{j,w} \pi_s \mathbf{C}^p_{j,w} \mathbf{D}^t \, p_{j,w} - \sum_i \pi_s \mathbf{C}^f_i x^f_i && (1)\\
\forall i,k,w, \quad & d_{i,k,w-1} \leqslant d_{i,k,w} && (2)\\
\forall i,k, \quad & d_{i,k,\mathbf{To}_{i,k}-1} \leqslant 0 && (3)\\
\forall i,k, \quad & d_{i,k,\mathbf{Ta}_{i,k}} \geqslant 1 && (4)\\
\forall w, \quad & \textstyle\sum_{i,k} p_{i,k,w} + \sum_j p_{j,w} = \mathbf{Dem}^w && (5)\\
\forall j,w, \quad & \mathbf{Pmin}_{j,t} \leqslant p_{j,w} \leqslant \mathbf{Pmax}_{j,t} && (6)\\
\forall i,k,w, \quad & p_{i,k,w} \leqslant \mathbf{Pmax}_{i,w}(d_{i,k,w-\mathbf{Da}_{i,k}} - d_{i,k+1,w}) && (7)\\
\forall i,k, \quad & \mathbf{Rmin}_{i,k} \, d_{i,k,W} \leqslant r_{i,k} \leqslant \mathbf{Rmax}_{i,k} \, d_{i,k,W} && (8)\\
\forall i, \quad & x^{init}_{i,0} = \mathbf{Xi}_i && (9)\\
\forall i,k, \quad & x^{fin}_{i,k} = x^{init}_{i,k} - \textstyle\sum_t \mathbf{D}^t \, p_{i,k,w} && (10)\\
\forall i,k, \quad & x^{init}_{i,k} - \mathbf{Bo}_{i,k} = r_{i,k} + \tfrac{\mathbf{Q}_{i,k-1}}{\mathbf{Q}_{i,k}}(x^{fin}_{i,k-1} - \mathbf{Bo}_{i,k-1}) && (11)\\
\forall i,k, \quad & x^{init}_{i,k} \leqslant \mathbf{S}_{i,k} && (12)\\
\forall i,k, \quad & x^{fin}_{i,k} \leqslant \mathbf{A}_{i,k+1} + (\mathbf{S}_{i,k} - \mathbf{A}_{i,k+1})(1 - d_{i,k+1,W}) && (13)\\
\forall i,k, \quad & x^f_i \leqslant x^{fin}_{i,k} + \overline{S}_i(d_{i,k,W} - d_{i,k+1,W}) && (14)\\
\forall c \in \mathcal{C}, w \in W_c, \quad & \textstyle\sum_{(i,k)\in \mathbf{A}^c} (R^c_{a,w} - R^c_{a,w-1}) d_{i,k,w} \leqslant \overline{R}^c_w && (15)\\
& d \in \{0,1\}^N, r, p, x \geqslant 0 && (16)
\end{aligned}
$$

4.3 Adding stability cost for dynamic reoptimization

Previously, the objective function (1) consider only the financial cost. With the extension of stability cost as defined in Section 2.2, we shall consider two objectives, the financial cost $\mathbf{C}^{roadef}$ and the stability cost $\mathbf{C}^{desorg}$, defined with:

$$
\begin{aligned}
\mathbf{C}^{roadef} &= \sum_{i,k} \mathbf{C}^r_{i,k} r_{i,k} + \sum_{j,w} \pi_s \mathbf{C}^p_{j,w} \mathbf{D}^t \, p_{j,w} - \sum_i \pi_s \mathbf{C}^f_i x^f_i && (17)\\
\mathbf{C}^{desorg} &= \sum_{i,k,w} \mathbf{C}^{pen}_{i,k,w} (d_{i,k,w} - d_{i,k,w-1}) && (18)
\end{aligned}
$$

The two objectives can be adversarial, the solution of the bi-objective optimization is thus a set of Pareto non-dominated solutions. We note that in the case $\mathbf{C}^{pen}_{i,k,w}$ is defined by $\mathbf{C}^{pen}_{i,k,w} = 1$ for all $w \neq W^0_{i,k}$. and $\mathbf{C}^{pen}_{i,k,W^0_{i,k}} = 0$, i.e. $\mathbf{C}^{desorg}$ counts the number of desorganization, this makes the objective $\mathbf{C}^{desorg}$ integer with a granularity of 1, which is interesting considering an ε-constraint method with $\varepsilon = 1$ ([21]). In this case, the constraint to bound the number of modifications of the original planning by at most **Nmax** is given by:

$$\sum_{i,k}(1 + d_{i,k,W^0(i,k)-1} - d_{i,k,W^0(i,k)}) \leqslant \mathbf{Nmax} \tag{19}$$

Another application of this extension is to repair an initial solution $W^1(i,k)$ of maintenance planning, that may be infeasible, looking for the closest feasible solution as in Feasibility Pump (FP, [4]) using $\mathbf{C}^{pen}_{i,k,w} = |w - W^1_{i,k}|$ or $\mathbf{C}^{pen}_{i,k,w} = (w - W^0_{i,k})^2$ and minimizing $\mathbf{C}^{desorg}$ or $\mathbf{C}^{desorg} + \epsilon\mathbf{C}^{roadef}$ with $\epsilon > 0$ small.

4.4 Adding lighter CT6 constraints

This section provides a lighter formulation for stretch constraints enforcing only upper bounds on the production as illustrated Figure 2. As illustrated in Figure 2, the equation linking modes $m-1$ et m is:

$$Y = \frac{\mathbf{c}_{i,k,m-1} - \mathbf{c}_{i,k,m}}{\mathbf{f}_{i,k,m-1} - \mathbf{f}_{i,k,m}}(X - \mathbf{f}_{i,k,m}) + \mathbf{c}_{i,k,m}$$

where $Y \in [0,1]$ is the ratio maximal power in stretch / maximal power $\overline{\mathbf{P}}_i^w$, and X denotes the residual fuel stock.

To write mathematically constraints to have upper bounds for the T2 production, we use that stretch decreasing profile is concave. Thus, the production domains are defined with the intersection of the semi spaces defined with the equations of the different modes. Adding in the model variables $x_{i,w}$ denoting the residual fuel stock of the unit i at time period w, the intersection of the semi spaces gives rise to following stretch constraints

$$\forall i,k,w,m > 0, \quad \frac{p_{i,k,w}}{\overline{\mathbf{P}}_i^t} \leqslant \frac{\mathbf{c}_{i,k,m-1} - \mathbf{c}_{i,k,m}}{\mathbf{f}_{i,k,m-1} - \mathbf{f}_{i,k,m}}(x_{i,w} - \mathbf{f}_{i,k,m}) + \mathbf{c}_{i,k,m} \tag{20}$$

The definition of new variables $x_{i,w}$ requires following linking constraints to enforce $x_{i,w}$ to be the residual fuel:

$$\forall i,k,w, \quad x_{i,w} \leqslant x^{init}_{i,k,s} - \sum_{t' \leqslant t} \mathbf{D}^{t'}\, p_{i,k,w'} + M_i\,(1 - d_{i,k,w_t} + d_{i,k-1,w_t}) \tag{21}$$

where $M_i = \max_k \overline{\mathbf{S}}_{i,k}$, such M_i verifies $x_{i,t} \leqslant M_i$. Indeed, if $d_{i,k,w} - d_{i,k-1,w} = 1$, week w happens in cycle k, the active constraint is $x_{i,t} \leqslant x^{init}_{i,k,s} - \sum_{t'<t} \mathbf{D}^{t'} p_{i,k,w'}$, otherwise we have $x_{i,t} \leqslant M_i$, which is trivial thanks to the definition of M_i.

