Scheduling the Equipment Maintenance of an Electric Power Transmission Network Using Constraint Programming

Louis Popovic ✉
Computer Engineering and Software Engineering Department, Polytechnique Montréal, Canada

Alain Côté ✉
IREQ, Varennes, Canada

Mohamed Gaha ✉
IREQ, Varennes, Canada

Franklin Nguewouo ✉
Hydro-Québec, Canada

Quentin Cappart ✉
Computer Engineering and Software Engineering Department, Polytechnique Montréal, Canada

Abstract

Modern electrical power utilities must maintain their electrical equipment and replace it when the end of its useful life arrives. The Transmission Maintenance Scheduling (TMS) problem consists in generating an annual maintenance plan for electric power transportation equipment while maintaining the stability of the network and ensuring a continuous power flow for customers. Each year, a list of equipment (power lines, capacitors, transistors, etc.) that needs to be maintained or replaced is available and the goal is to generate an optimal maintenance plan. This paper proposes a constraint-based scheduling approach for solving the TMS problem. The model considers two types of constraints: (1) constraints that can be naturally formalized inside a constraint programming model, and (2) complex constraints that do not have a proper formalization from the field specialists. The latter cannot be integrated inside the model due to their complexity. Their satisfaction is thus verified by a black box tool, which is a simulator that mimics the impact of a maintenance schedule on the real power network. The simulator is based on complex differential power-flow equations. Experiments are carried out at five strategic points of Hydro-Québec power grid infrastructure, and involve more than 200 electrical equipment and 300 withdrawal requests. Results show that the model is able to comply with most of the formalized and unformalized constraints. It also generates maintenance schedules within an execution time of only a few minutes. The generated schedules are similar to the ones proposed by a field specialist and can be used to simulate maintenance programs for the next 10 years.

2012 ACM Subject Classification Applied computing → Operations research

Keywords and phrases Transmission maintenance scheduling, Electric power network, Constraint programming

Digital Object Identifier 10.4230/LIPIcs.CP.2022.34

1 Introduction

Modern electrical power utilities must maintain their electrical equipment and replace them as they reach the end of their useful life. Asset management is becoming strategically important for transmission utilities around the world. The International Electrotechnical Commission (IEC) has recognized that a specific power network approach to monitoring and managing assets is required [18]. To respond to these challenges, Hydro-Québec, a public utility company operating in Quebec, started the PRIAD-project, which aims to develop an integrated decision support system including predictive modeling methods for asset management [9]. This project includes different modules for cloud data warehouses, asset behavior, reliability, transmission system simulation, risk and optimization. Such initiatives to develop decision systems for identifying and prioritizing the replacement and maintenance of electrical equipment with an asset risk framework have been undertaken by many other utilities [3]. The main objective of every electric power system is to transport electricity from the generating units to the load centers in a secure manner. To do so, one of the main tasks of the *network control center* (NCC) is to use a contingency approach to ensure that maintenance activities do not lead to interrupted power supply [19]. To reduce the number of failures and improve power network reliability, assets are periodically removed from the grid network for preventive maintenance. Scheduling the preventive maintenance activities of electrical power utilities relates to two well-known problems: (1) the *generator maintenance scheduling* (GMS) problem, and (2) *the transmission maintenance scheduling* (TMS) problem. Solving both of these problems efficiently is crucial for network reliability and fluidity. However, they are also NP-hard due to the complexity of the constraints included. On the one hand, many approaches have been proposed for solving the GMS problem. For a recent review of these studies, the reader is referred to the survey proposed by Froger et al. [8].

On the other hand, the TMS problem has been less studied in the literature. The goal is to generate an annual maintenance plan for electric power transportation equipment while maintaining the stability of the network and ensuring a continuous power flow for customers. It is noteworthy to highlight that the TMS problem is limited to transmission equipment and does not include, for instance, distribution equipment. This is intended as most of the distribution equipment are not maintained and are used as run-to-fail, yielding a specific resolution process. Pandžić et al. [15] proposed a bi-objective mathematical model for solving a TMS problem involving only transmission lines. The idea is to compute an appropriate trade-off between ensuring transmission capacity and minimizing the maintenance impact on power system operation and then the market. To do so, they proposed to recast a non-linear formalization of the problem into a mixed-integer linear program, and to solve it using a standard branch-and-cut algorithm. In addition, Mei et al. [14] proposed another mixed-integer program that aims to maximize the maintenance willingness of transmission lines under security and capacity constraints of the transmission power system. To improve the computational efficiency, the authors proposed a machine learning approach accelerating the branching procedure of the solving algorithm. These two works applied the timetable obtained to IEEE 24-bus and 30-bus reliability test systems, which are relatively simple and not representative of networks involving complex electrical constraints. More recently, Rocha et al. [17] proposed a mixed-integer program for solving the TMS problem on a IEEE-24 system, similar to the one considered in this paper. Unlike the previous works, they consider a complete transmission grid and not only transmission lines. Solving is carried out by splitting the initial problem into two smaller optimization problems with the use of Benders decomposition [16]. However this approach does not consider advanced constraints related to the limitations of power transit inside the grid.

To the best of our knowledge, there are no related works that solve TMS problems on a complete transmission grid with various electrical equipment and transit-power constraints. Ensuring that transit-power limits are never violated is a critical concern in practice. This motivates our work to solve a TMS problem from the point of view of NCC operation. Each year, the NCC operator receives the annual maintenance plan with a suggested starting maintenance date and duration. The operator tries to satisfy the proposed maintenance plan, while satisfying electrical constraints known as *transit-power limits*. Such constraints guarantee power network stability during maintenance. In addition, a list of withdrawal rules that ensure stable power system operation is available. These rules, based on expert knowledge and power network analysis, represent restrictions on equipment that can or cannot be removed simultaneously from the grid. These constraints can be naturally formalized inside a mathematical model. However, transit-power constraints are trickier to handle. The theoretical values of the transit-power limits inside a power network subject to inactive equipment are based on complex constrained differential power-flow equations and are tedious to compute. For this reason, these constraints do not have a proper closed-form expression from the field specialists and cannot be easily integrated inside a mathematical model. In practice, the satisfaction of such constraints can be verified by a simulator that mimics the impact of a maintenance schedule on the real power network.

