An efficient decomposition matheuristic for the transient stability constrained unit commitment at Hydro-Quebec

E. M. Er Raqabi, A. Bani, M. Morabit, A. Blondin Massé, A. Besner, J. Fournier, I. El Hallaoui

An efficient decomposition matheuristic for the transient stability constrained unit commitment at Hydro-Quebec

El Mehdi Er Raqabi [a]

Abderrahman Bani [a, b]

Mouad Morabit [b]

Alexandre Blondin Massé [b]

Alexandre Besner [a, b]

Julien Fournier [c]

Issmaïl El Hallaoui [a]

[a] *Mathematics and Industrial Engineering Department, Polytechnique Montréal & GERAD, Montréal, (Qc), Canada, H3T 1J4*

[b] *Hydro-Québec Research Institute (IREQ), Varennes, (Qc), Canada J3X 1S1*

[c] *Hydro-Quebec Operation Department, Montreal, (Qc), Canada H2Z 1A4*

el-mehdi.er-raqabi@polymtl.ca
bani.abderrahman@hydroquebec.com
Morabit.mouad@hydroquebec.com
blondinmasse.alexandre@hydroquebec.com
besner.alexandre@hydroquebec.com
fournier.julien@hydroquebec.com
issmail.elhallaoui@polymtl.ca

Abstract : This paper tackles a complex variant of the unit commitment (UC) problem at Hydro-Quebec, referred to as the transient stability constrained unit commitment (TSCUC) problem. First, the complete problem is described as a mixed-integer linear program (MILP). Next, an investigation strategy is conducted to identify complexity sources. Then a matheuristic is proposed, taking advantage of the temporal dimension of the problem. Finally, the matheuristic is enhanced by tuning the solver's configuration and by relying on linearization techniques. The benefits of the matheuristic are highlighted by using real-life instances from Hydro-Quebec.

Keywords: MILP, large-scale optimization, matheuristic, unit commitment

Notation

	Sets		$\phi_{g,k}^{y,d}$	Flow turbined by generator g at yield y and drop height d when configuration k is active
$c \in \mathcal{C}$	Spillways		δ_p	Duration of time step p
$d \in \mathcal{D}$	Drop heights (low, hi)		$\rho_{n,p}^{p'}$	Proportion of volume whose transit between time steps p and p' in river n is completed
$g \in \mathcal{G}$	Hydraulic generators		$\underline{\xi}^{sfc}$	Low power range threshold for secondary frequency control (SFC)
$h \in \mathcal{H}$	Hydro plants			
$l \in \mathcal{L}$	Links		$\bar{\xi}^{sfc}$	High power range threshold for SFC
$m \in \mathcal{M}$	Stability zones		$\bar{\eta}^{sfc}$	Total power range threshold for SFC
$n \in \mathcal{N}$	Rivers		ξ_m^{abs}	Minimum stability reserve threshold to maintain in stability zone m
$r \in \mathcal{R}$	Reservoirs			
$e \in \mathcal{E}$	Topology constraints		ξ_m^{rate}	Rate of stability power to maintain in stability zone m
$p \in \mathcal{P}$	Time steps			
$p \in \mathcal{P}'$	Time steps, excluding the initial time step p_0		ν_r	Initial volume of reservoir r
$y \in \mathcal{Y}$	Normal generator yields (min, opt, max)		λ_r^d	Level of reservoir r at drop height d
$y \in \mathcal{Y}'$	Generator yields (min, opt, max, $stab$)			*Variables*
$z \in \mathcal{Z}$	Zones		$\mathbf{x}_k^p$	Indicator if configuration k is active during time step p
	Relations		$\mathbf{x}_g^p$	Indicator if generator g is active during time step p
G_h	Generators of hydro plant h		$\mathbf{u}_h^p$	Flow passing through hydro plant h during time step p
$\mathcal{H}_r$	Hydro plants supplied by reservoir r			
$\mathcal{H}_n$	Hydro plants discharging into river n		$\mathbf{b}_s$	Indicator if super-generator s produces with high yield, i.e., yield between opt and max
$\mathcal{H}_m$	Hydro plants in stability zone m			
$\mathcal{K}_g$	Configurations involving generator g		$\mathbf{q}_w^p$	Worst north first-contingency production loss at time step p
$\mathcal{K}_h$	Configurations of hydro plant h			
$\mathcal{K}_s$	Configurations involving super-generator s		$\mathbf{q}_{\mathsf{w}}^p$	Worst FCPL power at time step p
$\mathcal{L}_z^-$	Incoming links of zone z		$\mathbf{q}_g^p$	Power produced by generator g during time step p
$\mathcal{L}_z^+$	Outgoing links of zone z		$\bar{\mathbf{q}}_g^p$	Available power increase of generator g during time step p
$\mathcal{N}_r$	Rivers ending in reservoir r			
$\mathcal{S}_p$	Super-generators active at time step p		$\mathbf{q}_h^p$	Power produced by hydro plant h in time step p
$\mathcal{C}_r$	Spillways available to reservoir r			
$\mathcal{C}_n$	Spillways discharging into river n		$\underline{\mathbf{q}}_h^p$	Available power decrease of hydro plant h in time step p
s_g^p	Super-generator containing generator g during time step p			
p_s	Time step of super-generator s		$\bar{\mathbf{q}}_h^p$	Available power increase of hydro plant h in time step p
r_s	Reservoir associated with super-generator s			
p_e	Time step at which topology constraint e is active		$\overleftarrow{\mathbf{q}}_l^p$	Power coming into link l during time step p
p_f	Time step at which first-contingency production loss (FCPL) f is active		$\overrightarrow{\mathbf{q}}_l^p$	Power leaving link l during time step p
			$\mathbf{q}_s^y$	Power produced by super-generator s at yield y
	Parameters		$\mathbf{q}_s$	Power produced by super-generator s
$\alpha_{g,k}^{y,d}$	Power of generator g at yield y and drop height d when configuration k is active		$\tilde{\mathbf{q}}^s$	Max power that can be produced by super-generator s during transient events
$\alpha_{s,k}^{y,d}$	Power of super-generator s at yield y and drop height d when configuration k is active		$\hat{\mathbf{q}}_p$	Max power that can be produced for time step p during transient events
$\underline{\alpha}_{s,k}^{y,d}$	Available power decrease of super-generator s at yield y and drop height d when configuration k is active		$\underline{\mathbf{q}}_s$	Available power decrease of super-generator s
			$\bar{\mathbf{q}}_s$	Available power increase of super-generator s
$\bar{\alpha}_{s,k}^{y,d}$	Available power increase of super-generator s at yield y and drop height d when configuration k is active		$\mathbf{q}_s^{pfc}$	Available reserve power of super-generator s
			$\mathbf{a}_r^p$	Average level of reservoir r during time step p
$\upsilon_{s,k,y,d}^{\text{trans}}$	Maximum power output margin of super-generator g at yield y and drop height d when configuration k is active during transient events		$\mathbf{v}_r^p$	Volume in reservoir r at the end of time step p
			$\overleftarrow{\mathbf{v}}_n^p$	Volume entering river n during time step p