A special case is interesting: if ($\mathbf{c}_{i,k,m}, \mathbf{f}_{i,k,m-1}$) do not depend on indexes k, we can have a MIP formulation using less constraints than previously, writing the constraints for the global production power $\sum_k p_{i,k,w}$:

$$\forall i,w,m > 0 \quad \sum_k \frac{p_{i,k,w}}{\overline{\mathbf{P}}_i^t} \leqslant \frac{\mathbf{c}_{i,m-1} - \mathbf{c}_{i,m}}{\mathbf{f}_{i,m-1} - \mathbf{f}_{i,m}}(x_{i,w} - \mathbf{f}_{i,m}) + \mathbf{c}_{i,m} \tag{22}$$

5 Constructive matheuristics

This section aims to design constructive matheuristics. The modeling and solving facilities with MIP ensures constraint feasibility, with heuristic reductions to have tractable MIP resolutions for large scale instances.

5.1 A very simplified MIP formulation

A first heuristic approximates the previous MIP formulation, noticing that nuclear energy is cheaper that flexible energy from T1 power plants. A MIP formulation is considered where the nuclear production is either null (during outages), either at its maximal level, ie $p_{i,k,w} = \mathbf{Pmax}_{i,w}(d_{i,k,w-\mathbf{Da}_{i,k}} - d_{i,k+1,w})$. This can lead to infeasibilities, it requires some relaxations. Nuclear power can be upper than some low demands $\mathbf{Dem}^w$, leading infeasibilities with 5 written with an equality. Relaxing 5 with minimal production inequalities fixes such problems. Furthermore, if a maximal nuclear production can be seen as economically optimal, it can be infeasible with the fuel constraints. Indeed, a maximal nuclear production (and thus a maximal fuel consumption), can be in contradiction with CT13 earliest dates of outages and positivity of fuel levels Hence, we introduce continuous variables $\Delta_{i,k}$ to authorize negative fuel stocks in soft constraints with high values of $C^{\Delta}_{i,k}$ in the objective function, in a following simplified MIP formulation:

$$
\begin{array}{rc}
v^{simpl} = \min & \sum_{i,k} \mathbf{C}^{rld}_{i,k} r_{i,k} + \sum_{j,t} \mathbf{C}^{prd}_{j,t} \mathbf{D}^t \, p_{j,t} - \sum_i \mathbf{C}^{val}_i x^{fin}_i + \sum_i C^{\Delta}_{i,k} \Delta_{i,k} \\
 & A^{ordo} d \geqslant a^{ordo} \\
 & \mathbf{Rmin}_{i,k} \, d_{i,k,W} \leqslant r_{i,k} \leqslant \mathbf{Rmax}_{i,k} \, d_{i,k,W} \\
\forall i,k, & x^{init}_{i,k} \leqslant \mathbf{Smax}_{i,k} \\
\forall i,k, & x^{fin}_{i,k} = x^{init}_{i,k} - \sum_t \mathbf{D}^t \, p_{i,k,w} + \Delta_{i,k} \\
\forall i,k, & x^{fin}_{i,k} \leqslant \mathbf{Amax}_{i,k+1} + (\mathbf{Smax}_{i,k} - \mathbf{Amax}_{i,k+1}) \, (1 - d_{i,k+1,W}) \\
\forall i,k, & x^{init}_{i,k} - \mathbf{Bo}_{i,k} = r_{i,k} + \frac{\mathbf{Q}_{i,k}-1}{\mathbf{Q}_{i,k}} (x^{fin}_{i,k-1} - \mathbf{Bo}_{i,k-1}) \\
\forall i,k, & x^{fin}_i \leqslant x^{fin}_{i,k} + \mathbf{Smax}_i (d_{i,k,W} - d_{i,k+1,W}) \\
\forall j,t, & \mathbf{Pmin}_{j,t} \leqslant p_{j,t} \leqslant \mathbf{Pmax}_{j,t} \\
\forall i,w, & p_{i,k,w} = \mathbf{Pmax}_{i,w} (d_{i,k,w_t - \mathbf{Da}_{i,k}} - d_{i,k+1,w_t}) \\
\forall w, & \sum_{i,k} p_{i,k,w} + \sum_j p_{j,w} \geqslant \mathbf{Dem}^w \\
 & d \in \{0,1\}^N, r, p, x, \Delta \geqslant 0
\end{array}
$$

This simplified MIP formulation is used to build a first maintenance planning. Such planning requires only a LP computation to compute the optimal production cost related, ensuring feasibility of all the constraints. If this high level planning cannot lead to feasible production and fuel levels, a local reparation is required.

5.2 Restrict- Relax-and-Fix algorithm

To decrease the size of the MIP computation, Restrict- Relax-and-Fix (RRF, [16]) strategies partition variables and iterate with partial MIP computations with fixing and relaxation reductions. A RRF strategy can be implemented partitioning the variables following the cycle index k. It computes successively the solutions for cycles k, the dates of cycles $k' < k$ are fixed by the previous optimizations, and the variables related to the cycles $k' > k$ are relaxed continuously or even simplified thanks to the aggregation used in [11].

As is, some feasibility difficulties can occur when a cycle optimization fixes outages in a way that no solution exist thereafter to place the remaining outage because of scheduling constraints CT13-CT21 and/or fuel constraints CT11. A crucial point is also to have enough flexibility in the iterated fixing decisions to have feasible solutions for the next iterations. This induces in Algorithm 1 to consider binaries for cycle k and $k+1$ for the iteration placing outages k, and to relax continuously/simplify formulations only for cycles $k' > k+1$. The remaining fuel costs are relaxed also to focus on the local optimization around the cycles to optimize, aggregating the cycles $k > k_0$ in the cycle k_0+1 relaxing the outage and the fuel constraints. The MIP formulation considered for each iteration is thus:

$$v_{RRF}(k^0) = \min \quad \sum_{i,k} \mathbf{C}^{rld}_{i,k} r_{i,k} + \sum_{j,w} \mathbf{C}^{prd}_{j,w} \overline{D}^w \; p_{j,w} \tag{23}$$

$$\forall i, k \leqslant k_0+1, w, \quad d_{i,k,w-1} \leqslant d_{i,k,w} \tag{24}$$

$$\forall i, k \leqslant k_0+1, \quad d_{i,k,\mathbf{To}_{i,k}-1} \leqslant 0 \tag{25}$$

$$\forall i, k \leqslant k_0+1, \quad d_{i,k,\mathbf{Ta}_{i,k}} \geqslant 1 \tag{26}$$

$$\forall w, \quad \textstyle\sum_{i,k \leqslant k_0+1} p_{i,k,w} + \sum_j p_{j,w} = \mathbf{Dem}^w \tag{27}$$

$$\forall j, w, \quad \mathbf{Pmin}_{j,t} \leqslant p_{j,w} \leqslant \mathbf{Pmax}_{j,t} \tag{28}$$

$$\forall i, k \leqslant k_0+1, w, \quad p_{i,k,w} \leqslant \mathbf{Pmax}_{i,w}(d_{i,k,w-\mathbf{Da}_{i,k}} - d_{i,k+1,w}) \tag{29}$$

