Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 371–378
ISBN 978-83-60810-51-4

Orchestrating Constrained Programming and Local Search to Solve a Large Scale Energy Management Problem

Mirsad Buljubašić
Faculty of Natural Sciences, University of Sarajevo
Sarajevo, Bosnia and Herzegovina
Email: mirsad_bulj@yahoo.com

Haris Gavranović
International University of Sarajevo
and Strategeme laboratory
Sarajevo, Bosnia and Herzegovina
Email: haris.gavranovic@gmail.com

***Abstract*—This paper presents a heuristic approach combining constraint satisfaction, local search and a constructive optimization algorithm for a large-scale energy management and maintenance scheduling problem. The methodology shows how to successfully combine and orchestrate different types of algorithms and produce competitive results. The local search for production assignment is a simple yet optimal solution for the relaxed initial problem. We also propose an efficient way to scale the method for huge instances. A large part of the presented work is done to compete in the ROADEF/EURO Challenge 2010, organized jointly by the ROADEF, EURO and the Électricité de France. The numerical results obtained for the official competition instances testify about the quality of the approach. The method achieves 3 out of 15 possible best results.**

I. Introduction

IN THIS work we take the perspective of a large utility company, tackling their problems in modeling and planning production assets, i.e., a multitude of power plants. The goal is to fulfill the respective demand of energy over a time horizon of several years, with respect to the total operating cost of all machinery. Determining optimal maintenance schedules and production plans is not easy because of the number of alternatives to assess. The scheduling of outages has to comply with various constraints, regarding safety, maintenance, logistics and plant operation while it must lead to production programs with minimum costs. The proposed subject consists of modeling the production assets and finding an optimal outage schedule and includes two mutually dependent and related sub problems:

1) Determining the schedule of plant outages. This schedule must satisfy some additional constraints in order to comply with limitations on resources, which are necessary to perform refueling and maintenance operations.
2) Determining an optimal production plan to satisfy demand, i.e. a quantity of energy to be produced by each plant at each time step, for each scenario. Production must also satisfy some technical constraints.

In medium-term electricity management, numerous uncertainties have to be taken into account (demand, generation units availability, spot market prices, quantities that can be bought or sold), and that leads to the need of considering multiple scenarios. The objective is to minimize the expected cost of production. The full and detailed description of the problem is given in [9] together with 15 official instances and an official solution checker. This problem was proposed at the ROADEF/EURO Challenge 2010, a competition announced by the French Operational Research and Decision Support Society (ROADEF) and the European Operational Research Society (EURO). Our solution combines constraint programming and local search approaches.

The company portfolio includes different kinds of energy production such as thermal (nuclear, coal, fuel oil and gas), hydraulic and other renewable energies. Thermal power facilities usually produce about 90 percent of energy, of which about 80 percent is produced by nuclear power plants. Therefore, this subject is focused on thermal power plants, especially on those that have to be repeatedly shut down for refueling and maintenance, e.g. nuclear ones.

In what follows, we use, when possible, the same names for data, decision variables and indices as in the subject (see [9]). On the other hand we need several new kinds of notation and new constructions. The second section introduces them while giving an idea how they would be useful. The third section outlines the proposed algorithm explaining its coarse structure first. Thereafter, in the same section, we describe in detail the most important subprocedures that make the whole algorithm. The fourth section presents possible improvements of the solution, playing with parameters and solutions given by either the constraints satisfaction solver or the linear programming solver. This section also aims to assess the quality and importance of the building blocks of the proposed solution and measures the effects of different synergies between these blocks. The fifth section presents the numerical results for the given ROADEF Challenge instances and compares them with best known, public results.

II. Notation, assumptions and reductions of the problem

In this section we introduce several new kind of notation and simplifications of the problem that proved to be helpful either for the presentation itself or they improved the quality and the efficiency of the solution. The complexity of the problem, the extraordinary size of certain instances and time limits imposed on the solver set a stage for these simplifications and reductions. Moreover, they often become necessary just to solve to the feasibility the most difficult instances of the problem.

The objective function consists of three parts that in a more or less obvious and direct way influence each other. For official instances A the part of the cost associated with the production cost for Type-1 plants differs by an order or two of magnitude with respect to the part associated with the fuel cost of Type-2 plants. In bigger instances B and X these two costs are comparable. As the cost associated with the nuclear plants (Type-2 plants) in most of the given instances is smaller than the cost incurred with Type-1 plants and in a real life the nuclear energy is less expensive than the energy from conventional energy plants, we assume that Type-2 power plants are in general cheaper than Type-1 power plants. This assumption is used throughout the production planing algorithm. Nevertheless, this planning for Type-2 plants has a direct influence on the total cost of energy produced by Type-1, i.e. conventional plants.