$\overrightarrow{\mathbf{v}}_n^p$	Volume leaving river n during time step p		limit of power going into link l at time step p
$\mathbf{v}_c^p$	Volume passing through spillway c during time step p	$\underline{\boldsymbol{\Lambda}}_l^p$	Piecewise linear expression representing the lower
$\mathbf{w}_{s,k}^{y,d}$	Production weight of super-generator s at yield y and		limit of power coming from link l at time step p
	drop height d when configuration k is active	$\underline{\boldsymbol{\Pi}}_l^p$	Piecewise linear expression representing the upper

Expressions

limit of power coming from link l at time step p

$\boldsymbol{\Psi}_e$	Piecewise linear expression representing	
	the upper limit of topology constraint e	

Functions

$\bar{\boldsymbol{\Lambda}}_m^p$	Piecewise linear expression representing the lower	$\Xi_{m,z,z'}^p(\cdot)$
	limit of power going into the MTDC grid at time step p	

$\Xi_{m,z,z'}^p(\cdot)$ Linear function correcting a power transiting through the MTDC grid between zones z and z' at time step p by removing its loss

$\bar{\boldsymbol{\Pi}}_m^p$	Piecewise linear expression representing the upper
	limit of power going into the MTDC grid at time step p

$\Upsilon_p^{\text{pfc}}(\cdot)$ Piecewise linear function computing the power limit at time step p

$\underline{\boldsymbol{\Lambda}}_m^p$	Piecewise linear expression representing the lower
	limit of power coming from the MTDC grid at time step p

$\Xi_l^p(\cdot)$ Linear function correcting a power transiting through link l at time step p by removing its loss

$\underline{\boldsymbol{\Pi}}_m^p$	Piecewise linear expression representing the upper
	limit of power coming from the MTDC grid at time step p

$\Psi_r(\cdot)$ Linear function approximating the level of reservoir r from its volume

$\bar{\boldsymbol{\Lambda}}_l^p$	Piecewise linear expression representing the lower
	limit of power going into link l at time step p

$\Omega_p(\cdot)$ Piecewise linear function computing a stability threshold at time step p from the worst north and south FCPL

$\bar{\boldsymbol{\Pi}}_l^p$	Piecewise linear expression representing the upper

1   Introduction

Hydro-Quebec (HQ) is a vertically integrated utility responsible for electricity generation, transmission, and distribution in the province of Quebec. Its large grid comprises over 60 hydroelectric facilities and more than 260,000 Km transmission and distribution lines, of which over 11,000 Km are covered by 735kV lines, making it the most extensive network in North America. The long transmission lines and large hydroelectric dams in the far north give the power grid its unique architecture, enabling the power generation to reach most destinations located thousands of kilometers in the south of Quebec. Grid stability is a big concern that the company must consider when planning operations. The stability aspects include multiple, nonlinear phenomena, such as voltage, frequency, and angular stabilities. However, since the generation and operations planning tools are mainly based on mixed integer linear programming (MILP), integrating stability considerations becomes quite challenging. Hence, this paper aims to tackle the transient stability constrained unit commitment (TSCUC) problem at HQ. Starting from a detailed formulation of the problem (Besner et al., 2024), an investigation of the complexity sources is carried out, and a decomposition matheuristic is proposed, which significantly improve the solving of realistic instances provided by HQ.