$$\forall i, k \leqslant k_0+1, \quad \mathbf{Rmin}_{i,k} \; d_{i,k,W} \leqslant r_{i,k} \leqslant \mathbf{Rmax}_{i,k} \; d_{i,k,W} \tag{30}$$

$$\forall i, k \leqslant k_0+1, \quad x^{fin}_{i,k} = x^{init}_{i,k} - \textstyle\sum_t \mathbf{D}^t \; p_{i,k,w} \tag{31}$$

$$\forall i, k \leqslant k_0+1, \quad x^{init}_{i,k} - \mathbf{Bo}_{i,k} = r_{i,k} + \tfrac{\mathbf{Q}_{i,k}-1}{\mathbf{Q}_{i,k}}(x^{fin}_{i,k-1} - \mathbf{Bo}_{i,k-1}) \tag{32}$$

$$\forall i, k \leqslant k_0+1, \quad x^{init}_{i,k} \leqslant \mathbf{S}_{i,k} \tag{33}$$

$$\forall i, k \leqslant k_0, \quad x^{fin}_{i,k} \leqslant \mathbf{A}_{i,k+1} + (\mathbf{S}_{i,k} - \mathbf{A}_{i,k+1})(1 - d_{i,k+1,W}) \tag{34}$$

$$\forall i, k \leqslant k_0+1, \quad x^f_i \leqslant x^{fin}_{i,k} + \overline{S}_i(d_{i,k,W} - d_{i,k+1,W}) \tag{35}$$

$$\forall c \in \mathcal{C}, w \in W_c, \quad \textstyle\sum_{(i,k \leqslant k_0+1) \in \mathbf{A}^c} (R^c_{a,w} - R^c_{a,w-1}) d_{i,k,w} \leqslant \overline{R}^c_w \tag{36}$$

$$d \in \{0,1\}^N, r, p, x \geqslant 0 \tag{37}$$

Algorithm 1: Restrict- Relax-and-Fix algorithm

```
Initialisation: define matrix $W_{i,k} = 0$ for all $i \in \mathcal{I}, k \in \mathcal{K}$.
for $k = 1$ to $K - 1$
    solve the MIP $v_{RRF}(k)$ with $\mathbf{Ta}_{i,k'} = \mathbf{To}_{i,k'} = W_{i,k'}$ for $k' < k$
    store the solutions $d_{i,k}$
    update $W_{i,k} = \sum_w (1 - d_{i,k})$
end for
solve the MIP $v_0$ with $\mathbf{Ta}_{i,k'} = \mathbf{To}_{i,k'} = W_{i,k'}$ for $k' < K$
return the solution of the last MIP
```

5.3 Construct, Merge, Solve & Adapt algorithm

Another RRF decomposition iterating unit by unit could be considered. Such heuristic has several weaknesses a priori. Firstly, having an infeasibility at an iteration does not allow to finish the resolution with the sequentiality of the algorithm. Secondly, the choice of the iterated unit order can be crucial.

To face these two points, a Construct, Merge, Solve & Adapt (CMSA, [5]) is implemented, computing independently (for a possible parallel computation) planning for all units as if they were alone, seeing the other units as T1 power plants. Merging these planning has significant chances to be unfeasible for the scheduling constraints CT14-CT21, a second phase repairs the feasibility around this "ideal" solution, which makes no asymmetry in the choice of the units.

Algorithm 2: Construct, Merge, Solve & Adapt

```
Input: a repairing operator $\mathcal{R}$
Initialisation: define matrix $W_{i,k} = -1$ for all $i \in \mathcal{I}, k \in \mathcal{K}$.
.
for each T2 unit $i$:
    define new demands $\mathbf{Dem}^w_{new} = \frac{\mathbf{Pmax}_{i,w}}{\sum_{i'} \mathbf{Pmax}_{i',w}} \mathbf{Dem}^w$
    solve the MIP $v_0$ with $\mathcal{I} = \{i\}$ and new demands $\mathbf{Dem}^w_{new}$
    store the maintenance planning $W_{i,k} = \sum_w (1 - d_{i,k})$
end for
Merge the maintenance plannings $W_{i,k}$.
Compute as an LP the related production and fuel decisions.
if the current planning is unfeasible:
    repair the current solution with $\mathcal{R}$
end if
return the last current solution
```

6 POPMUSIC-VND matheuristic

Once a feasible solution is built with previous constructive matheuristics, this section improves the current solution in a local search procedure.

Algorithm 3: multi neighborhood descent with MIP neighborhoods

```
Input: an initial solutions, a set and order of neighborhoods to explore
Initialisation: currentSol = initSolution, $\mathcal{N}$ =initial neighborhood.
while the stopping criterion is not met
        define the MIP with incumbent currentSol and the neighborhood $\mathcal{N}$)
        define currentSol as warmstart
        currentSol = solveMIP(MIP,timeLimit( $\mathcal{N}$))
        $\mathcal{N}$ = nextNeighborhood($\mathcal{N}$)
end while
return CurrentSolution
```

6.1 General algorithm

The general local search algorithm is described in the Algorithm 3. The neighborhoods are defined using MIP definitions, similarly to [1]. MIP neighborhoods allow to explore large neighborhoods using recent progresses of MIP solvers, to have fewer and better local optimums than small neighborhoods approaches. Many MIP neighborhoods can be defined, Algorithm 3 changes systematically the choice of the neighborhood within the local search, similarly with multi neighborhood search approaches. It induces that a local optimum for the whole local search is a local optimum for all the neighborhoods considered in Algorithm 3. This a second property which ensures to have less and better local optimums than classical local search approaches. The stopping criterion could be a maximal time limit or a maximal number of iterations, or being in a local extremum for all neighborhoods.

Usually, MIP neighborhoods are defined for small sub-problems where the exact B&B computation to optimality is quick, as in [20]. In this work, MIP neighborhoods are defined with three characteristics for an efficient MIP neighborhood searc of primal solutions in small time limits. Parametrization is empiric, for a good trade-off between solution quality and time spent in MIP solving. The MIP neighborhoods are thus defined as following:

- The **restriction of search space**: Variable fixations or other extra constraints that the current solution satisfies to have an easier B&B solving.
- a **B&B stopping criterion**: it must defined so that the B&B search is efficient in a short solving time. Parametrizing short time limits is a first step, additional parameters like stopping a threshold in the gap between the best primal and dual bounds can be defined in order to not waste time in proving optimality without improving significantly the best primal solutions.
- a **specific MIP solving parametrization**: for an efficient B&B search in the defined time limit. Easy neighborhoods need no specific MIP parametrization, for a better efficiency in short resolution time for difficult sub MIPs, emphasizing the heuristic search with `mipemphasis` parameter,

disabling or limiting cutting plane passes with `cutpass` parameter (in the terminology of Cplex).

The current solution is the primal solution given by the last B&B resolution and it is also defined as warmstart for the next B&B resolution to improve the efficiency of B&B primal heuristics, enabling RINS or Local Branching heuristics from the beginning. This ensures that the solution given by the MIP resolution is at least as good as the current solution at each iteration. This algorithm is thus a steepest descent algorithm, similarly to MIP-VND as in [20].