Based on this context, the goal of this paper is to find the optimal periods for removing specific transmission equipment from the grid for maintenance without impeding energy delivery. By doing so, we aim to provide planners with insight in order to help them in their decisions, which are currently done manually based on their field expertise. The complexity of the maintenance task will be reduced and they will be able to dedicate a specific focus on the most challenging aspects of the task. The specific contributions of this paper are as follows: (1) an approach based on constraint programming (CP) for solving a TMS problem on a transmission grid with transit-power constraints; (2) the use of a black-box simulator approximating the electrical impact for each proposed maintenance schedule in order to validate the satisfaction of transit-power constraints; (3) a two-step objective function dedicated to maximizing the balance of the schedule and to maximizing the overlap of withdrawal requests involving the same equipment; and (4) experiments on five strategic points of a real power grid infrastructure that involves more than 200 pieces of electrical equipment and 300 withdrawal requests. Results show that the model is able to comply with most of the power-transit constraints. The maintenance schedules are generated within an execution time of few minutes and are similar to the ones proposed by field specialists. The next section formalizes the TMS problem and introduces the constraint programming model that we have designed. The solving process is then described in Section 3. Lastly, experiments and results are presented in Section 4.

2 Modelling the Transmission Maintenance Scheduling Problem

The goal is to generate an annual maintenance plan of withdrawal requests for electric power transportation equipment (power lines, capacitors, transistors, etc.) while maintaining the stability of the network. Let W be the set of withdrawal requests that must be scheduled inside the planning horizon, and let E be the set of electrical equipment involved in the network grid. Each withdrawal request $w_i \in W$ has a duration $l_i \in \mathbb{N}$, a list of equipment $E_i \in 2^E$ to withdraw, and a size $n_i \in \mathbb{N}$, corresponding to the number of equipment pieces related to the request ($n_i = |E_i|$). Each equipment $e_j \in E$ can be associated to one or several withdrawal requests, indicating that the equipment e_j must be withdrew when the request is fulfilled. We use the notation $e^i_k \in E_i$ to refer to the k-th equipment associated to the withdrawal request w_i. Similarly, $id(e^i_k) \in E$ and $ch(e^i_k) \in \mathbb{R}$ refer to the equipment identifier of the k-th equipment associated to the withdrawal request w_i, and the corresponding electrical charge $[Mvar]$[1], respectively. Finally, the planning horizon is defined as the days

[1] Megavolt-ampere of reactive power; an AC electrical measurement unit.

between d_0 and d_m. The annual period during which withdrawals are permitted is limited from March 15th to November 15th. This gives 245 consecutive days ($m = 245$). All withdrawal requests must be started and finished within this horizon. No maintenance is allowed outside this period. The reason is that the peak of electrical power consumption in Quebec happens during winter. The parameters introduced are summarized in Table 1.

Table 1 List of parameters used in the constraint programming model.

Entity	Parameter	Description
Request (W)	l_i	Duration (in days) of the withdrawal request w_i
	n_i	Number of equipment to withdraw for the request w_i
	E_i	Set of equipment to withdraw for the request w_i
Equipment (E)	$id(e^i_k)$	Identifier of the k-th equipment of request w_i
	$ch(e^i_k)$	Electrical charge $[MVar]$ of the k-th equipment of request w_i
Horizon	d_0	First day allowed for the maintenance
	d_{245}	Last day allowed for the maintenance

2.1 Decision Variables

We model this problem as a constraint-based scheduling model with time-interval variables, also referred to as *activities*, using the formalism proposed by Laborie et al. [11, 13]. Each withdrawal request $w_i \in W$ is modelled as an activity and is composed of four variables: a start time $s(w_i)$, a duration $d(w_i)$, a completion time $c(w_i)$, and a binary execution status $x(w_i)$. In our case, the duration of each request is known ($d(w_i) = l_i$), and all the requests must be executed, yielding $x(w_i) = 1$ for all $w_i \in W$. Each piece of equipment $e^i_k \in E_i$ related to a withdrawal request w_i is also associated to an activity. Then, a situation involving 10 requests and 20 pieces of equipment will generate at most 200 activities, as a specific piece of equipment can be involved in several withdrawal requests. As a simple synchronization constraint, an equipment item must be withdrawn during the same period as its request. The domain of all the activities are presented below. A visualization of the decision variables and the temporal relations is proposed in Figure 1.

$$\forall w_i \in W : \begin{cases} s(w_i) \in [d_0, d_{245} - l_i] \\ d(w_i) = l_i \\ c(w_i) = s(w_i) + d(w_i) \\ x(w_i) = 1 \end{cases} \qquad \forall e^i_k \in E_i : \begin{cases} s(e^i_k) = s(w_i) \\ d(e^i_k) = d(w_i) \\ c(e^i_k) = c(w_i) \\ x(e^i_k) = x(w_i) \end{cases} \tag{1}$$

2.2 Constraints

The model leverages the *cumul function* introduced in constraint-based scheduling by Laborie et al. [13]. Briefly, such a function is used to represent the accumulated consumption of a resource by activities over a timing horizon. When a new activity is started, the consumption of the resource increases. Similarly, the consumption goes down when the activity is completed. This behaviour is related to the *cumulative* global constraint [2, 10]. Besides, our model is based on the *alwaysIn* and *noOverlap* constraints. Following the formalization of Laborie et al. [13], they are defined as follows:

- `alwaysIn`$(f, u, v, h_{min}, h_{max})$ ensures that the accumulated consumption of the cumul function f remains between h_{min} and h_{max} inside the interval $[u, v)$.
- `noOverlap`(A) ensures that the activities $a \in A$ do no overlap in time.