We perform simple pre-processing of input data in order to simplify the discussion and the implementation of the solution. In fact, some of the Type-1 power plants have a positive lower bound on the production level for each time step in each scenario. The problem is simplified by decreasing the customer demand by the sum of these lower bounds over all power plants while setting the new lower bound on production to zero. This alternation of input data obviously does not change the structure of the initial problem. From now on, we assume that these lower bounds are equal to zero for all instances and the demand is properly changed.

A. Marginal cost associated to the time step

The demand for each time step in each scenario will be satisfied using both types of power plants. In order to have a smaller overall cost, demand should be distributed among the cheapest power plants. Since Type-2 power plants are, in general, cheaper, their production levels should be set to the maximum. At the same time, most of the outages must be scheduled (constraints $CT13$) and the constraints on refueling, in their own way, determine the production and the refueling on Type-2 plants. For some time steps and some scenarios, one part of the demand has to be distributed among Type-1 power plants. The problem of determining the production on Type-1 power plants, given the total production plan for the set of Type-2 power plants, gives rise to a very simple optimization problem similar to the relaxed knapsack problem. The solution consists of sorting the plants according to their respective production cost and fully employing them one by one, until the demand is satisfied. The cost of the last employed power plant deserves the particular name, i.e. the marginal cost associated to the time step. If the demand would increase (decrease) by one in a given time step, than the total cost of production would increase (decrease) by the marginal cost associated to that time step. This notion of the marginal cost can be extended to the marginal cost of distance as follows :

The sum of $MC_{ts}(0)$ over all time steps is an approximation of the objective function. The more important the cost associated to the Type-1 power plants in the total cost of production (*i.e.* instances A) the more accurate the approximation. For instances A this notion becomes one important ingredient of the solution. Notice that the quality of the outage schedule could be measured by the associated marginal cost.

B. Decreasing the problem size

One of the things that makes this problem very hard is the size of its instances. As the problems may contain a huge number of variables, it is an advantage both with respect to computational time and memory consumption to reduce the problem size. We propose two natural ways to reduce the size of the problem while keeping some of its original features.

a) Shrink a set of time steps to one : This is a simple, straightforward manner to reduce the size of an instance by decreasing the total number of time steps. The demand, maximal power and cost of Type-1 plants, maximal power of Type-2 plants vary over the set of time steps and the set of scenarios. For example, the variation of the demand in a given scenario is mostly determined by the weather forecast and historical data for this period. We will assume that these variations are not big and that we could approximate well these values (demand, maximal power, cost etc.) over several consecutive time steps. On the other hand, the important constraints on the set of outages are most often given in terms of weeks and it is also important to respect their structure in newly created, reduced instances. The reduction consists of shrinking several consecutive time steps into one while respecting their appartenance to the corresponding weeks. Therefore, the simplest reduction would be to shrink all time steps from one week into one while calculating for all scenarios:

- the average demand over all time steps in the original week
- the average max power over all time steps in the original week for each power plant
- the average cost over all time steps in the original week for each Type-1 power plant

and set these numbers as the new values for demand, max power and cost. Setting the length of new time steps to be equal to the total length of time steps in the week, while keeping all other parameters like fuel levels, fuel constraints, outages constraints the same, even the values of objective functions of normal and shrinked instances turned out to be comparable and fairly close. In Table II we compare the size of shrinked instances to real instances and corresponding costs when every single week is shrunk to one time step. If the length of week is a composite number, which is the case for B and X instances, we can shrink just a fraction of a

Shrinked instances				
instance	size(MB)	shr. size(MB)	cost	shr. cost
dataB6	139.9	5.6	8.38e10	8.13e10
dataB7	144.3	5.6	8.20e10	7.93e10
dataB8	262.0	10.3	8.75e10	8.34e10
dataB9	262.0	10.3	9.00e10	8.49e10
dataB10	251.7	9.8	7.98e10	7.51e10
dataX11	140.0	5.5	7.97e10	7.71e10
dataX12	143.2	5.5	7.85e10	7.50e10
dataX13	262.1	10.4	7.78e10	7.42e10
dataX14	262.1	10.4	7.80e10	7.42e10
dataX15	249.8	9.7	7.64e10	7.09e10