The UC problem is fundamental in the electric power industry. It aims to determine an optimal schedule that ensures fulfilling the electricity demand at a minimum cost for the system as a whole, by deciding which generation unit should be activated and at which intensity. In the literature, there have been many relevant mathematical optimization models for the UC problem (Anjos et al., 2017). Contingency analysis is a computer application that uses a simulated model of the power system. This analysis is one of the crucial components in power system security analysis since it evaluates outage events that affect the power system performance (Musto, 2020). Compared to other cases like ERCOT (Hui, 2013), HQ has had an everlasting interest in online real-time security-based monitoring systems. As early as 1982, HQ put the first operational tool to provide dynamic transfer limits during real-time operations from the control center. Over several decades, it integrated more sophisticated and complicated tables into their system called LIMSEL (Huang et al., 2012). Mathematically speaking, contingency adds several non-linear constraints to the optimization model. This problem, known in the literature as the min-max linear programming (mmLP), involves solving linear programs having min and max functions added to their constraints. One of the ways to tackle these constraints is linearization to maintain a MILP formulation (Burks and Sakallah, 1993). More details about the UC problem and its resolution techniques are in the recent survey of Montero et al. (2022).

This paper highlights a generic and practical approach to tackling the TSCUC. The contribution is fourfold: (1) an exploratory analysis identifying the complexity sources when solving the TSCUC problem at HQ, (2) an efficient decomposition matheuristic, and multiple variants of the method, that successfully solve instances of the problem, (3) a computational study on real-world instances to assess the efficiency of the proposed method compared to an out-of-the-box CPLEX formulation, and (4) a post-computational analysis that highlights managerial insights.

The remainder of the paper is as follows. The problem formulation is summarized in Section 2. Section 3 and Section 4 present the exploratory analysis and the solution methodology, respectively. The computational results and managerial insights are detailed in Section 5.

2 Problem description

In this section, the TSCUC is detailed in the MILP framework. The model is essentially an adaptation of the one given in Besner et al. (2024), with a more compact notation. In Subsection 2.1, the main power grid components are described for modeling the TSCUC in a hydraulic context. Then, in Subsection 2.2, the MILP formulation of the problem is detailed.

2.1 Representation of the power grid

Timeline. A time reference is required, because the states of the model's components are subject to change. We denote by $\mathcal{P}$ the set of all time steps in the horizon. In particular, the *initial time step* is denoted by p_0. It marks the beginning of the simulation and has a null duration.
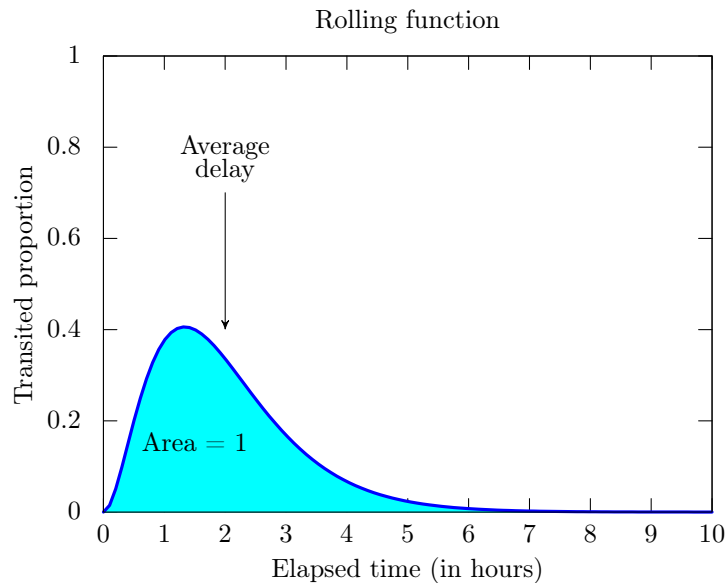
Figure 1: A theoretical rolling function

Hydraulic network. More than 90% of HQ's production comes from hydro plants, so the hydraulic network must be adequately modelized. In that spirit, it is represented as a directed graph whose nodes represent *reservoirs* (where water can be accumulated), *rivers* (where water can travel to another downstream reservoir), *spillways* (which can be used to evacuate water without turbinating it) and *hydro plants* (where water flows are converted to power), and whose edges indicate the direction in which water travels. Moreover, since water does not travel at a constant speed down a river, a *rolling function* is associated with every river n. This function is of the form $\Phi_n : \mathbb{R}^+ \to [0, 1]$, where each elapsed time t is mapped onto the proportion

of water $\Phi_n(t)$, which has reached the end of river n. To guarantee the conservation of water volume, the rolling function curve should cover an area equal to 1, as visualized in Figure 1. This function is not known, but it can be approximated by estimating the coefficients $\rho_{n,p}^{p'}$, whose value equals the proportion of water that reached the end of river n between time steps p and p'.