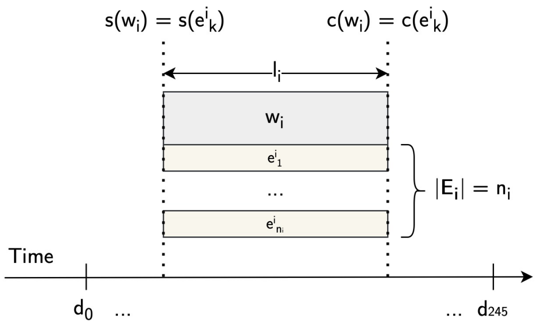


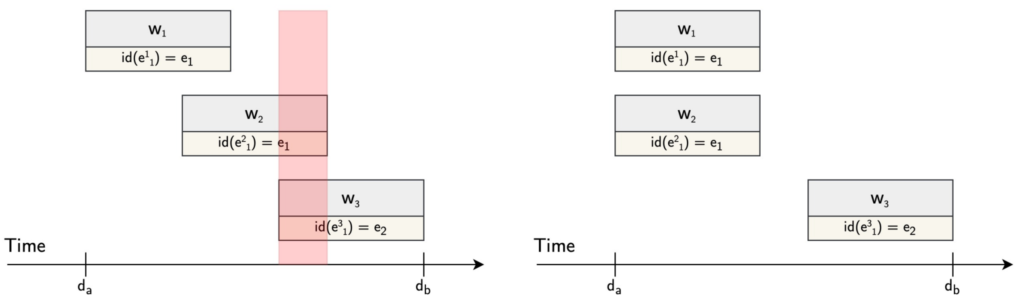
Figure 1 Illustration of the decision variables considered in the model.

Four constraints are involved in our model, two of them are based on *alwaysIn* constraint and the other twos are based on *noOverlap* constraint. The remaining of this section is dedicated to describe them.

Constraint 1: Limitation on Simultaneous Equipment Withdrawals. Let $S \in 2^E$ be an arbitrary set of equipment. This constraint states that a maximum of h pieces of equipment from the set S can be withdrawn together between the days d_a and d_b. We introduce a cumul function $f_1 : [d_0, d_{245}] \times S \to \mathbb{N}$ indicating the number of equipment items from S that are currently withdrawn for each time step of the planning horizon. The restriction is then modelled using the *alwaysIn* constraint [1] as follows.

$$\texttt{alwaysIn}(f_1, d_a, d_b, 0, h) \tag{2}$$

It ensures that the number of equipment items withdrew from S (returned by f_1) is always included between the range $[0, h]$ during the time interval $[d_a, d_b)$. A valid and an invalid solution for the configuration $S = \{e_1, e_2\}$ and $h = 1$ is illustrated in Figures 2a and 2b.



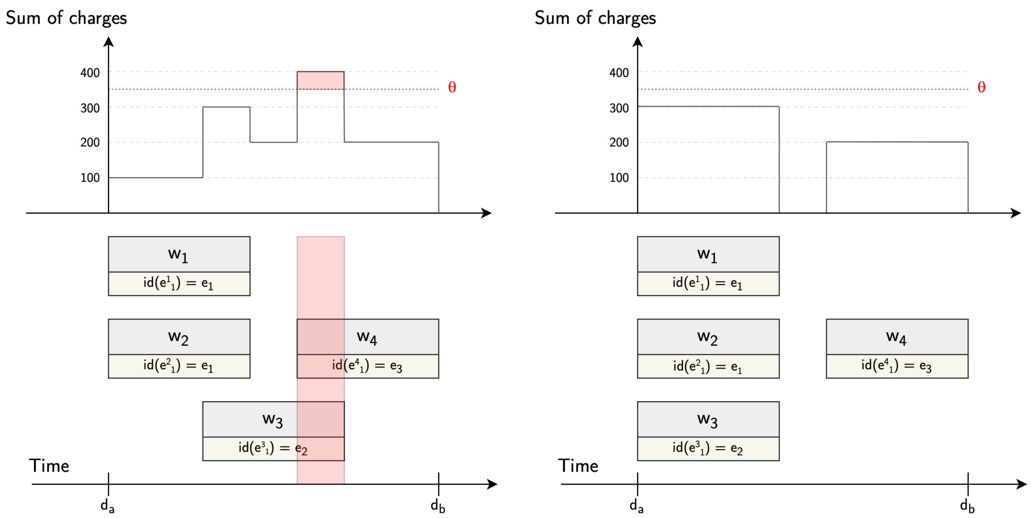
(a) Example of an invalid solution. **(b)** Example of a valid solution.

Figure 2 Illustration of Constraint 1 (limitation on simultaneous equipment withdrawal).

Constraint 2: Limitation on the Electrical Charge during Withdrawals. Let $S \in 2^E$ be an arbitrary set of equipment and θ a threshold of an electrical charge. This constraint states that the sum of the charges of the withdrawn equipment from S must always be below θ. We introduce a cumul function $f_2 : [d_0, d_{245}] \times S \to \mathbb{N}$ indicating the accumulated charge of the equipment, i.e. $\sum_{e \in S} ch(e)$, that is currently withdrawn for each time step of the planning horizon. The constraint is modelled as follows.

$$\texttt{alwaysIn}(f_2, d_0, d_{245}, 0, \theta) \tag{3}$$

It ensures that the electrical charge returned by f_2 is always included between the range $[0, \theta]$ during the complete planning horizon $[d_0, d_{245})$. A valid and an invalid solution for the configuration $S = \{e_1, e_2, e_3\}$, $\theta = 350$, $ch(e_1) = 100$, $ch(e_2) = 200$, and $ch(e_3) = 200$ is illustrated in Figures 3a and 3b.



(a) Example of an invalid solution. **(b)** Example of a valid solution.

Figure 3 Illustration of Constraint 2 (limitation on the electrical charge during withdrawals).

Constraint 3: No Overlap on Equipment Withdrawals. Let $\Lambda = \{S_1, S_2, \ldots, S_K\}$ be a set containing K sets of equipment $S_k \in 2^E$. This constraint states that equipment coming from different sets of Λ cannot be withdrawn together. Only equipment included in the same set or that are identical (same identifier) can be withdrawn together. The *noOverlap* constraint [1] is used for this purpose.