TABLE I

FOR EXAMPLE, THE INSTANCE HAVING 250 WEEKS, 1750 TIME STEPS (WEEK LENGTH = 7) AND TIME STEP DURATION 24, THE SHRINK OF A WEEK INTO ONE TIME STEP RESULTS INTO AN INSTANCE WITH 250 WEEKS, 250 TIME STEPS AND TIME STEP DURATION 168. SHRINKED INSTANCES ARE MUCH SMALLER AND EASIER TO SOLVE. NUMERICAL EXPERIMENTS HAVE SHOWN HOW THIS REDUCTION IS USEFUL PROVIDING THE FIRST COMPETITIVE RESULTS VERY FAST. THE TABLE SHOWS THE SIZE OF SHRINKED INSTANCES IN MB AND THE SIZE OF REAL INSTANCES IN MB. THE LAST TWO COLUMNS PRESENT ASSOCIATED COSTS FOR OBTAINED SOLUTIONS.

week into one time step. In this way the resulting reduced instances are slightly larger, but at the same time they are better approximations for the real instances.

b) Aggregating scenarios: Our approach does not always scale well with the size of problem. We experimented, therefore, with other ways to reduce the problem and the approach trying to decrease the total number of scenarios proved to be the most useful. The possible number of scenarios is up to 500 and this reduction becomes indispensable for certain instances. It is possible to choose several existing scenarios as the representatives or to create new scenarios, aggregating the existing ones on a weekly or daily basis. Further, one can eventually associate weights different from one to each scenario, giving to some of them more importance. Several strategies proved to be useful either for experimentation or for effective resolution : minimum demand scenario, maximum demand scenario and average demand scenario.

III. SOLUTION

The proposed solution of the problem is divided into the outage scheduling phase and the power assignment phase applied first in a consecutive manner. Once the feasible solution is found for the complete problem it is then improved using a series of local improvements. The newly obtained locally optimal solution gives rise to a new outage scheduling phase and the cycle repeats. This would be, in short, a description of the solution we proposed here. The complexity of the problem as whole and the nature of constraints were important when we decided to apply this approach. The set of constraints can be divided in a natural way in two subsets: one dealing exclusively with outages of Type-2 power plants (constraints CT13 to CT21) and the other consisting of constraints imposed on the level of production and the fuel consumption in all power plants (constraints CT1 to CT12). The first set of constraints is the outage constraints set, denoted OCS and

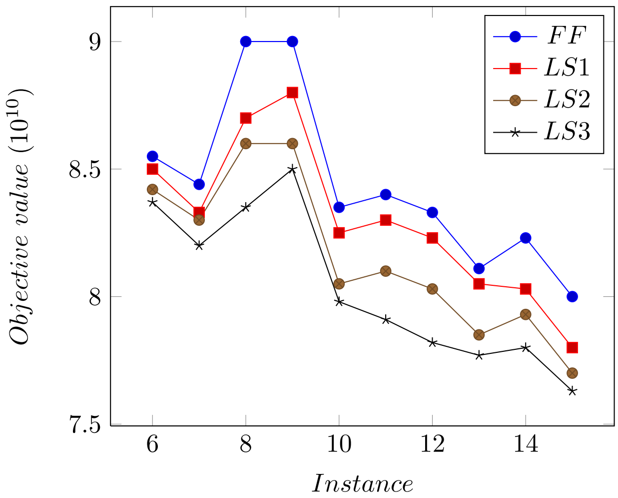

Fig. 1. This figure illustrates solution improvement process in the first iteration of the algorithm on data set B and data set X instances (B6-B10, X11-X15). The blue line presents the first feasible solution of the problem (FF) while the red, green and black lines present the solutions after marginal cost (LS1), tuning fuel levels (LS2) and moving outages (LS3) local improvements respectively. Note also that all these improvements contribute together to the overall quality of the solution in every single iteration.

the second one, the production constraints set, is denoted PCS. Note that OCS only consists of discrete constraints that one usually solves using CSP ad-hoc techniques or an appropriate constraint satisfaction solver, while PCS often mostly looks like the set of linear constraints and one could use different approaches such as linear programming or greedy algorithms to find feasible solutions or even ones close to optimal solutions. Of course, the constraints CT5 and CT6 are nonlinear and make the problem difficult in a certain way. Still, the outages scheduling problem and production planing problem are both hard to solve to optimality, at least theoretically. The drawback of this approach is also that we are not able to propose an exact objective function for the outage scheduling problem. Local improvements applied to the feasible solution consider how to improve the refueling levels for the set of outages, power level of Type-2 power plants in given production campaign and the scheduling of the set of outages respectively.