6.2 MIP neighborhoods

Multiple types of large and variable neighborhoods can be defined. We denote with $W^o_{i,k}$ the beginning week of outage (i,k) in the current incumbent solution. Neighborhoods define strategies to fix variable $d_{i,k,w}$ around the solution defined by $W^o_{i,k}$. Generic neighborhoods are first derived from [8, 12]:

- $\mathcal{N}^{rins}$: Similarly to RINS heuristic [8], variables are fixed if they are a common integer value in the LP relaxation and in the current solution.
- $\mathcal{N}^{LB}_k$: for k an integer, k-Local Branching neighborhood allows only k modifications to the incumbent, adding constraints as in [12]:

$$\sum_{i,k,w:w \geqslant W^o_{i,k}} d_{i,k,w} + \sum_{i,k,w:w<W^o_{i,k}} (1-d_{i,k,w}) \leqslant k$$

We note that neighborhoods $\mathcal{N}^{LB}_k$ have the same number of variables than the original problem, the acceleration opf the MIp solving is induced by the less nuimber of branching to explore the space thanks to the additional constraints.

The multi-index structure allows to define neighborhoods fixing variables along a type of indexation:

- $\mathcal{N}^{units}_I$ unit selection: only T2 units $i \in I \subset \mathcal{I}$ are reoptimized.
- $\mathcal{N}^{TW}_{(a,b)}$: all outages are reoptimized in the time window $[W^o_{i,k} - ak - b, W^o_{i,k} + ak + b]$.
- $\mathcal{N}^{cycles}_{k,k'}$: variables relative cycles k'' with $k \leqslant k'' \leqslant k'$ are reoptimized.

$\mathcal{N}^{TW}_{(0,b)}$ would be a reoptimization with a fixed radius b around the incumbent, $\mathcal{N}^{TW}_{(0,1)}$ is thus a natural neighborhood widely used for the Challenge. The linear dependence with $a > 0$ induces a "funnel" structure, to model that a move on the first outages can imply larger moves for the succeeding outages.

The weakness of such neighborhoods is that it may not be possible to swap the order of outages constrained with scheduling constraints like CT14 with short neighborhoods. $\mathcal{N}^{cycles}_{k,k'}$ is a first answer to have complementary neighborhoods to swap the order of outages. $\mathcal{N}^{units}_I$ can be implemented for all single units, in this case, it authorizes large neighborhoods, and the operational "winter jump" neighborhoods: it may be interesting to change radically an outage scheduling, from an Autumn period to a Spring period, to avoid Winter periods where the substituted production cost is higher.

This illustrated that VND-MIP allows to define easily neighborhoods induced by operational expertise. In that case, the genericity of modeling and resolution is useful to avoid specific implementations imposed by traditional local search. An application of MIP-VND scheme is also to design and test neighborhoods with few implementation efforts, to select neighborhoods that should be investigated for a proper implementation.

6.3 Sequences of neighborhoods

A key point in the Algorithm 3 is the sequence of neighborhoods. Applying Algorithm 3 with a single neighborhoods, it allows to analyze the impact of neighborhoods in the quality and number of local extrema. In this case, different neighborhoods are compared starting with the same initial solutions, and analyzing the different qualities of the local minimum where the steepest descent local search converges. Here, there is a specific interest to an analyze the quality of neighborhoods $\mathcal{N}^{TW}_{(O,1)}$, that contains widely used neighborhoods with decomposition approaches for the Challenge. It is interesting also to analyze the impact to add swap possibilities with $\mathcal{N}^{cycles}_{k,k'}$ or "winter spring" with $\mathcal{N}^{units}_{I}$.

A traditional idea with VND is to increase the size of the neighborhoods when a local minimum is reached, for a better compromise between the computation time and the local minimum reached. It can be implemented for Local Branching neighborhoods $\mathcal{N}^{LB}_{k}$ with k increasing , as developed in [23]. It is similar with neighborhoods $\mathcal{N}^{TW}_{(0,k)}$. In our case study, long computation time are allowed in the operational process for the monthly reoptimization of the maintenance planning (for instance several hours of computation by night). We do not seek to find the best solutions in short resolution time, where the previous sequencing strategy of neighborhoods is relevant. Having many types of neighborhoods, the stopping criterion is firstly to reach a local minimum for all the types of neighborhoods. The choice of neighborhoods is nested, alternating deterministically the order of neighborhoods and stopping the resolution when no neighborhood can improve the current best solution.

Using neighborhoods $\mathcal{N}^{units}_{I}$ or $\mathcal{N}^{cycles}_{k,k'}$, we can design different partitions of the binary variables. A partition of units $P \subset \mathcal{P}(\mathcal{I})$ give rise to neighborhoods $\mathcal{N}^{units}_{p}$ for $p \in P$ partitioning the variables of the problem. Many such partition can be considered, the partition units by units with $P = \{\ \{i\}\ \}_{i^i n \mathcal{I}}$ is a first natural choice, reoptimizing all the planning separately. To control the size of the MIP problems, one can consider partitions with subsets of at least k elements where k is given. In the applications, natural partition occurs in sites with several nuclear reactors, in subsets of the constraints CT14-CT15. VND iterations can also try successively to reoptimize along different partitions, similarly with the partitioning local search POPMUSIC (Partial optimization meta-heuristic under special intensification conditions, see [28]). $\mathcal{N}^{cycles}_{k,k}$ neighborhoods for all

Table 4: Characteristics of the instances ROADEF and their truncated ones to 3 cycles and 120 weeks: I,J number of T2 and T1 units, K number of cycles for all T2 unit, S number of scenarios, T,W, number production and weekly time steps, varBin denotes the number of binaries

Instances	I	J	K	S	T	W	varBin	Troncat.	varBin
A1	10	11	6	10	1750	250	3892	A1_3_120	245
A2	18	21	6	20	1750	250	7889	A2_3_120	663
A3	18	21	6	20	1750	250	8162	A3_3_120	568
A4	30	31	6	30	1750	250	17465	A4_3_120	1305
A5	28	31	6	30	1750	250	15357	A5_3_120	1868
B6	50	25	6	50	5817	277	24563	B6_3_120	1519
B7	48	27	6	50	5565	265	35768	B7_3_120	4658
B8	56	19	6	121	5817	277	69653	B8_3_120	11057
B9	56	19	6	121	5817	277	69306	B9_3_120	11146
B10	56	19	6	121	5565	265	29948	B10_3_120	1816
X11	50	25	6	50	5817	277	20081	X11_3_120	1470
X12	48	27	6	50	5523	263	27111	X12_3_120	1927
X13	56	19	6	121	5817	277	30154	X13_3_120	2838
X14	56	19	6	121	5817	277	30691	X14_3_120	2844
X15	56	19	6	121	5523	263	27233	X15_3_120	1784

k define also a partition, "orthogonal" to the unit reoptimization. Considering all the $\mathcal{N}^{cycles}_{k,k+1}$ does not define a partition of the variables anymore, but this is not a fundamental property for the VND.

7 Computational results

The computational analyses have two goals. On one hand, we are interested in methodological results, examining the limits of MIP solving and matheuristics and trying to explain some results/ranking after the ROADEF Challenge. On the other hand, industrial results analyze the impact of different modeling, e.g. the impact of CT6 constraints and of the extension with stability costs, for a better operational application. Tests were computed with a laptop running Linux Ubuntu 14.04 with an Intel Core2 Duo processor, 2.80GHz. Our implementation used the modeling language OPL to solve MIP with Cplex 12.5 and OPL script to iterate MIP computations. If this implementation is far from being optimal, this step is useful to understand better an industrial problem, to design efficient meta-heuristic operators with a low implementation effort, for a further proper implementation of selected operators.