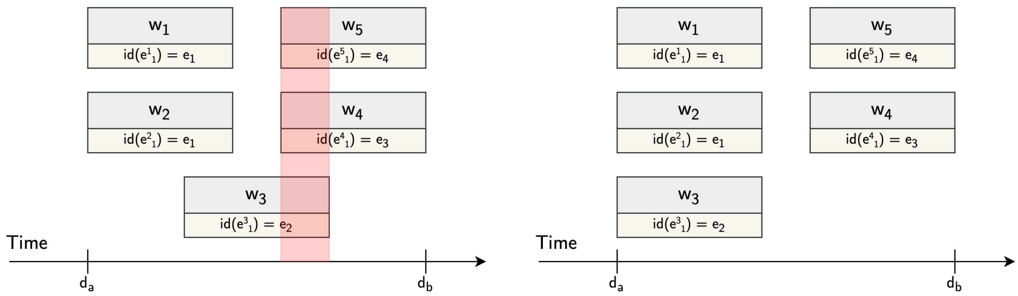
$$\texttt{noOverlap}\Big(\{e_i, e_j\}\Big) \;\; \forall e_i \in S_i \wedge \forall e_j \in S_j \wedge \forall S_i \in \Lambda \wedge \forall S_j \in \Lambda \wedge S_i \neq S_j \wedge id(e_i) \neq id(e_j) \quad (4)$$

This constraint ensures that for each pair of different equipment belonging to different sets, one of them must have its activity ended before the other one starts. A valid and an invalid solution for the configuration $\Lambda = \big\{\{e_1, e_2\}, \{e_3, e_4\}\big\}$ is illustrated in Figures 4a and 4b.

Constraint 4: No Overlap inside a Set of Equipment. Let $S \in 2^E$ be an arbitrary set of equipment. This constraints states that equipment from this set cannot be removed together and is modelled as follows.

$$\texttt{noOverlap}\Big(\{w_i \mid \forall w_i \in S\}\Big) \quad (5)$$

It ensures that for each pair of equipment in set S, one of them must finish before the other one starts.



(a) Example of an invalid solution. **(b)** Example of a valid solution.

Figure 4 Illustration of Constraint 3 (no overlap on equipment withdrawals).

2.3 Objective Functions

A solution is currently feasible if all the withdrawal requests have been successfully scheduled while ensuring the satisfaction of the four constraints presented in the previous section. However, the transit-power constraints are not yet taken into account and can break the feasibility of a solution. The challenge is that these constraints do not have a closed-form expression and cannot be integrated inside the model. That being said, as a heuristic rule from field specialists, a solution is more likely to satisfy the transit-power constraints when (1) the withdrawal activities are properly balanced inside the planning horizon, and (2) when the activities related to a same equipment are scheduled together. We propose to integrate these rules inside the model through two objective functions having a lexicographic importance.

Objective 1: Maximizing the Schedule Balance. The goal is to balance the withdrawal requests inside the planning horizon. This is related to the *balance* constraint introduced by Bessiere et al. [4]. Generally speaking, the planning horizon has a length of $d_m - d_0$ days. We split this interval into r sub-periods of p days, i.e. $r = \lceil \frac{d_m - d_0}{p} \rceil$. In our case, $m = 245$ and following the recommendations of field specialists, a value of 50 days (p) has been selected, yielding 5 sub-periods (r). This value has been fixed empirically and validated by planners. It is possible that a request overlaps over several sub-periods. For instance, a request between day 0 and day 60 will be counted in the sub-periods $[0, 50]$ and $[51, 100]$. Let R be the set of sub-periods and Ω be a list storing, for each sub-period $r \in R$, the number of requests withdrawn during the sub-period r. In practice, Ω is computed using the well-known function *count*, which is dedicated to counting the number of variables in a list that has a given value [7]. The objective function is then as follows. It drives the solving process to find a schedule that minimizes the largest difference between the number of activities scheduled across the sub-periods.

$$\texttt{minimize} \left| \max_{r \in R} \left(\Omega_r\right) - \min_{r \in R} \left(\Omega_r\right) \right| \tag{6}$$

Objective 2: Maximizing Same Equipment Withdrawal Overlaps. The goal is to maximize the number of overlapping withdrawals of the same equipment. The rationale is that when the same equipment is involved in different withdrawal requests, it is preferable to withdrawn them simultaneously. Let $D = E_1 \cup E_2 \cup ... \cup E_n$ be a set containing the sets of equipment from all the withdrawal requests. The objective function is defined as follows. It is based on the *overlapLength* function that computes the number of overlapping days between two activities [12]. All the overlaps are then summed up and maximized.

$$\texttt{maximize}\left(\sum_{e_1\in D}\sum_{e_2\in D}\left\{\texttt{overlapLength}(e_1,e_2)\;\middle|\; e_1\neq e_2 \wedge id(e_1)=id(e_2)\right\}\right) \quad (7)$$

It is noteworthy to highlight that the standard objective of minimizing the *makespan* is not considered. In our application, there are no benefits to finishing the maintenance as soon as possible. Spreading the maintenance schedule in the complete horizon is much more desirable as it allows a better flexibility for the field operators. For instance, it allows the planner to readjust dynamically the maintenance schedule when unpredictable events happen, such as an equipment outage.

3 Solving the Transmission Maintenance Scheduling Problem

So far, the transit-power constraints have not been taken into account. Although such constraints cannot be integrated inside the model, their satisfaction can be easily checked thanks to a simulator that mimics the impact of a maintenance schedule on the real power network. The simulator is based on complex differential power-flow equations and has been developed internally by Hydro-Québec. It simulates power flow thanks to PSS/E software.

We propose to leverage this simulator, as a black-box tool, in order to verify the satisfiability of a schedule. Let γ be the power grid considered, let s be a maintenance schedule obtained as a solution for γ, and let d a specific day on the planning horizon. The simulator consists of two black-box functions: (1) $\psi_1(\gamma, s, d) \to \mathbb{R}$ which computes the transit-power generated by the solution for a specific day, and (2) $\psi_2(\gamma, s, d) \to \mathbb{R}$, which computes a lower bound on the transit-power that must be satisfied for the obtained schedule, also for a specific day. A solution s on the power grid γ is feasible if it is always above the threshold during the planning horizon.

$$\psi_1(\gamma, s, d) \geq \psi_2(\gamma, s, d) \qquad \forall d \in [d_0, d_{245}] \quad (8)$$

The idea of the solving process is to generate diverse solutions using the constraint programming model and to filter them using the simulator. Solutions that are compliant with the simulator are feasible and can be used in practice. This process is illustrated in Figure 5.