The solution therefore contains three different parts applied in this order :

1) Constraint programming(CSP) - used to construct a feasible outages schedule
2) Constructive production setting procedure - used to construct a feasible solution for a given schedule
3) Local Search - to improve the solution quality

Here we describe the details of all three building blocks of the proposed solution.

A. Outages scheduling

Different approaches can be used to solve the outages scheduling problem such as local search, constraint programming or constructive procedure, but using CSP solver emerged to be the most reliable, and probably the most elegant approach, with very good results. The CSP solver search for outages scheduling is guided, when possible and proved to be useful, with different objective functions.

Type-1 power plants are not scheduled for outages, therefore, only Type-2 power plants are considered. The set of constraints OCS is modeled using the OPL Optimization Programming Language offered within IBM Ilog suite. An outage is modeled with an interval decision variable characterized by its start date and its length. OPL provides basically all necessary constructions to efficiently and easily represent every single OCS constraint. For instance, using the elementary cumulative function IloPulse to express the group constraint $CT19$ is straightforward. We performed the test and solved the outage scheduling problem with different solvers (Comet, IBM Ilog CP, Mistral) and they were all capable of efficiently solving official instances. The solution time varies from a fraction of second up to several seconds on the standard test machine for smaller data sets and a few minutes on larger ones. The resulting schedule is valid with respect to the outages scheduling constraints $CT13$ to $CT21$.

The obtained solutions for outages scheduling suffer from a lack of diversification on one hand and on the other hand from the fact that these schedules do not always allow a feasible production plan. This is why further constraints are added to the model to ensure the feasibility of the remaining production assignment problem, which is the next step of the algorithm and will be discussed later. A certain amount of fuel has to be consumed in every production campaign to reach the fuel level limits that apply before or after the upcoming refueling (see $CT11$). Thus, based on the existing outage schedule and approximate unfeasible production plan, necessary spacing constraints are calculated and added between every two successive outages. Besides the approximate, unfeasible production, several other characteristics of the problem, such as the minimum refueling and maximum power contribute to calculating a good estimate for minimum spacing. Every solution with the $CT11$ constraint violated mandates the addition of a new set of increased spacings between outages and the CSP solver is used again. This procedure repeats until the feasible outage schedule and production is available. This quest for feasibility comes at a cost, but the number of these iterations is usually small and ranges from 0 to 5 on a provided set of instances.

Simple outages scheduling described above gives outages dates satisfying the set of constraints from $CT13$ to $CT21$ for which a realizable production plan exists. The outages scheduling from the very beginning of the solution procedure could be optimized with respect to the value of an objective function. The marginal cost introduced in II-A and the numerical experiments endorse the scheduling, when possible, of outages in the weeks of low demand over the scheduling of outages in weeks where the demand is high. The solver is guided to these solutions either by specifying the appropriate objective function or by using a set of weeks of low demand as the starting search point, or a combination of the two (see [6]). This simple heuristic is used exclusively in the beginning when no solution for the whole problem is available. It does not come as a surprise that this guided search shows better numerical results than randomized solutions for the outage schedule problem. In the advanced stages of search, when feasible or even good solutions are available, the weeks with the smallest marginal cost are used as the starting point for the solver. This will be explained in detail later.

Algorithm 1 Scheduling Outages

```
Calculate successive outages spacings approximately using
min refuel max power strategy
Find outages schedule using solver
Greedy Production Planning
while Feasible Production Plan doesn't exist do
   Increase distance between two successive outages if CT11
   is violated
   Find outages schedule using solver
   Greedy Production Planning
end while
```

B. Production Assignment

The next step in constructing the solution of the problem is production planning. This part of the solution assigns production levels for all plants and refueling amounts in all outages. For each outages schedule given by the solver, the constructive production setting procedure proposes a solution. This solution is not always feasible in which case new additional constraints are added to the CSP model and solver is used again. The algorithm, in a greedy manner, assigns as much production as possible to the cheapest plants until the demand is covered. Although cheap production is the ultimate goal, sometimes a Type-2 power plant has to be utilized at any cost to comply with the given production constraints. We assume, that it is desirable for each Type-2 power plant to produce as much as possible and adjust the refueling strategy accordingly. Using the same assumption the algorithm chooses a big refueling amount in order to increase Type-2 power plant utilization. Quite the opposite happens to the objective function when a certain amount of fuel is lost during refueling because the fuel stock before refueling was too big. To summarize, the refueling strategy is to choose the minimum refuel amount at each outage and increase it when the plant enters imposed decrease profile, if possible. The algorithm successfully constructs feasible refuelling and production plans in all provided instances. Further refueling optimization will be presented later.