Production of hydro units. Once water reaches the hydro plant and the hydro units inside it, its flow can be converted to power as follows. Let $\rho \approx 997$ be the volumic mass of water (in kg/m^3), $g \approx 9.81$ be the gravity acceleration (in m/s^2), Q be the turbined flow (in m^3/s), H be the drop height (in meter), and $\eta(Q) < 1$ be the generator yield when the flow is Q, the power produced by a hydro generator is given by Ginocchio and Viollet (2012)

$$P(H,Q) = \rho \cdot g \cdot Q \cdot H \cdot \eta(Q) \tag{1}$$

The formula (1) has two disadvantages. First, it is nonlinear. Second, estimating η is complex in practice, and may depend on the state of the other hydro generators inside the hydro plant. Hence, to maintain a MILP context, a piecewise linear function can approximate the feasible power surface (Borghetti et al., 2008). More precisely, the possible drop heights can be linearized by interpolating between the fixed minimum (*low*) and maximum (*hi*) values. Similarly, to take into account the concave shape of hydro unit production, the possible *yields* of each generator can be discretized using 3 possible values, based on whether the generator outputs its maximum power (*max*), its optimal power (*opt*), or its lowest power (*min*). Also, in the event of a stability event, generators can produce more than *max* for a brief period of time. This special yield is denoted by *stab*.

Super-generator. Even if each hydro generator has its unique characteristics, it is possible (and even desirable) to make uniform decisions over groups of similar generators for operational reasons. Following this idea, the concept of *super-generator* was introduced in Besner et al. (2024). A super-generator is basically a collection of generators whose production should be managed in a similar manner at a given time step. More precisely, two generators g and g' are considered equivalent if each of the following prerequisites is satisfied: (1) g and g' belong to the same hydro plant, (2) neither g nor g' is subject to some operational restriction, (3) g and g' do not belong to different FCPL sets and (4) g and g' do not belong to different topology constraints. Hence, a super-generator is simply a class of the equivalence relation described above.

Hydro plant configurations. As mentioned in the previous paragraphs, the production of a hydro generator depends on its unique features, as well as on the presence or absence of other active generators within the same hydro plant. To take this reality into account, the notion of *(hydro plant) configurations* is introduced. Configurations are simply subsets of the set of generators belonging to the hydro plant. Theoretically, given that a generator is either active or inactive, there can be up to $2^{|G_h|}$ possible configurations for each hydro plant h where G_h is the set of generators of h, which could potentially makes the MILP hard to solve. Fortunately, in practice, many of those configurations need not be considered. More precisely, by taking into account the planned unavailabilities of some generators for maintenance, the power and flow restrictions for operational purposes, and the first-contingency production loss (FCPL) constraints and topology-based constraints (TC) (detailed in the following paragraphs), it is possible to significantly reduce the number of *interesting* hydro plant configurations. An algorithm enumerating those configurations is described in Besner et al. (2024).

Transmission network. As for the hydraulic network, the transmission network can be represented as a directed graph, whose nodes are *zones* and *links*, and whose directed edges indicate the direction in which power is transmitted. Although electricity can circulate in both directions, the majority of the network has a radial structure, i.e. most electricity always travels in the same direction (from the north to the south). Because of this characteristic, it is possible to neglect the impedance model and use linear approximation of the losses to describe the flows between the zones. HQ also manages a *multi-terminal direct current* (MTDC) grid, requiring more intricate modeling due to its unique operational constraints. In this paper, for simplicity's sake, the MTDC is represented as a special link.

Stability limits. Power circulating through transportation links and the MTDC grid must be monitored and constrained for various stability and operational reasons. In article Besner et al. (2024), it is described how transient simulation results are aggregated as min-max functions that are then linearized using binary variables. Those linearized constraints are used to monitor the flow over the link between the zones. In Section 3.2.2, the complexity created by this formulation will be discussed and Section 4.3 will propose a novel linearization that eliminates the need for binary variables.

Frequency curtailment reserves. Finally, in addition to stability limits, it is also important to monitor adequate reserve levels in case of a loss of generation that leads to frequency events. Two reserves are monitored in the TSCUC problem: *primary frequency control* (PFC) and *secondary frequency control* (SFC). PFC levels are determined from so-called *first-contingency production loss* (FCPL) sets of generators, i.e. PFC guarantees adequate response for the primary controller of each generator. Meanwhile, SFC ensures adequate margins for the area generation control (AGC) algorithm, which maintains the frequency to the nominal 60Hz during normal operations.

2.2   Mathematical model

Following the power grid representation described in Subsection 2.1, the underlying mathematical model can be introduced. For sake of readability, the notation has been adapted from Besner et al. (2024) according to the following convention. Sets and relations are in calligraphic style, parameters are in lowercase Greek letters, decision variables are in bold lowercase letters, and expressions and functions are in uppercase Greek letters (See the alphabetically ordered Notation Section for more details). Physical quantities are measured with respect to the international system of units, i.e., power is in megawatts (MW), time is in seconds (s), distance is in meters (m), volume is in m^3, and flow is in m^3/s.