7.1 Instances characteristics

We used the datasets provided by EDF for the Challenge, after a preprocessing to aggregate production time steps to week and the scenarios using weighted sums with respective weights the length of the time stem and the probability of occurrence of a scenario. The full data characteristics are provided in Table 4. Dataset A contains five small instances in a horizon of five years, with 10 to 30 nuclear units having 6 production cycles. Instances B and X are more representative of real-world size instances, with 20 to 30 T1 units, around 50 nuclear units. These datasets are now public, dataset X was secret for the challenge, with instance characteristics that had to be similar to dataset B. This symmetry holds for the number of units, cardinal of sets of Table 1 and number of constraints of each type, but this does not hold for the number of binaries; Table 4 shows in the column `nbVarBin` the number of binary variables, the cumulated amplitude of time windows. Instances B8 and B9 are more combinatorial, with no time window constraints for cycles $k \geqslant 3$, which are likely the most representative instances from the real-world application. A special interest is given to be able to solve efficiently instances B8 and B9, B7 is also difficult considering the `nbVarBin` indicator. Lastly, we used truncated instances planning at least 3 maintenances in the 120 first weeks of the previous instances to have easier instances and possible comparisons of heuristics to optimally proven solutions. Such instances are suffixed with `_3_120`. Table 4 shows also the number of binaries of these smaller instances.

Table 5: Comparison of termination time of B&B to optimality: without and with the exact pre-processing of [10], preprocessing and B&B warmstart with the optimal (or best known) solution, and with penalization costs to the baseline solution given by 5.1

Instances	no PP	+ PP	+warmstart	penal
A1_3_120	0,12	0,12	0,04	0,13
A2_3_120	0,31	0,29	0,27	0,58
A3_3_120	0,23	0,21	0,21	0,14
A4_3_120	1,55	0,89	0,45	0,39
A5_3_120	127	82	70,86	15,3
Total A	129,21	83,51	71,83	16,54
B6_3_120	32,3	17	13,8	4,98
B7_3_120	916,9	512	437,9	20,1
B8_3_120	>3600	1483	1461	45
B9_3_120	>3600	>3600	>3600	264
B10_3_120	60,8	52	43,8	5,4
Total B	8210	5664	5556,5	339,48
X11_3_120	60,5	40,7	21,75	5,7
X12_3_120	111	69,4	43,6	17,2
X13_3_120	27,6	22	16,2	5,98
X14_3_120	98,8	56,1	53,2	16,3
X15_3_120	79,3	14	8,7	15,4
Total X	377,2	202,2	143,45	60,58
Total	8716,41	5949,71	5771,78	416,6

7.2 MIP solving characteristics

Table 5 compares the termination time of B&B to optimality on the smallest instances: without and with the exact pre-processing of [10], the influence of warmstarting the B&B algorithm with the optimal (or best known) solution. On short instances, the B&B algorithm converges in less than one hour proving the optimality for all instances except for B8_3_120 and B9_3_120. The exact pre-processing of [10], tightening time windows thanks to constraints CT5 and CT8-11, has an strong influence in the computation times to optimality. It allows toprove the optimality in one hour for B8_3_120. Warmstarting has less influence, it is however relevant for the application in Algorithm 3.

Table 6 indicates the quality of the solutions found in one hour computation with B&B for the largest instances comparing the gaps th the Best Known Solutions (BKS) and provides the dual bounds in terms of gaps to the BKS using the results of [10] with 1h computations. These results on the deterministic average scenario justify the commonly used simplification to aggregate production time steps to weekly time steps, and to aggregate scenarios. Using Cplex 12.3, the frontal resolution is completely inefficient on instances B8 and B9, 1h resolution time is not enough to compute the LP relaxation. The sizes of these instances are a strong limiting factor. For the other instances, the frontal MIP resolution is efficient, with low gaps between the best primal and dual bounds in a 1h resolution time. Using Cplex 12.5, LP relaxations can be computed for all instances in 1h.

However, the solutions of the LP relaxation are not useful to compute primal solutions for B8 and B9. In these cases, several production cycles can overlap and the LP relaxation gives very few integer variables, specially for last cycles. Hence, this is a big handicap for MIP primal heuristics, relying on the LP relaxation such as Feasibility Pump, RINS or Local Branching, and tailored variable fixing heuristics. The MIP frontal resolution is always inefficient on B8 and B9, if LP relaxation can be computed in one hour, MIP primal heuristics are inefficient. These resolution facts explains and justify the approximations made for the Challenge by [22, 18, 27].

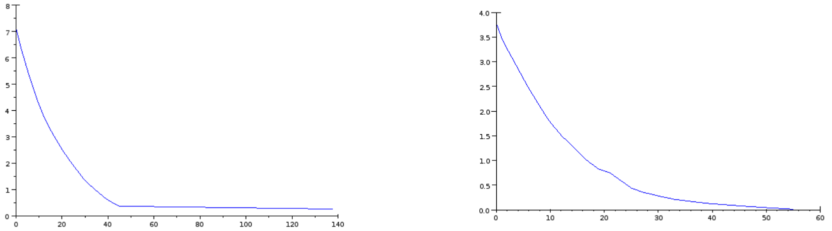

Figure 3: Instance B7-3-120

Figure 4: Instance X11-3-120

Figure 5: Pareto frontiers for the best compromises cost/stability.

7.3 Optimizing around an initial solution

In the instances from the ROADEF Challenge, there were no initial solution given. To analyze the impact of having an initial solution and minimizing with stability costs, we considered the initial solutions given by the matheuristic of Section 5.1 without the repairing post-processing. Such initial solutions are not always feasible, and are of medium quality, which seemed a good choice to represent monthly reoptimization, the current solution computed the previous month can become infeasible or improvable after the realization of real-life uncertainties.

Adding penalization to a baseline solution in the objective function changes fundamentally the MIP convergence characteristics, guiding the solution search around the baseline solution improves very significantly the resolution time, as shown in Table 5. The higher are the stability coefficients $\mathbf{C}^{pen}_{i,k,w}$, the more the B&B algorithm is accelerated. An explication is that a lot of solutions with similar ROADEF costs exist which was already noticed for the Challenge. It induces "pseudo-symetries" in the B&B tree search, which is known to be a bottleneck for the B&B search. It explains the difficulty of the B&B convergence previously outlined.

An application of these properties is to repair partial infeasible solutions built by a constructive heuristic, with a high penalization of the distance to the partial solution, similarly to Feasibility Pump ([4]).

These good B&B characteristics allow easily to considering an ε-constraint method to draw the Pareto curve of non dominated solutions in the bi-objective optimization cost/stability ([21]). We note that the granularity for the ε-constraint method can be set to $\varepsilon = 1$ along integer objectives measuring the stability penalization, which induces the whole Pareto frontier without granularity errors. Figure 5 presents such Pareto frontiers for instances `B7-3-120` and `X11-3-120` with the baseline solution given by the constructive heuristic of section 5.1. The quality of the baseline solution has a consequent impact for the Pareto frontiers, with a poor quality solution like the ones given by the constructive heuristic of section 5.1, an important trade-off exist between cost and stability, whereas good initial solutions produce a flat Pareto frontier and in the case where the baseline solution is optimal the Pareto frontier would be reduced to a single point.