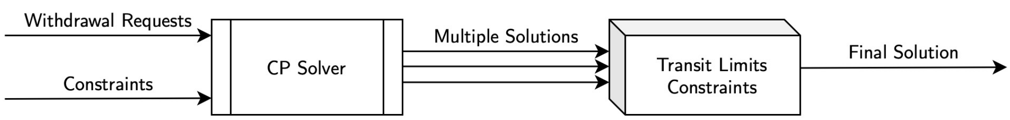
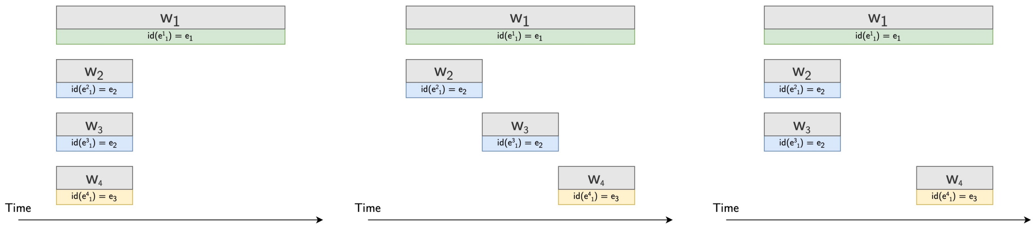


Figure 5 Illustration of the solving pipeline.

One challenge is to generate solutions that are diverse in order to maximize the chance of having at least one schedule accepted by the simulator. We resort to three mechanisms to ensure the diversity of solutions: (1) integrating domain knowledge as objective function, (2) adding constraints dynamically when a solution has been found, and (3) directing the search by a *multi-point* strategy. This section presents how these ideas are integrated into the solving process. We assume that a solution satisfying all the formalized constraints is obtained.

Mechanism 1: Injecting Domain Knowledge as Objective Function. Field specialists have heuristic rules for creating scheduling satisfying the transit-power constraint. Those have been formalized in Equations (6) and (7). The idea is to integrate such rules as a two-steps objective function. The problem is first solved by maximizing the balance of the schedule. From the solution obtained, a second solving process is executed in order to maximize the withdrawal overlaps of the same equipment. The value of the first objective is allowed to decrease up to a given threshold ϵ. This process is illustrated in Figure 6.



(a) Initial solution. **(b)** Adding Equation (6). **(c)** Adding Equation (7).

Figure 6 Illustration of the two-steps objective function.

Mechanism 2: Adding new Constraints Dynamically. Two solutions successively generated by a CP solving process are likely to share many characteristics. We improve the diversity of the solutions obtained by adding a new constraint each time a new feasible solution has been found. Let $w^\star$ refer to the value of the variable w in the last solution found. The new constraint ensures that the next generated solution must have at least l withdrawal requests moved from at least d days of the previous solution. The start time is used as reference. We empirically set $d = 1$ and $l = 1$, which already yielded diverse enough solutions.

$$\sum_{w \in W} \left(\left| s(w) - s(w^\star) \right| \geq d \right) \geq l \tag{9}$$

Mechanism 3: Directing the Search by a Multi-Point Strategy. Finally, a *multi-point* strategy with the default search heuristics proposed by CP Optimizer is used for driving the solving process [13]. This strategy creates an initial set of solutions and combines them together in order to produce improved solutions. It has the benefit of providing a more diversified solution than a standard depth-first search. However, it acts as an incomplete search procedure and cannot prove the optimality of a solution. That being said, this limitation is not restrictive in our case, as we only need to find feasible solutions. The objective functions are only used as heuristics.

4 Experimental results

The goal of the experiments is to show the adequacy of the approach to generate schedules that can be used in practice for the geographic area considered. To do so, the maintenance planning designed by field specialists for the year 2020 is considered and compared with the planning obtained by our approach. In total, 359 withdrawal requests were considered and 271 electrical equipment items are involved. Each withdrawal request involves at most 8 items, yielding a maximum of 2872 activities. The maintenance schedule has an impact on five strategic points of the power grid infrastructure, also referred to as *interface*. Each interface has its own transit-power constraints. It is interesting to mention that due to Québec geography, the network is not intensively meshed. The production infrastructures are all located in the North while most of the consumption is made in the South. The experiments are executed on an Intel i5-8520 processor (1.6 GHz) with 16 GB of RAM. The model is implemented in C++ using CP Optimizer 20.1. In total, the solving process took 1000 seconds. Roughly 85% of the execution time was dedicated to finding solutions and 15% was used to verify the transit-power constraints with the simulator.

4.1 Visualization of a Feasible Schedule

A visualization of the maintenance schedule obtained for the first interface is proposed in Figure 7. The x-axis represents the planning horizon from day 0 to day 245. As commonly done for scheduling problems, each gray bar represents the execution of the withdrawal request associated with each equipment item. For practical reasons, only equipment that affects the transit-power limit is displayed. For reasons of confidentiality, equipment names are omitted. For instance, they can correspond to power lines, capacitors, transistors, etc. The red curve indicates the transit-power generated by the maintenance schedule (output of ψ_1 function) while the blue curve indicates the transit-power limit (output of ψ_2 function). Consequently, a schedule satisfies the transit-power constraints if and only if the red curve is always above the blue curve, which is the case for this interface. A similar result for three other interfaces is presented in Appendix A. These results demonstrate that our approach is sufficient to satisfy the transit-power constraints on these interfaces.