Objective function. From the point of view of the transmission grid dispatchers, the main objective is the minimization of the number of maneuvers that need to be performed, which can be expressed as

$$\min \sum_{g \in \mathcal{G}} \sum_{p \in \mathcal{P}} \beta_g^p |\mathbf{x}_g^p - \mathbf{x}_g^{p-1}| \tag{1}$$

The absolute values can be linearized by introducing variables $\Delta_g^p = |\mathbf{x}_g^p - \mathbf{x}_g^{p-1}|$ for each $g \in \mathcal{G}$ and $p \in \mathcal{P}$, so that the objective can be rewritten as

$$\min \sum_{g \in \mathcal{G}} \sum_{p \in \mathcal{P}} \beta_g^p \Delta_g^p \tag{2}$$

and by adding the following constraints

$$\Delta_g^p \geq \mathbf{x}_g^p - \mathbf{x}_g^{p-1} \qquad\qquad \forall g \ \forall p \in \mathcal{P}' \tag{3}$$

$$\Delta_g^p \geq \mathbf{x}_g^{p-1} - \mathbf{x}_g^p \qquad\qquad \forall g \ \forall p \in \mathcal{P}' \tag{4}$$

A second important objective is to minimize water spilling:

$$\min \sum_{r \in \mathcal{R}} \sum_{p \in \mathcal{P}'} \sum_{c \in \mathcal{C}_r} \beta_c^p \mathbf{v}_c^p \tag{5}$$

The objective function is a linear combination of the two objectives above (using weights β_g^p and β_c^p).

Water conservation constraints. The following constraints ensure water conservation throughout all the hydraulic components, i.e. hydro plants, reservoirs, rivers, and spillways:

$$\mathbf{v}_r^{p_0} = \nu_r \qquad\qquad \forall r \tag{6}$$

$$\mathbf{v}_r^p = \mathbf{v}_r^{p-1} + \sum_{n \in \mathcal{N}_r} \overrightarrow{\mathbf{v}}_n^p - \sum_{h \in \mathcal{H}_r} \delta_p \mathbf{u}_h^p - \sum_{c \in \mathcal{C}_r} \mathbf{v}_c^p \qquad \forall r \; \forall p \in \mathcal{P}' \tag{7}$$

$$\mathbf{a}_r^p = \Psi_r((\mathbf{v}_r^{p-1} + \mathbf{v}_r^p)/2) \qquad \forall r \; \forall p \in \mathcal{P}' \tag{8}$$

$$\overleftarrow{\mathbf{v}}_n^p = \sum_{h \in \mathcal{H}_n} \delta_p \mathbf{u}_h^p + \sum_{c \in \mathcal{C}_n} \mathbf{v}_c^p \qquad \forall n \; \forall p \tag{9}$$

$$\overrightarrow{\mathbf{v}}_n^p = \sum_{p' \le p} \rho_{n,p'}^p \overleftarrow{\mathbf{v}}_n^{p'} \qquad \forall n \; \forall p \tag{10}$$

Production constraints. The next constraints express the core of the unit commitment problem, taking into account the production characteristics of generators:

$$\mathbf{b}_s \ge \sum_{k \in \mathcal{K}_s} \sum_{d \in \mathcal{D}} \mathbf{w}_{s,k}^{max,d} \qquad \forall s \tag{11}$$

$$1 - \mathbf{b}_s \ge \sum_{k \in \mathcal{K}_s} \sum_{d \in \mathcal{D}} \mathbf{w}_{s,k}^{min,d} \qquad \forall s \tag{12}$$

$$1 = \sum_{k \in \mathcal{K}_h} \mathbf{x}_k^p \qquad \forall h \; \forall p \tag{13}$$

$$\mathbf{x}_k^{p_s} = \sum_{y \in \mathcal{Y}} \sum_{d \in \mathcal{D}} \mathbf{w}_{s,k}^{y,d} \qquad \forall k \; \forall s \tag{14}$$

$$\mathbf{x}_g^p = \sum_{k \in \mathcal{K}_g} \mathbf{x}_k^p \qquad \forall g \; \forall p \tag{15}$$

The power of the hydro plants, hydro plant configurations, generators, and super-generators can be computed once all units are committed. Furthermore, we can also compute the level variations of the reservoirs and the flows turbinated by the hydro plants. This is expressed through the following constraints:

$$\mathbf{q}_h^p = \sum_{g \in G_h} \mathbf{q}_g^p \qquad \forall h \; \forall p \tag{16}$$

$$\mathbf{q}_g^p = \sum_{k \in \mathcal{K}_g} \sum_{y \in \mathcal{Y}} \sum_{d \in \mathcal{D}} \alpha_{g,k}^{y,d} \mathbf{w}_{s_g^p,k}^{y,d} \qquad \forall g \; \forall p \tag{17}$$

$$\mathbf{q}_s^y = \sum_{k \in \mathcal{K}_s} \sum_{d \in \mathcal{D}} \alpha_{s,k}^{y,d} \mathbf{w}_{s,k}^{y,d} \qquad \forall s \; \forall y \tag{18}$$

$$\mathbf{q}_s = \sum_{y \in \mathcal{Y}} \mathbf{q}_s^y \qquad \forall s \tag{19}$$

$$\mathbf{u}_h^p = \sum_{g \in G_h} \sum_{k \in \mathcal{K}_{s_g^p}} \sum_{y \in \mathcal{Y}} \sum_{d \in \mathcal{D}} \phi_{g,k}^{y,d} \mathbf{w}_{s_g^p,k}^{y,d} \qquad \forall h \; \forall p \tag{20}$$

$$\mathbf{a}_{r_s}^{p_s} = \sum_{k \in \mathcal{K}_s} \sum_{y \in \mathcal{Y}} \sum_{d \in \mathcal{D}} \lambda_{r_s}^d \mathbf{w}_{s,k}^{y,d} \qquad \forall s \tag{21}$$