7.4 Constructive matheuristics

With the characteristics of B&B search previously outlined, the matheuristic effort to design constructive approaches focused on the ability to get feasible solutions for the most difficult instances B7, B8 and B9. The numerical results of constructive heuristics are highlighted in Table 6. Simplified MIP computations of section 5.1, the CMSA and the R-R&F heuristics succeeded to provide feasible solutions for all types of instances.

Focusing the binaries of the problem, the MIP formulation of section 5.1 accelerated significantly the search of primal solutions, easing the MIP primal

Table 6: Result comparison for constructive primal math-heuristics

	BKS	Dual	Frontal	Simp	CMSA	R-R&Fix	VND
B6	76966	0,13%	0,00%	3,22%	3,8%	0,34%	0,00%
B7	74234	0,52%	0,02%	3,68%	0,75%	0,25%	0,00%
B8	73240	7,95%	NS	52,54%	52,21%	2,02%	0,00%
B9	72812	6,69%	NS	27,46%	28,95%	1,15%	0,00%
B10	69501	0,07%	0,02%	0,21%	0,2%	0,21%	0,00%
X11	73018	0,37%	0,00%	0,45%	0,53%	0,38	0,00%
X12	70604	0,20%	0,00%	0,25%	0,24%	0,31%	0,00%
X13	69231	1,09%	0,04%	7,37%	9%	0,71%	0,00%
X14	68395	0,89%	0,02%	6,02%	1,21%	0,54%	0,00%
X15	66029	0,10%	0,06%	0,12%	0,12%	0,38%	0,00%
Total	714031	0,00%	NS	10,31%	9,89%	0,63%	0,00%

heuristics. For the instances B8 and B9, feasible solutions were found in 15 minutes including the repairing post-processing. R-R&F approach gives very good solutions, but require sometimes long calculus times to converge to a feasible solution in the MIP iterations (around 20 or 30 minutes to compute the most difficult computations in intermediary cycles, for a total solving time in more than one hour). The CMSA strategy using the previous repairing strategy to build a feasible solution from partial solutions built by single unit optimization. It allows to build quickly primal solutions for all instances, but with a lower quality of primal solutions than the relax-and-fix strategies.

7.5 POPMUSIC-VND matheuristics

Table 7 compares the quality of the local minimum considering each type of single neighborhoods separately int the MIP-VND local search.

Neighborhoods $\mathcal{N}^{TW}_{(0,1)}$ include the natural neighborhood to move one single outage from one week. These neighborhoods were considered by many approaches for the ROADEF Challenge. $\mathcal{N}^{TW}_{(0,1)}$ includes also 1-translation neighborhoods, translating from one week the maintenance planning, which are also natural designing local search methods for the problem. Table 7 shows that it induces a poor quality of local minimums. Many local minimums exist with neighborhoods $\mathcal{N}^{TW}_{(0,1)}$ and also for the previously mentioned smaller neighborhoods. The quality of local minimums is improved increasing the radius with $\mathcal{N}^{TW}_{(0,k)}$ neighborhoods. However, significant gaps to the BKS remain. It justifies to consider the other types of neighborhoods in a local search method for the problem. We note that "funnel" neighborhoods $\mathcal{N}^{TW}_{(a,b)}$ with $a > 0$ are interesting instead of using only $\mathcal{N}^{TW}_{(0,b)}$ neighborhoods.

Cycle neighborhoods $\mathcal{N}^{cycles}_{k,k'}$ allow to consider swap moves that can be impossible with small $\mathcal{N}^{TW}_{(a,b)}$ neighborhoods with spacing constraints CT14-CT15. Significant improvements are observed in Table 7. Such swap moves are also possible with $\mathcal{N}^{units}_{I}$ neighborhoods, when subset I contains units in a site de-

Table 7: Quality of local minimums considering one single type of neighborhoods in the VND of Algorithm 3

Instances	Init	$\mathcal{N}^{TW}_{(0,1)}$	$\mathcal{N}^{TW}_{(1,2)}$	$\mathcal{N}^{TW}_{(0,3)}$	$\mathcal{N}^{TW}_{(1,3)}$	$\mathcal{N}^{TW}_{(0,5)}$
B6	3,22%	0,54%	0,13%	0,26%	0,13%	0,13%
B7	3,68%	1,14%	0,41%	0,47%	0,35%	0,19%
B8	52,54%	38,34%	19,70%	31,12%	16,05%	26,73%
B9	27,46%	15,91%	7,54%	12,35%	7,54%	8,89%
B10	0,21%	0,1%	0,05%	0,06%	0,02%	0,05%
Total B	17,40%	11,18%	5,55%	8,83%	4,80%	7,18%
X11	0,45%	0,31%	0,10%	0,26%	0,22%	0,27%
X12	0,25%	0,13%	0,10%	0,09%	0,07%	0,06%
X13	7,37%	2,23%	1,30%	1,47%	0,93%	1,01%
X14	6,02%	1,64%	0,48%	0,61%	0,45%	0,54%
X15	0,12%	0,08%	0,06%	0,07%	0,06%	0,06%
Total X	2,82%	0,87%	0,41%	0,50%	0,34%	0,39%

	Init	$\mathcal{N}^{units}_{I}$	$\mathcal{N}^{units}_{I}$	$\mathcal{N}^{units}_{I}$	$\mathcal{N}^{cycles}_{k,k'}$	$\mathcal{N}^{cycles}_{k,k'}$
		$\lvert\mathcal{I}\rvert = 1$	$\lvert\mathcal{I}\rvert = 5$	$\lvert\mathcal{I}\rvert = 8$	$k = k'$	$k' - k = 1$
B6	3,22%	0,26%	0,03%	0,04%	0,31%	0,13%
B7	3,68%	0,34%	0,06%	0,02%	0,55%	0,40%
B8	52,54%	1,18%	0,36%	0,36%	35,79%	27,63%
B9	27,46%	1,31%	0,05%	0,05%	14,48%	12,12%
B10	0,21%	0,05%	0,03%	0,03%	0,08%	0,03%
Total B	17,40%	0,63%	0,11%	0,10%	10,21%	8,04%
X11	0,45%	0,16%	0,09%	0,09%	0,30%	0,26%
X12	0,25%	0,08%	0,05%	0,05%	0,15%	0,10%
X13	7,37%	0,51%	0,07%	0,07%	2,43%	0,28%
X14	6,02%	0,28%	0,13%	0,13%	1,40%	0,60%
X15	0,12%	0,09%	0,05%	0,05%	0,10%	0,08%
Total X	2,82%	0,22%	0,08%	0,08%	0,87%	0,26%

fined by constraints CT14-CT15. The quality of local minimums considering only unit reoptimization with $|\mathcal{I}| = 1$ is excellent, for very quick convergence of the B&B algorithm in the MIP subproblems. It justifies to consider these neighborhoods with "winter jump" possibilities, and explains the efficiency of approaches [13, 24] for the Challenge.