The following problems with feasibility can occur while setting up the production level for Type-2 power plants :

- overproduction(total Type-2 plants production is bigger than customer demand)
- over modulation(CT12)
- fuel stock too big before or after refueling(CT11)

Overproduction can occur if total Type-2 power plants production extends over the customer demand at a certain time step for a certain scenario. In fact, if each Type-2 power plant works to its maximum at each time step overproduction problem can surge on all B and X instances. This means that at certain time steps (where the demand is relatively small) Type-2 power plants could not be powered to their maximum. The problem is then to choose some Type-2 power plants and drive them to use the associated modulation on the production. On the other hand, solving the overproduction problem by using modulation can lead to the problem that the production on some Type-2 power plants results in bigger modulation than it is allowed (imposed by $CT12$). The total modulation necessary to cope with the overproduction problem should be properly distributed among Type-2 plants. Simultaneously, this way of solving the overproduction problem can lead to a problem with the violation of fuelstock before (or after) refueling constraints. The order of Type-2 power plants, while setting their production levels is crucial for these problems as they happen while setting production levels for the last few power plants. The algorithm handles this by sorting Type-2 power plants before setting production for each time step. The power plants for which these problems are highly probable are found and then place on the top of the list for production setting procedure. The priority is given to the plants with decrease profile at a given time step, to the power plants with high modulation before given time step or to the power plants that must have high production levels during given production campaign to satisfy constraint $CT11$. This is done for each time step independently (see 3). Overproduction and the associated problems do not occur in A instances, which makes them relatively easier to solve. The algorithm runs in ascending order of time steps and for all scenarios in parallel. This way, it is easy to adjust the fuel level using the level from the previous time step. When Type-2 power plants production levels are assigned, remaining demand is distributed among Type-1 plants which are previously sorted by production cost in increasing order. Presented production assignment procedure (2) efficiently finds the initial, feasible production plan for all 15 provided data sets.

Algorithm 2 Production setting procedure

```
while solution unfeasible do
  Set refueling amounts to minimum
  for t = 0 to T do
    Sort Type-2 power plants using sorting procedure
    for j = 0 to J do
      Set power plant j production level at time step t
      to maximum possible with adding fuel on previous
      outage(if possible) if plant will enter imposed power
      profile in the next time step (respecting min power
      and max power constraints)
    end for
    Sort Type-1 power plants according to production cost
    at given time step in increasing order and set produc-
    tion levels to maximum possible (respecting demand
    constraint)
  end for
  if CT11 violated then
    Add spacing constraints and find new outages schedule
  end if
end while
```

Algorithm 3 Sorting Type-2 power plants at time step t

```
list1 = empty
list2 = empty
for j = 0 to J do
  if power plant j in decrease profile at time step t then
    Add power plant j to list1
  else
    Add power plant j to list2
  end if
end for
Sort power plants in list2 in increasing order by maximum
amount of modulation they can have for given production
campaign and still satisfy CT11 and CT12
return Merge(list1,list2)
```

C. Local Improvements

Even for small instances of the problem, good approximation of an optimal solution could be hard to achieve. In some cases just to find a feasible solution proves to be laborious task. The internal difficulty of the problem is due to, on one hand, the extraordinary size of instances, and on the other hand the presence of non-linear constraints on the production and on the outages schedule. The non-linear changes of fuel consumption and refuel in Type-2 plants make, in a sense, even objective function non-linear. Our solution tackles these non-linearities with three carefully designed and tested types of local improvements. The local improvements often substantially improve the feasible solution when applied all alone but also mutually influence each other in a positive way. Two strategies optimize the production plan, the first improving the refueling process while the other directly seeking better power settings of plants. The third one moves outages locally and thus improves the outages schedule quality, and finally the value of the overall objective function.

Here we present three local search strategies to optimize the solution.

c) Local Search based on marginal cost: This procedure tends to improve the feasible solution modifying the production levels at only two time steps at once, of the same Type-2 power plant. Both time steps should be inside a single production campaign. The fuel levels and the refueling amounts at the beginning and at the end of the campaign do not change. The production settings for all other Type-2 power plants do not change either. The change in the production of Type-2 power plants, therefore, dictates the adjustment of the production levels of Type-1 power plants and only at two appropriate time steps. The overall cost of the solution changes only according to the change at the production levels of Type-1 power plants. Here, the notion of marginal cost proved to be useful, both to find interesting pairs of time steps and to determine the scope of change.