Transmission constraints. The power circulating through the zones, the transportation links, and the MTDC grid must balance with the non dispatchable power (transactions at interconnectors, expected zone load, interrupted powers, non controllable planned generation) in each zone:

$$\alpha_z^p = \sum_{h \in \mathcal{H}_z} \mathbf{q}_h^p + \sum_{l \in \mathcal{L}_z^-} \overrightarrow{\mathbf{q}}_l^p - \sum_{l \in \mathcal{L}_z^+} \overleftarrow{\mathbf{q}}_l^p \tag{22}$$

$$+ \overrightarrow{\mathbf{q}}_{m,z}^p - \overleftarrow{\mathbf{q}}_{m,z}^p \qquad \forall z \; \forall p \tag{23}$$

Moreover, as mentioned earlier, the transit going through each link and the MTDC grid must take into account the losses, and is also constrained by piecewise linear expressions:

$$\overrightarrow{\mathbf{q}}_l^p = \Xi_l^p(\overleftarrow{\mathbf{q}}_l^p) \qquad \forall l \; \forall p \tag{24}$$

$$\mathbf{\Lambda}_l^p \leq \overrightarrow{\mathbf{q}}_l^p \leq \mathbf{\Pi}_l^p \qquad\qquad \forall l \ \forall p \tag{25}$$

$$\bar{\mathbf{\Lambda}}_l^p \leq \overleftarrow{\mathbf{q}}_l^p \leq \bar{\mathbf{\Pi}}_l^p \qquad\qquad \forall l \ \forall p \tag{26}$$

$$\overrightarrow{\mathbf{q}}_{m,z}^p = \Xi_{m,z,z'}^p(\overleftarrow{\mathbf{q}}_{m,z'}^p) \qquad\qquad \forall z \ \forall z' \ \forall p \tag{27}$$

$$\underline{\mathbf{\Lambda}}_m^p \leq \overrightarrow{\mathbf{q}}_{m,z}^z p \leq \underline{\mathbf{\Pi}}_m^p \qquad\qquad \forall z \ \forall p \tag{28}$$

$$\bar{\mathbf{\Lambda}}_m^p \leq \overleftarrow{\mathbf{q}}_{m,z}^p \leq \bar{\mathbf{\Pi}}_m^p \qquad\qquad \forall z \ \forall p \tag{29}$$

Stability constraints. The total production of each topology constraint e generated by the LIMSEL tool is capped by a piecewise linear limit expression $\mathbf{\Psi}_e$, with a protection margin:

$$\sum_{g \in \mathcal{S}_e} (\mathbf{q}_s^{p_e} + \bar{\mathbf{q}}_s^{p_e}) \leq \mathbf{\Psi}_e \qquad\qquad \forall e \tag{30}$$

The following set of constraints is used to ensure sufficient power reserve for secondary frequency control (SFC):

$$\bar{\mathbf{q}}_s = \sum_{k \in \mathcal{K}_s} \sum_{y \in \mathcal{Y}} \sum_{d \in \mathcal{D}} \bar{\alpha}_{s,k}^{y,d} \ \mathbf{w}_{s,k}^{y,d} \qquad\qquad \forall s \tag{31}$$

$$\underline{\mathbf{q}}_s = \sum_{k \in \mathcal{K}_s} \sum_{y \in \mathcal{Y}} \sum_{d \in \mathcal{D}} \underline{\alpha}_{s,k}^{y,d} \ \mathbf{w}_{s,k}^{y,d} \qquad\qquad \forall s \tag{32}$$

$$\bar{\mathbf{q}}_h^p = \sum_{s \in Sg_{h,p}} \bar{\mathbf{q}}_s \qquad\qquad \forall h \ \forall p \tag{33}$$

$$\underline{\mathbf{q}}_h^p = \sum_{s \in Sg_{h,p}} \underline{\mathbf{q}}_s \qquad\qquad \forall h \ \forall p \tag{34}$$

$$\sum_{h \in \mathcal{H}} \underline{\mathbf{q}}_h^p \geq \underline{\xi}^{sfc} \qquad\qquad \forall p \tag{35}$$

$$\sum_{h \in \mathcal{H}} \bar{\mathbf{q}}_h^p \geq \bar{\xi}^{sfc} \qquad\qquad \forall p \tag{36}$$

$$\sum_{h \in \mathcal{H}} \left(\underline{\mathbf{q}}_h^p + \bar{\mathbf{q}}_h^p \right) \geq \bar{\eta}^{sfc} \qquad\qquad \forall p \tag{37}$$

The next constraints compute the maximum output of all generators during a frequency event. This output is used to approximate the system inertia and strength. It also allows to determine the maximum value of the worst FCPL and the required primary frequency control (PFC) margins.

$$\tilde{\mathbf{q}}^s = \sum_{k \in \mathcal{K}_s} \sum_{d \in \mathcal{D}} \sum_{y \in \mathcal{Y}} \alpha_{s,k}^{stab,d} \ \mathbf{w}_{s,k}^{y,d} \qquad\qquad \forall s \tag{38}$$

$$\hat{\mathbf{q}}_p = \sum_{s \in \mathcal{S}_p} \tilde{\mathbf{q}}^s \qquad\qquad \forall p \tag{39}$$