Table 8 shows the final results considering in the POPMUSIC-VND neighborhhhods $\mathcal{N}^{TW}_{(1,3)}$, $\mathcal{N}^{cycles}_{k,k}$ and $\mathcal{N}^{units}_{I}$ with $|\mathcal{I}| = 5$. The VND local search starting with the CMSA solutions is very efficient, improving quickly primal solutions and requiring few VND iterations to compute a local minimum for all neighborhoods. The resulting matheuristic is more efficient in the ratio improvement of solution/computation times than the R-R&F heuristic. Furthermore, comparing the local extrema given by the VND to the optimal solutions of small instances, we had no example where the optimal solutions were better than the VND solutions more than the tolerance gap of Cplex parametrized to $0,01\%$. The tolerance gap induces sometimes that the VND solutions were better than the "optimal" frontal solutions, within the $0,01\%$ tolerance gap.

Table 8: Comparison of the gaps to the BKS for the POPMUSIC-VND and for the VND with single types of large neighborhoods

Instances	Initial Solution	$\mathcal{N}^{TW}_{(1,3)}$	$\mathcal{N}^{cycles}_{k,k+1}$	$\mathcal{N}^{units}_{I}$ $\lvert\mathcal{I}\rvert = 8$	POPMUSIC VND
B6	3,22%	0,13%	0,13%	0,04%	0,00%
B7	3,68%	0,35%	0,4%	0,02%	0,00%
B8	52,54%	16,05%	27,63%	0,36%	0,00%
B9	27,46%	7,54%	12,12%	0,05%	0,00%
B10	0,21%	0,02%	0,03%	0,03%	0,00%
Total B	17,4%	4,8%	8,04%	0,1%	0,00%
X11	0,45%	0,22%	0,26%	0,09%	0,00%
X12	0,25%	0,07%	0,1%	0,05%	0,00%
X13	7,37%	0,93%	0,28%	0,07%	0,00%
X14	6,02%	0,45%	0,6%	0,13%	0,00%
X15	0,12%	0,06%	0,08%	0,05%	0,00%
Total X	2,82%	0,34%	0,26%	0,08%	0,00%

7.6 Adding CT6 constraints

Adding the lighter CT6 constraints in the MIP formulation, the solving capabilities are more limited. Computations were not possible for the real size instances of datasets B and X, inducing memory errors. The computations for the smallest instances showed that the number of variables is multiplied by a factor 4 adding the CT6 constraints, and the preprocessing of Cplex maintain this factor 4, reducing very significantly the LP computations. the characteristics of the B&B convergence highlighted for the MIP formulation without CT6 constraints are similar, the bottleneck is in the LP computations.

We note that light CT6 constraints with (20-21) or with (21-22) has few implications in terms of number of variables, adding $I \times S \times T$ continuous variables $x_{i,w}$, whereas $I \times S \times T \times K$ continuous variables were already in the model with T2 productions $p_{i,k,w}$. The main difference appears in the number of constraints, there were mainly $I \times S \times T \times K$ constraints in the MIP of Section 4 with constraints (7), (21-22) require to add $I \times S \times T \times (K + M)$ constraints, whereas (20-21) require to add $I \times S \times T \times (M + 1) \times K$ constraints. The aggregated formulation (22) provided improvements in the computation times compared to (20), but the formulation is still intractable for real size instances. Adding CT6 constraints (21-20) or (21-22) has furthermore very few impact in the quality of the LP relaxation.

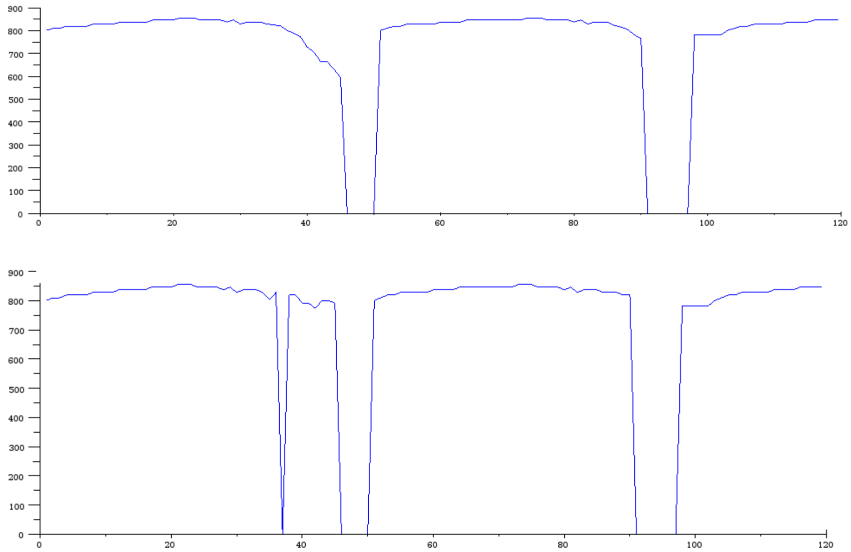

Figure 6: Impact of the CT6 constraints in a case where it is activated: upper case consider upper bounds of productions whereas the lower case is a full relaxation of CT6 constraints

A first issue is to analyze whether it is insightful to consider only upper bounds for the production in decreasing profile phases. Actually, decreasing profile phases are not often activated. An explanation is that the upper bounds restrict the nuclear production capacities. The nuclear units have the lowest marginal costs of production, the financial optimization avoids the over-costs of production with mainly maximal nuclear productions. In that goal, maximizing the nuclear production available tends to not use the decreasing profile phases, preferring to have earlier outages to be able to produce at the maximum power in the production cycles. Unavoidable and optimal decreasing profile uses occur when time windows constraints impose a cycle to be longer than the duration of maximal power before reaching $\mathbf{Bo}_{i,k}$, which is the case in the situation met to have a decreasing profile illustrated in Figure 6. In such cases, the financial optimization tends actually to produce at the upper bound of production thanks to the difference in marginal costs between T1 and T2 units, which justifies the lighter formulation of section 4.4.

A second issue is to analyze the impact of considering CT6 constraints in the optimal outages dates compared to the model of section 4.2 with a full relaxation of CT6 constraints. The over-costs to project the optimal (or best known) solutions computed thanks to the MIP of section 4.2 into a MIP with stretch constraints is given for the small instances in Table 9. A little over-cost is observed , around 0.3% over-cost in average. It justifies that the CT6 constraints has no incidence on the feasibility of the outage planning, and that the full

Table 9: Over-Costs of the outage solutions with the MIP of section 4.2 in the model of section 4.4 with CT6 constraints, and local improvements with two iterations of VND with neighborhoods $\mathcal{N}^{TW}_{(0,2)}$ using formulation 4.4.

	BKS	Projected 4.2	+LS
A1_3_120	7612M	0,01%	0,00%
A2_3_120	7213M	0,01%	0,00%
A3_3_120	7638M	0,02%	0,00%
A4_3_120	5 122M	0,05%	0,00%
A5_3_120	5 852M	0,08%	0,03%
B6_3_120	31 950M	0,10%	0,00%
B7_3_120	30 364M	0,18%	0,06%
B8_3_120	29 438M	0,14%	0,13%
B9_3_120	29 239M	0,18%	0,00%
B10_3_120	27 957M	0,34%	0,06%
X11_3_120	30 199M	0,20%	0,00%
X12_3_120	29 669M	0,25%	0,00%
X13_3_120	29 148M	0,36%	0,00%
X14_3_120	28 438M	0,23%	0,00%
X15_3_120	27 722M	0,37%	0,00%

relaxation of CT6 constraints produce solutions of very good quality. Figure 6 suggests that some local modifications around the CT6 situations can improve the previous projected solution. Table 9 furnishes also the improvements of the projected solutions with only two iterations of local search with neighborhoods $\mathcal{N}^{TW}_{(0,2)}$ using formulation 4.4. Indeed, some very local modifications around the CT6 situations improve significantly the projected costs.