Consider a given Type-2 power plant j such that there is a time step ts_1 where it is not employed with maximal power, and it is not in decreasing profile. Suppose we have another time step ts_2 in the same production campaign such that $MP_{ts_2}(0) \leq MP_{ts_1}(0)$ (the cost at ts_1 on the most expensive employed Type-1 power plant is bigger than the corresponding cost at ts_2). It is simple to check that it will be profitable to increase the power of j at ts_1 for some small δ and decrease the power for δ at ts_2. The fuel consumption in the production campaign of power plant j will not change. It will change only the power levels of Type-1 power plants at ts_1 and ts_2 such that demand constraint is satisfied. The total cost will decrease by $(MP_{ts_1}(0) - MP_{ts_2}(0)) * \delta$. The bigger δ the better approximation is. On the other hand, this formula will not be necessary exact for a big δ but will still remain an approximation of the change.

The whole procedure is carefully employed and optimized using a preprocessed list of good candidates for change. A similiar change could be devised for pairs of time steps belonging to different production campaigns. The implied time needed to estimate the quality of the change becomes prohibitive, and this is not kept in the proposed solution.

It would not be difficult to prove the optimality of this local search for the instances where demand is always bigger than the sum of powers on all available Type-2 power plants. In the presence of low demand this remains an approximation method with fairly good results.

d) Tuning the Refuel Levels: It is not obvious how to determine the best refueling amounts. The change in one refuel propagates through all production campaigns and the presence of decreasing profile makes these changes non-linear. Production assignment procedure sets up initial, satisfactory good amounts of refuel at each outage. These amounts are further optimized with respect to the total cost of the solution using small incremental changes. A random Type-2 power plant i and a random outage j are chosen and the difference in overall cost for a small changing refuel amount δ is calculated. If the solution improves, the change is made and the process continues. Different, positive and negative, values for δ are chosen. The whole process continues until the local optimum is met.

e) Local improvements for outages schedule: These local improvements prove to achieve the most valuable search in terms of the quality of the solutions. At the same time, the idea to move around one or several outages, exploring the search space, is also the most natural one. This exploration comes at a cost that consists mainly of the evaluation of the improvement (or deterioration) of the solution. Changing the start time of the outage will also change the fuel levels at the beginning and at the end of the production campaign, production levels of other plants (usually Type-1 power plants) as well as some other less important characteristics of the solution. All these changes will affect the overall solution cost. The move and the associated production assignment have to be feasible to be accepted. In the developing and testing phase the exact cost of the new solution is calculated and compared with the actual one. When there is an improvement the move is accepted and effectuated. These moves on at most two outages are examined simultaneously. Creating the whole solution and comparing its cost is time consuming. In the deployed solver the quality of the moves of the outages is estimated using the changes in marginal cost and the estimation of the implied refuel change. In this way, the search becomes efficient and complies with the time constraints imposed by the rules of the Challenge.

The order of selection of the outages to be moved influences the final quality of the solution. We tried the following approaches :

- Choose a random outage and a random move and evaluate it
- Choose an outage and a move that decrease the total cost at most
- Choose randomly one of several best outages
- Choose the first outage and the first move that decrease the total cost

Moving the best of all outages shows the best performance despite the fact that it is not the most efficient in terms of time.

IV. Improved solution

This section explores and studies the ways of how to orchestrate and take full advantage of the procedures and reductions explained so far. Different set-ups are examined. Still our choices here are often driven by the wish to throw the problem to the solver and the different set of combinations of local improvements and their mutual interactions. This approach is feasible and natural having solved the difficult problem of production assignment successfully. We were thus able to create a spectrum of solutions and choose the best one, while the process proves to be also helpful for comparing the qualities and characteristics of different solutions and getting a better inner understanding of the problem. The quality of the final solutions and the efficiency of the method for B and X instances are strongly affected by their size. The idea to shrink the instances proves to be useful to tackle this problem to a certain degree. The full method for B and X instances is presented in the algorithm 4. Nevertheless, exactly the same method is used to solve the smaller instances A. The only difference is that these small instances are not shrinked in the pre-processing phase.