Finally, the PFC margins are tied to the worst FCPL and can be constrained as follows:

$$\mathbf{q}_s^{pfc} = \sum_{k \in \mathcal{K}_s} \sum_{y \in \mathcal{Y}} \sum_{d \in \mathcal{D}} \upsilon_{s,k,y,d}^{\text{trans}} \ \mathbf{w}_{s,k}^{y,d} \qquad\qquad \forall s \tag{40}$$

$$\mathbf{q}_w^p \leq \Upsilon_p^{\text{pfc}}(\hat{\mathbf{q}}_p) \qquad\qquad \forall p \tag{41}$$

$$\mathbf{q}_w^{p_f} \geq \sum_{g \in G_f} (\mathbf{q}_s + \bar{\mathbf{q}}_s) \qquad\qquad \forall f \tag{42}$$

$$\sum_{s \in \mathcal{S}_p} \mathbf{q}_s^{pfc} \geq \Omega_p(\mathbf{q}_w^p) \qquad\qquad \forall p \tag{43}$$

$$\sum_{h \in \mathcal{H}_m} \sum_{s \in Sg_{h,p_s}} \mathbf{q}_s^{pfc} \geq \min(\xi_m^{abs}, \xi_m^{rate} \hat{\mathbf{q}}_p) \qquad\qquad \forall m \tag{44}$$

3 Exploratory analysis

This section aims to explore the root causes of the TSCUC complexity. We first outline the problematic, conduct a thorough model analysis, and highlight the relevant results.

3.1 Problematic

Over several TSCUC instances, HQ's operators used to rely on simulations followed by manual tuning. While these solutions are implementable in practice, they do not guarantee quality and they are time-consuming. Another weakness of the manually-tuned methods lies in their inability to incorporate all the constraints imposed by the operational rules of the industry. Furthermore, these methods imply the need to solve instances regularly to incorporate the perturbations that can impact the data. For example, demand is subject to market fluctuations and is likely to change frequently. Also, the production process is subject to disruptions due to maintenance requirements or group failures. Thus, recovery in short intervals of time is a must.

3.2 Model analysis

To detect the TSCUC's complexity sources, we conducted an exploratory analysis. The latter investigates each component of the TSCUC model, i.e., the objective function, the decision variables, and the constraints. We used a representative instance of the TSCUC's model for the investigation. The default parametrization of the CPLEX version 20.1.0.0 was used for all experiments.

3.2.1 Objective function

The objective function does not show any complexity, since it mainly minimizes the maneuvers number and water spilling.

3.2.2 Constraints

All model constraints are linear except the stability limits constraints discussed in Section 2. One of the main challenges is incorporating those transient stabilities while maintaining the TSCUC as a MILP model. The functions L and U produced by the LIMSEL tool are arbitrarily nested compositions of linear expressions, together with the n-ary max and min functions.

Currently, these constraints are linearized following the four elementary cases: $x \geq min(y, z)$, $x \leq min(y, z)$, $x \geq max(y, z)$, and $x \leq max(y, z)$, where x, y, z are some continuous variables. The first and fourth elementary cases require adding binary and big-M constraints into the model, which makes it more complex to solve.

Using the classic linearization above, we confirm that, without the transient stability constraints, the representative TSCUC instance becomes relatively easier to solve. On the contrary, with these constraints, the model is more difficult and requires huge computational time to find sufficiently good solutions. It suggests that the MILP with these constraints is significantly less loosely coupled compared to the MILP without these constraints. For the remaining constraints, the investigation shows that they do not have a significant impact on the solution time and the gap. Based on these findings, we conclude that the main source of constraints complexity is the transient stability constraints.

3.2.3 Decision variables

For decision variables, the other difficulty to tackle is the hydro plant configurations. Although the number of hydro plant configurations can be reduced by exploiting generators symmetries, the number of possible configurations remains relatively high in the worst case. Under this case, when exploring the space of possible configurations during the solving, the branching seems hard to optimize.

To further explore variables, we observed that the solution of the TSCUC's linear relaxation (LR) is obtained rapidly and is very fractional. A large time amount is then consumed in the branch and bound ($B\&B$) process to find a feasible solution before improving it. Based on this observation, it is necessary to check the impact of the binary decision variables on the solution process. As per Section 2, we have three classes of binary variables, $\mathcal{B}_c, c \in \mathcal{C}$ where $\mathcal{C}$ denotes the set of classes. These classes are namely: the set of binary variables $\mathbf{x}_k^p$ related to active configurations, the set of binary variables $\mathbf{x}_g^p$ related to active generators, and the set of binary variables $\mathbf{b}_s$ related to super-generators production yields.

To distinguish which classes of binary variables are at the origin of the slow $B\&B$ process, we consider two measures: the percentage of each class among all the binary variables (% Class) and the percentage of non-zero variables per class in the solution of the representative instance (% Non-zeros). To be more realistic, we consider a representative instance of the worst-case scenario. Table 1 shows these measures for each class of binary variables.