These conclusions justify the simplified models considered by [22] to relax fully the CT6 constraints to have good quality solutions with a simpler MIP model considering binaries only for the outage decisions.

8 Conclusions and perspectives

To optimize refueling and maintenance scheduling of nuclear power plants, this paper investigated matheuristic approaches. To face the limits of straightforward B&B search, constructive matheuristics generate more consistently feasible solutions solving smaller problems in decomposition schemes. Once feasible solutions are built, a local search matheuristic POPMUSIC-VND leads to solutions with outstanding quality. Such approach requires reasonable computation times for an industrial application.

Intermediate results are of interest for the industrial application, and explains the ranking of the EURO/ROADEF 2010 Challenge. This article justified some approximations made by [22, 27, 18] to have tractable MIP formulations to face the large size of real size instances: aggregation of production time steps to week, at least partial aggregation of stochastic scenarios. The relaxation of the constraints CT6 are also justified in this study, studying the compromise between solution quality and computation times. The analysis of neighborhoods explains why many decomposition approaches failed for the challenge: they were dealing with too short neighborhoods. Our matheuristic study explains why approaches [13, 24] were more efficient: using no time consuming LP, MIP or CP sub computations, more time could be devoted to explore solutions, and the use of variable and larger neighborhoods than $\mathcal{N}^{TW}_{(0,1)}$ for a better exploration. Especially, the "Winter Jump" neighborhoods are crucial. These results open perspectives for the development of an operational prototype.

Using matheuristics allows also a genericity in the objective functions. An extension with bi-objective optimization was designed in this study to find the best compromises solutions between the planning stability and the financial costs. As a perspective, other objectives can be considered: maximizing robustness to lessen financial risks, or sustainable development considerations like minimizing nuclear wastes or minimizing carbon emissions.

References

[1] T. Adamo et al. MIP neighborhood synthesis through semantic feature extraction and automatic algorithm configuration. *Computers & OR*, 83:106–119, 2017.

[2] D. Anghinolfi, L. Gambardella, et al. A matheuristic algorithm for a large-scale energy management problem. *Lecture Notes in Computer Science*, 7116:173–181, 2012.

[3] T. Benoist, B. Estellon, F. Gardi, R. Megel, and K. Nouioua. Localsolver 1. x: a black-box local-search solver for 0-1 programming. *4OR*, 9(3):299–316, 2011.

[4] L. Bertacco, M. Fischetti, and A. Lodi. A feasibility pump heuristic for general mixed-integer problems. *Discrete Optimization*, 4(1):63–76, 2007.

[5] C. Blum, Pedro P., M. López-Ibáñez, and J. Lozano. Construct, merge, solve & adapt a new general algorithm for combinatorial optimization. *Computers & Operations Research*, 68:75–88, 2016.

[6] F. Brandt. Solving a large-scale energy management problem with varied constraints. Master's thesis, Karlsruhe Institute of Technology, 2010.

[7] F. Brandt, R. Bauer, et al. A constraint programming-based approach to a large-scale energy management problem with varied constraints. *Journal of Scheduling*, 16(6):629–648, 2013.

[8] E. Danna, E. Rothberg, and C. Le Pape. Exploring relaxation induced neighborhoods to improve MIP solutions. *Math Program*, 102:71–90, 2005.

[9] N. Dupin. Tighter MIP formulations of the discretised Unit Commitment Problem with min-stop ramping constraints. *EURO Journal on Computational Optimization*, 5(1-2):149–176, 2017.

[10] N. Dupin and E. Talbi. Dual heuristics and new dual bounds to schedule the maintenances of nuclear power plants. *arXiv preprint arXiv:1806.00445*, 2018.

[11] N. Dupin and E-G. Talbi. Dual Heuristics and New Lower Bounds for the Challenge EURO/ROADEF 2010. In *Matheuristics 2016, 6th Int. Workshop on Model-based Metaheuristics*, pages 60–71, 2016.

[12] M. Fischetti and A. Lodi. Local branching. *Math Programming*, 98(1-3):23–47, 2003.

[13] F. Gardi and K. Nouioua. Local search for mixed-integer nonlinear optimization: a methodology and an application. *Lecture Notes in Computer Science*, 6622:167–178, 2011.

[14] H. Gavranović and M. Buljubasić. A hybrid approach combining local search and constraint programming for a large scale energy management problem. *RAIRO Operations Research*, 47(4):481–500, 2013.

[15] S. Godskesen, T. Jensen, N. Kjeldsen, and R. Larsen. Solving a real-life, large-scale energy management problem. *Journal of Scheduling*, 16(6):567–583, 2013.

[16] M. Guzelsoy, G. Nemhauser, and M. Savelsbergh. Restrict-and-relax search for 0-1 mixed-integer programs. *EURO Journal on Computational Optimization*, 1(1-2):201–218, 2013.

[17] C. Joncour. *Problèmes de placement 2D et application à l'ordonnancement: modélisation par la théorie des graphes et approches de programmation mathématique*. PhD thesis, Université Bordeaux, 2010, pp 147-166.

[18] V. Jost and D. Savourey. A 0-1 integer linear programming approach to schedule outages of nuclear power plants. *Journal of Scheduling*, 16(6):551–566, 2013.

[19] M. Khemmoudj. *Modélisation et résolution de systèmes de contraintes : application au problème de placement des arrêts et de la production des réacteurs nucléaires d'EDF*. PhD thesis, PhD Thesis Paris 13, 2007.

[20] H. Larrain et al. A variable MIP neighborhood descent algorithm for managing inventory and distribution of cash in automated teller machines. *Computers & OR*, 2017.

[21] M. Laumanns, L. Thiele, and E. Zitzler. An efficient, adaptive parameter variation scheme for metaheuristics based on the epsilon-constraint method. *European Journal of Operational Research*, 169(3):932–942, 2006.

[22] R. Lusby, L. Muller, and B. Petersen. A solution approach based on benders decomposition for the preventive maintenance scheduling problem of a stochastic large-scale energy system. *Journal of Scheduling*, 16(6):605–628, 2013.

[23] N. Mladenović, P. Hansen, and D. Urozević. Variable neighborhood search and local branching. *Computers & OR*, 33(10):3034–3045, 2006.

[24] J. Peekstok and E. Kuipers. Roadef/euro 2010 challenge. *Be Improved Tech report*, 2010.

[25] M. Porcheron et al. Challenge ROADEF/EURO 2010: a large-scale energy management problem with varied constraints. *EDF R&D Technical Report*, 2010.

[26] D. Rajan and S. Takriti. Min-Up/Down Polytopes of the Unit Commitment Problem with Start-Up Costs. Technical report, IBM Research Report, 2005.

[27] A. Rozenknopf, R. Wolfler Calvo, et al. Solving the electricity production planning problem by a column generation based heuristic. *Journal of Scheduling*, 2012.

[28] É. Taillard and S. Voss. POPMUSIC-Partial optimization metaheuristic under special intensification conditions. In *Essays and surveys in metaheuristics*, pages 613–629. Springer, 2002.