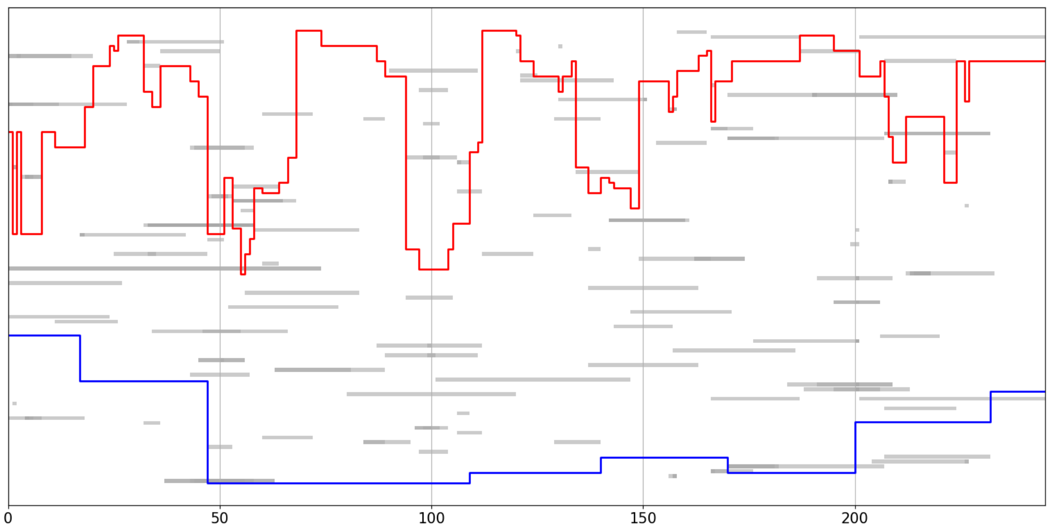


Figure 7 Visualization of a feasible maintenance schedule (first interface).

4.2 Visualization of an Unfeasible Schedule

Among the five interfaces considered in our power grid, four of them satisfy the transit-power constraints. A visualization of the maintenance schedule obtained for the last one is proposed in Figure 8. Interestingly, the transit-power constraints are violated only a few times (e.g., around day 50 and after day 200). Generally speaking, we also observe that the safety margin between the two curves is tinier than the one presented in Figure 8. This case was discussed with field specialists. They confirmed that ensuring the satisfaction of transit-power constraints at this interface is challenging. In practice, they regularly have to accommodate with a schedule that does not respect the constraints at this interface. Addressing this challenging interface is part of future work.

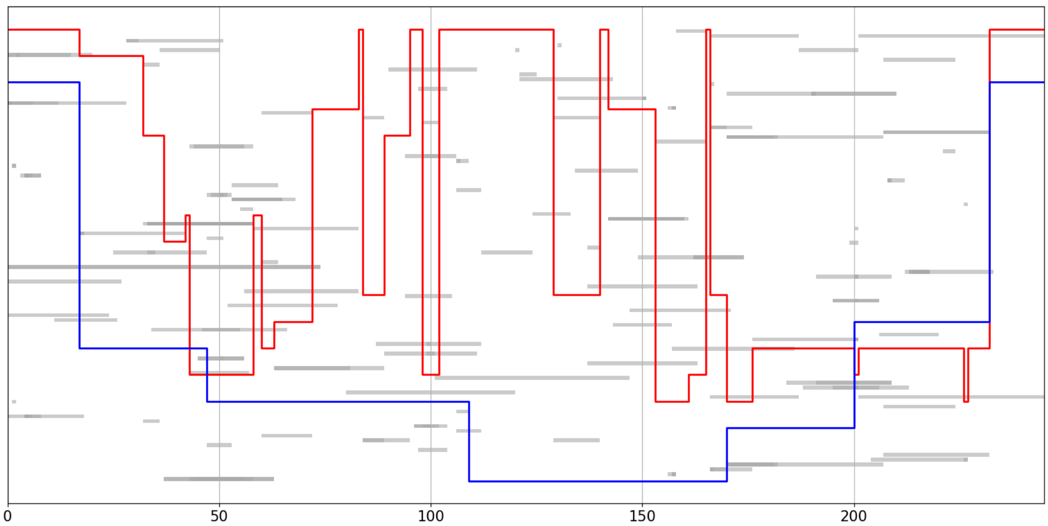


Figure 8 Visualization of an unfeasible maintenance schedule (fifth interface).

4.3 Evaluating the Similarity with the Historical Schedule

The goal of this analysis is to highlight the similarities and the differences between the solutions allowed by our constraints and the one designed by field specialists that was used in 2020. Then, we will be able to assess if the decisions made are consistent with the ones historically done, and otherwise discover potential sources of discrepancies. We propose a visualization of this information using a confusion matrix. Each request can be either *accepted* or *refused* by the field operator. By replaying the decision of the operator on historical requests of 2020, Table 2 shows the proportion of withdrawal requests that have the same or different status with our constraints and the historical model.

Table 2 Proportion of accepted or refused requests between both schedules in 2020.

		Schedule allowed by the constraints	
		Approved Requests	*Refused requests*
Historical schedule	*Approved requests*	**61.5%**	12.5%
	Refused requests	17%	**9%**

Interestingly, we notice that 70.5% (61.5% + 9%) of the requests have the same status. It means that the decision regarding these requests is identical as what has been done in 2020. In addition, 12.5% of the requests have been refused by our model while being accepted in 2020. This corresponds to situations where the field operator has accepted a request that will cause a constraint violation. This has been done either intentionally (e.g., constraint assessed to be too restrictive) or unintentionally given the complexity of this task. Finally, 17% of the requests have been approved by our model but were refused in 2020. This may be an indication that some constraints used in practice by field specialists are missing in the model. These can be either other technical constraints or non-related constraints such as budget or workforce constraints.

4.4 Evaluating the Schedule Balance

This experiment assesses the schedule balance obtained with our model. To do so, we count the number of requests per period of 31 days (one month) and analyze if the schedule provided by our model has a similar balance as the one designed in 2020. This is summarized in Table 3 for each period of 31 days since March 15. The spread value indicates the difference between the maximum and the minimum values inside the planning horizon. The lower the value is, the more balanced is the schedule. We can observe that our generated schedule is slightly more balanced.

Table 3 Comparison of the maintenance schedule maintenance.