A. *Relaxed associated problems*

Here, we present how the original problem can be relaxed into an easy linear program with a huge number of variables for big instances as well as into one less constrained problem. The study of these relaxed formulations establishes a possible lower bound on the value of the objective function and verifies the quality of our subprocedures. Suppose first that we have a complete feasible solution. Fixing the outage schedule and fixing the refueling of each outage, then fixing production in possible decreasing profiles of every Type-2 power plant, the

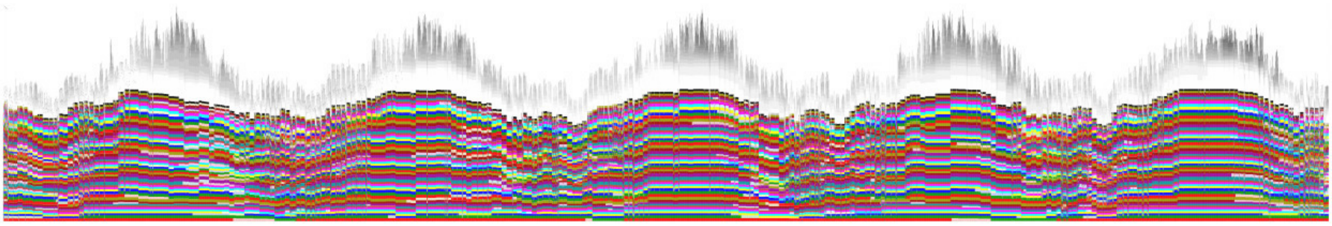

Fig. 2. The figure presents the structure of the best known result obtained on one of B instances. The rectangles of different colors represent the power demand satisfied by the Type-2 power plants. Above them come the grey rectangles representing the power satisfied by the Type-1 power plants. More expensive plants are presented with darker nuance of grey. The figure illustrates the complexity of one of the solutions to the problem.

Algorithm 4 Improving the solution

```
Shrink data set
Solve shrinked data set using algorithm described in section 3
Find feasible production plan for outages schedule from shrinked data set solution
Optimize solution by Local Search
while there is still time do
    Construct feasible outages schedule optimized with respect to the given criteria (actual marginal costs)
    Construct a feasible production plan
    Apply local improvements as long as the solution improves
end while
```

remaining problem could be easily modeled with a linear program that solves with optimality the production assignments for all power plants in all remaining time steps. This linear program for all instances A and for some of instances B and X is solvable on the testing machine in several minutes. The results confirm that local search based on marginal cost is the optimal procedure for instances A (see III-C0c). For other instances, the LP slightly improves the numerical results but could be very time and memory consuming. One way to cope with time complexity is to split the set of time steps into two subsets and solve the resulting programs. The combination of two solutions is not optimal for the whole problem but improves the existing result.

The classical approach to relax an optimization problem is to drop some constraints. The set of OCS constraints or its subset is a natural choice in this case, as some of them could be removed in reality due to the technological advances and because the structure of the relaxed solutions is similar to the associated complete solutions. One of the reasons is also that it is not difficult to drop them in the program. This relaxation also establishes the lower bound on the solutions of the presented method. The experimentation shows that these lower bounds are extremely good for instances A and make a bigger gap with actual solutions in instances B and X.

V. Evaluation

The whole solution is implemented in the C++ programming language on Linux x86-64 architecture and compiled with GCC version 4.4.3. The solver finally chosen for the constraint satisfaction part of the problem is IBM ILOG CPoptimizer version 12. The computer is equipped with an Intel i7 920 processor (2.66 GHz, 8M Cache, RAM 6 GB). All results reported in this paper are obtained on this architecture using the same software. The performance was not our main concern and we estimate that the speed of the proposed algorithm could be improved by 10-15 percent although the quality of the solution itself must not improve thereafter. Time limits imposed by ROADEF/EURO Challenge 2010 are respected (30 minutes for A instances and 60 minutes for B and X instances). All reported solutions passed the official solution checker provided by EDF. Almost all obtained solutions are in the gap within the range between 0.5 and 2 $\%$ of the best known solutions while on the instances A this gap is substantially smaller and is around 0.01%. For three instances from set A we obtain the best known result to our knowledge.

Note that our obtained solution quality for A instances is much better than for B and X instances. This originates from some nice properties of the A instances: smaller size, demand always bigger than Type-2 power plants production, comprise less spacing constraints and outages to schedule and thus the search spaces are smaller. Among the B and X instances we achieve good results with a robust deviation from the best known solution (see figure 1). It is important to mention here that only the B and X instances are considered for the final ranking of the ROADEF/EURO Challenge 2010.