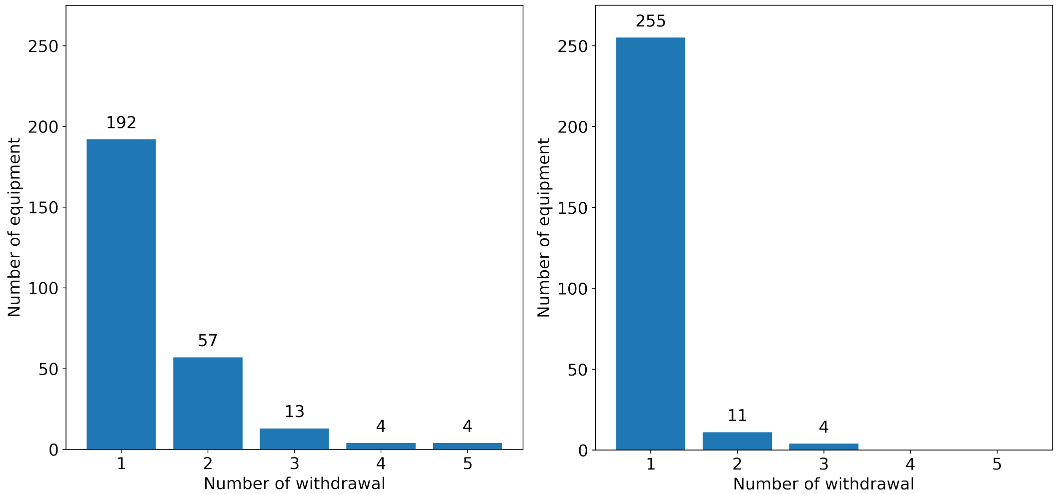
	Planning horizon (split into 8 months)								Spread value
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	
Historical schedule	45	57	72	55	36	42	62	21	51
Our schedule	58	61	32	56	63	40	63	19	42

4.5 Evaluating the Overlaps between Equipment Withdrawals

Most of the time, the same equipment unit is involved in different withdrawal requests. The second objective function is dedicated to maximizing the overlaps between these requests. To evaluate this objective, we count the number of times an equipment has been withdrawn from the network. Figure 9 shows the distribution of the number of withdrawals required per equipment for the historical schedule (left) and for our model (right). We can observe that the model is able to remove each equipment less often which is what is intended by this objective function.

5 Conclusion and perspective

Modern electrical power utilities must maintain their electrical equipment and replace them as they reach the end of their useful life. Generating an annual maintenance plan for the electric power transportation equipment while maintaining the stability and efficiency of the network is still a challenge at present. Based on this context, we proposed a constraint programming approach for solving a realistic transportation maintenance scheduling problem. The focus was to design an approach that could handle two types of constraints: (1) constraints that could be naturally formalized inside a constraint programming model, and (2) constraints that were too complex to be implemented but could be verified using a black-box tool. The objective was to generate schedules similar to what is currently being done by field specialists in order to simulate maintenance programs for the next 10 years. Experimental results show that the model captures most of the unformalized constraints and is able to generate realistic schedules. It is important to highlight that two kinds of constraints are not yet considered: budget constraints, and specialized crew availability constraints. In future work, we shall attempt to integrate such constraints into the model. Another interesting aspect is the assessment of increased risk of failures during maintenance. For such a criticality analysis,



(a) Historical schedule.

(b) Our schedule.

Figure 9 Comparison of the number of overlaps between equipment withdrawal.

other modeling and solving tools (e.g., stochastic programming [6]) may be considered. Finally, another idea is to leverage methods from *constraint acquisition* in order to learn new constraints from the interaction with the black-box simulator [5].

References

1 ILOG CPLEX – Cumul functions in CP Optimizer. `https://www.ibm.com/docs/en/icos/20.1.0?topic=concepts-cumul-functions-in-cp-optimizer`. Accessed: 2022-02-06.

2 Abderrahmane Aggoun and Nicolas Beldiceanu. Extending CHIP in order to solve complex scheduling and placement problems. *Mathematical and computer modelling*, 17(7):57–73, 1993.

3 David L Alvarez, Diego F Rodriguez, Alben Cardenas, F Faria da Silva, Claus Leth Bak, Rodolfo García, and Sergio Rivera. Optimal decision making in electrical systems using an asset risk management framework. *Energies*, 14(16):4987, 2021.

4 Christian Bessiere, Emmanuel Hebrard, George Katsirelos, Zeynep Kiziltan, Émilie Picard-Cantin, Claude-Guy Quimper, and Toby Walsh. The balance constraint family. In *International Conference on Principles and Practice of Constraint Programming*, pages 174–189. Springer, 2014.

5 Christian Bessiere, Frédéric Koriche, Nadjib Lazaar, and Barry O'Sullivan. Constraint acquisition. *Artificial Intelligence*, 244:315–342, 2017.

6 John R Birge and Francois Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

7 Mats Carlsson, Johan Widen, Johan Andersson, Stefan Andersson, Kent Boortz, Hans Nilsson, and Thomas Sjöland. *SICStus Prolog user's manual*, volume 3. Swedish Institute of Computer Science Kista, Sweden, 1988.

8 Aurélien Froger, Michel Gendreau, Jorge E. Mendoza, Eric Pinson, and Louis-Martin Rousseau. Maintenance scheduling in the electricity industry: a literature review. *European Journal of Operational Research*, 251(3):695–706, June 2016. `doi:10.1016/j.ejor.2015.08.045`.

9 Mohamed Gaha, Bilal Chabane, Dragan Komljenovic, Alain Côté, Claude Hébert, Olivier Blancke, Atieh Delavari, and Georges Abdul-Nour. Global methodology for electrical utilities maintenance assessment based on risk-informed decision making. *Sustainability*, 13(16), 2021. `doi:10.3390/su13169091`.

10 Steven Gay, Renaud Hartert, and Pierre Schaus. Simple and scalable time-table filtering for the cumulative constraint. In *International conference on principles and practice of constraint programming*, pages 149–157. Springer, 2015.

11 Philippe Laborie and Jerome Rogerie. Reasoning with conditional time-intervals. In *FLAIRS conference*, pages 555–560, 2008.

12 Philippe Laborie and Jérôme Rogerie. Temporal linear relaxation in IBM ILOG CP Optimizer. *Journal of Scheduling*, 19(4):391–400, 2016.

13 Philippe Laborie, Jérôme Rogerie, Paul Shaw, and Petr Vilím. IBM ILOG CP Optimizer for scheduling. *Constraints*, 23(2):210–250, 2018.

14 Jingcheng Mei, Guojiang Zhang, Donglian Qi, and Jianliang Zhang. Accelerated solution of the transmission maintenance schedule problem: A bayesian optimization approach. *Global Energy Interconnection*, 4(5):493–500, 2021. `doi:10.1016/j.gloei.2021.11.001`.

15 Hrvoje Pandzic, Antonio J. Conejo, Igor Kuzle, and Eduardo Caro. Yearly maintenance scheduling of transmission lines within a market environment. *IEEE Transactions on Power Systems*, 27(1):407–415, 2012. `doi:10.1109/TPWRS.2011.2159743`.

16 Ragheb Rahmaniani, Teodor Gabriel Crainic, Michel Gendreau, and Walter Rei. The benders decomposition algorithm: A literature review. *European Journal of Operational Research*, 259(3):801–817, 2017.

17 Mariana Rocha, Miguel F. Anjos, and Michel Gendreau. Optimal planning of preventive maintenance tasks on electric power transmission systems. In *31st European Safety and Reliability Conference (ESREL)*, pages 1025–1025, January 2021. `doi:10.3850/978-981-18-2016-8_721-cd`.

18 Hiroki Shigetsugu, Stewart Whyte, Paul Penserini, Boudewijn Neijens, David Neilson, and Jos Wetzer. Standardization activities of management of network assets in power systems in IEC, April 2022.

19 Gilles Trudel, Jean-Pierre Gingras, and Jean-Robert Pierre. Designing a reliable power system: Hydro-quebec's integrated approach. *Proceedings of the IEEE*, 93(5):907–917, 2005.

A Appendix: Solutions at Other Interfaces

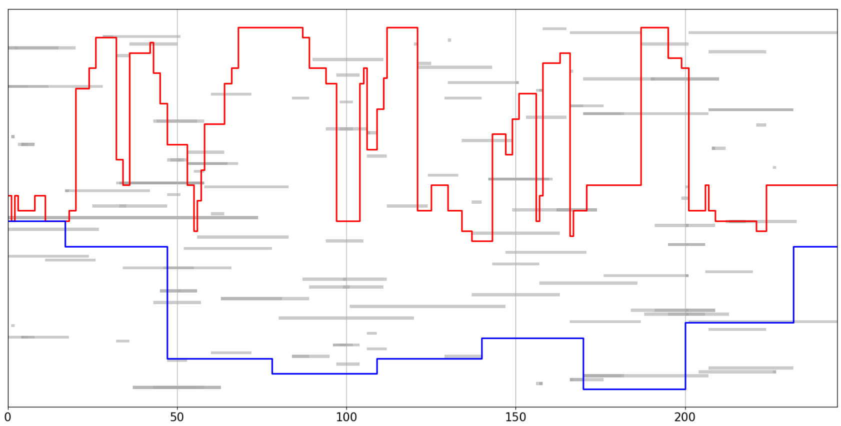


Figure 10 Visualization of a feasible maintenance schedule (second interface).

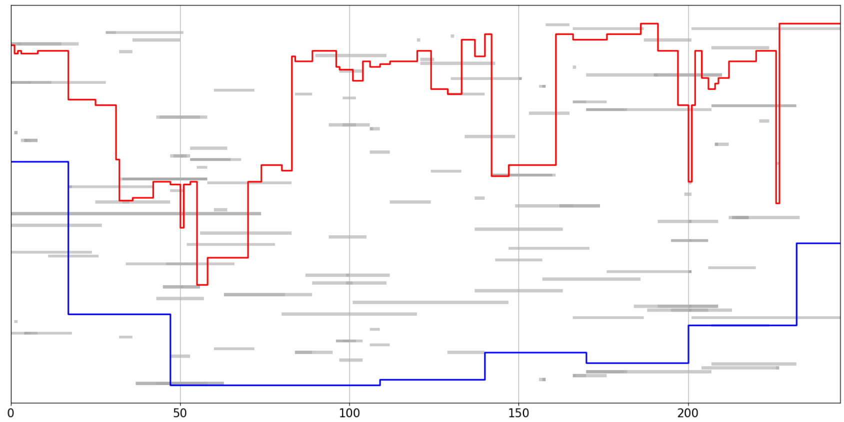


Figure 11 Visualization of a feasible maintenance schedule (third interface).

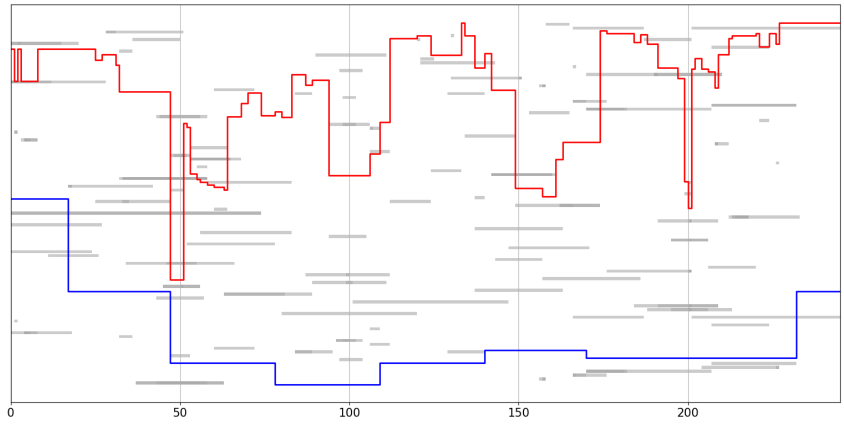


Figure 12 Visualization of a feasible maintenance schedule (fourth interface).