VI. Conclusion

In this paper we propose one hybrid method to solve a large scale energy management problem. The methodology used to tackle the problem consists of,among other things, the constructive and greedy algorithms, local search procedures and constraint satisfaction techniques. The method first solves two interdependent subproblems in a consecutive manner and constructs one feasible solution. It then proceeds with improvements of the entire solution applying several local search techniques. The method is fully implemented in C++ and is fully operational and can be used to solve real world instances. Towards the end of the study, we obtained the best available results for all 15 instances and were challenged and motivated to improve our own. The numerical results are comparable with the best known, for some of them our method

TABLE II
THE TABLE SHOWS THE VALUES FOR OBJECTIVE FUNCTION FOR OUR BEST SOLUTIONS. WE COMPARE THE COSTS OF THE BEST KNOWN RESULTS TO OUR BEST RESULTS. IN THE SECOND COLUMN WE REPORT THE RESULTS OBTAINED WITHOUT OCS CONSTRAINTS. THIRD COLUMN PRESENTS THE VALUES FOR THE INITIAL FEASIBLE SOLUTIONS OBTAINED. LAST THREE COLUMNS SUMMARIZE OUR BEST RESULTS AND COMPARE THEM WITH BEST KNOWN RESULTS.

Best Solutions					
instance	no $CT14 - CT21$ constraints	First feasible sol	Solution	Best Known Solution	Score(percents)
dataA1	169 400 736 729	171 141 268 419	169 474 519 241	169 474 800 000	**-0.00016**
dataA2	145 817 899 228	147 270 198 259	145 956 733 339	145 956 800 000	**-0.000045**
dataA3	154 135 275 550	155 834 657 751	154 277 239 128	154 316 000 000	**-0.0251**
dataA4	111 370 665 296	113 828 236 687	111 505 728 462	111 494 000 000	0.010
dataA5	124 426 695 127	129 167 264 913	124 716 680 000	124 543 900 000	0.128
dataB6	832 052 548 93	840 268 154 84	837 632 963 07	834 247 162 17	0.405
dataB7	810 272 294 44	826 597 146 91	820 702 010 04	810 997 200 00	1.196
dataB8	825 562 154 92	842 156 632 28	837 866 683 28	818 997 400 00	2.301
dataB9	840 658 648 12	881 695 954 71	875 425 269 18	816 895 600 97	7.164
dataB10	794 324 086 12	798 625 487 13	794 669 685 74	777 670 249 99	2.185
dataX11	788 769 223 10	801 297 556 00	796 508 419 33	790 096 500 00	0.811
dataX12	776 845 130 30	786 515 578 88	782 742 078 67	775 639 900 00	0.915
dataX13	769 032 927 96	782 239 963 78	777 210 105 49	762 885 200 00	1.877
dataX14	774 748 125 50	783 373 185 33	780 275 365 71	761 494 800 00	2.466
dataX15	761 565 831 11	766 269 957 85	763 107 410 43	743 883 700 00	2.584

finds the solutions of practically the same quality and even improve several best results.

REFERENCES

[1] F. Brandt. (2010) Solving a Large-Scale Energy Management Problem with Varied Constraints, Diploma Thesis, Department of Informatics Institute of Theoretical Informatics, Karlsruhe Institut of Technology

[2] F. Gardi, K. Nouioua. (2011, April) Local Search for Mixed-Integer Nonlinear Optimization: a Methodology and an Application, EvoCop: 11th European conference on evolutionary computation in combinatorial optimisation, 27-29 April 2011 Torino, Italy

[3] S. Godskesen, T. S. Jensen, N. Kjeldsen, and R. Larsen. (2010) Solving a real-life large-scale energy management problem, CoRR, http://arxiv.org/abs/1012.4691

[4] M. Khemmoudj, M. Porcheron, H. Bennaceur. (2006) When constraint programming and local search solve the scheduling problem of Électricité de France nuclear power plant outages, In: Benhamou, F. (ed.) CP 2006, the 12th International Conference on Principles and Practice of Constraint Programming. Lecture Notes in Computer Science, vol. 4204, pp. 271-283. Springer, Berlin, Germany

[5] R. Apt. Krzysztof. (2003) Principles of constraint programming, 420. Cambridge university press

[6] (2009) IBM ILOG CPoptimizer and Cplex User's Guide

[7] M. Milano, P. Van Hentenryck. (2010) Hybrid Optimization, ISBN 978-1-4419-1643-3, Springer Optimization and Its Applications

[8] G. Pesant and M. Gendreau. (1999) A Constraint Programming Framework for Local Search Methods, Journal of Heuristics 5(3): 255-279

[9] M. Porcheron, A. Gorge, O. Juan, T. Simovic, and G. Dereu. (2009) Challenge roadef/euro 2010 : A large-scale energy management problem with varied constraints, http://challenge.roadef.org/2010/sujetEDFv22.pdf,

[10] J.-P. Watson and J. C. Beck. (2008) A hybrid constraint programming/local search approach to the job-shop scheduling problem, Proceedings of the 5th international conference on Integration of AI and OR techniques in constraint programming for combinatorial optimization problems, 263-277