Table 1: Binary classes

Class	Index	# Variables	% Class	# Non-zeros	% Non-zeros
$\mathbf{x}_k^p$	1	17759	**88.41**	624	**3.51**
$\mathbf{x}_g^p$	2	384	1.91	14	3.65
$\mathbf{b}_s$	3	1944	9.68	1011	52.01

We first observe that variables $\mathbf{x}_k^p$ are dominant in the set of the binary variables with a percentage of 88.41%. Second, only 3.51 % of these variables take non-zero values in the obtained solution. Thus, these variables are more likely to impact significantly the solution process. To validate this hypothesis, we designed an investigation strategy, for which the mechanism is as follows. We solve the TSCUC using default CPLEX for a predefined time limit and collect the best solution found. Using the latter and considering one class of binary variables at a time, we fix the binary variables worth 1 in the solution and solve the TSCUC again using the optimal solution found as a warm-start. By doing this, we seek to check if the problem becomes easier when fixing some binary variables from each class. We describe the investigation strategy in Algorithm 1.

Algorithm 1: Investigation strategy

    Input: x, α
1   $x^0 = \textsc{SolveMILP}(x, \alpha)$
2   **for** $c \in \{1, 2, 3\}$ **do**
3   **forall** $x_n \in \mathcal{B}_c$ **do**
4   **if** $x_n^0 = 1$ **then**
5             $\textsc{SetBounds}(x_n, 1, 1)$
6   **end**
7   **end**
8       $\textsc{SetWarmStart}(x^0)$
9       $x^* = \textsc{SolveMILP}(x, \alpha)$
10   **forall** $x_n \in \mathcal{B}_c$ **do**
11       $\textsc{SetBounds}(x_n, 0, 1)$
12   **end**
13 **end**

Formally, we first solve the TSCUC for a predefined time limit α using function $\textsc{SolveMILP}(x, \alpha)$, where x is a vector of variables. The optimal integer solution found is saved in a vector x^0. We denote by x_n and x_n^0 the elements of vectors x and x^0, respectively. For each class of binary variables, we fix to 1 the variables of class c that are worth 1 in x^0 using the function $\textsc{SetBounds}(x_n, 1, 1)$ where $x_n \in \mathcal{B}_c$. For a given $l, u \in \mathbb{R}$ and a variable $x \in \mathbb{R}^n$ where $n \in \mathbb{N}$, function $\textsc{SetBounds}(x, \underline{x}, \bar{x})$ set the variable's lower bound to $\underline{x}$ and its upper bound to $\bar{x}$. In addition to the fixation, the optimal solution found x^0 is given as a warm-start using the function $\textsc{SetWarmStart}(x^0)$. It gives rise to a restricted sub-problem that is solved again in the same amount of time α. We denote by x^* the returned solution and x_n^* its elements.

We conducted three variable fixation scenarios. Table 2 gives the results on the representative instance. For each scenario, we report the number of binary variables in the model after fixation (# Binaries). We ran each scenario for one hour, and if no solution was found, the gap was set to 100%. The $\mathbf{x}_k^p$ variables proved to be the ones having a significant impact on the computational time. While the original model struggles to find the first gap, both the first gap (1st Gap) and the optimal Gap (Opt Gap) for the restricted $\mathbf{x}_k^p$ scenario problem are found rapidly, within 33s (1st Gap) and 96s (Opt Gap), respectively. For fixation scenarios related to the other classes of binary variables, the times required to reach the first integer and the optimal solutions was beyond one hour. In some cases, no integer solution was found.

Table 2: Investigation strategy results

Scenario	# Binaries	1st Gap(%)	1st Time (s)	Opt Gap(%)	Opt Time (s)
Original	20087	100%	3600	100.00	–
$\mathbf{x}_k^p$	15095	63	33	0.00	96
$\mathbf{x}_g^{\beta}$	314	100	3600	100.00	–
$\mathbf{b}_s$	1671	100	3600	100.00	–

These findings confirm the hypothesis, i.e., the variables $\mathbf{x}_k^p$ impact the solution process. We reaffirm this by the following observation. For each hydro plant, hundreds of possible group configurations could be active in a given time step. These possibilities give rise to high symmetry within the model, increasing the CPLEX time to reach satisfactory solutions, thus making the commercial solver impractical. Furthermore, from all these possible configurations, at most one configuration should be selected, thus making the $\mathbf{x}_k^p$ variables highly fractional in the LR.

4 Solution methodology

In this section, we first highlight the actions taken to tackle the different sources of complexity identified previously. Then, we present the decomposition matheuristic. After that, we implement a practical variant for which the decomposition criterion considered is time. We also discuss a new linearization for the transient stabilities. Lastly, we highlight the role of the solver's tuning.

4.1 Prescriptions against complexity

From the exploratory analysis, we generate the following. For the constraints, the classic linearization of the power stability constraints adds a lot of binary variables and big-M constraints into the model. For variables, the $\mathbf{x}_k^p$ variables permit the identification of another side from which we can tackle the MILP problem. The relaxed minimization of the maneuvers number provides a good lower bound. Thus, designing an efficient matheuristic that will rapidly provide good upper bounds and effectively tackle the complexities above is a suitable strategy in our context. Such a strategy will allow the elimination of unpromising branches in the B&B tree, making the solving process faster. We tackle the problem from three dimensions: decomposition of the time horizon, linearization of constraints, and solver's tuning. We present next the designed decomposition matheuristic, called hereafter the Relax-Solve-Fix-Correct ($\mathcal{RSFC}$) matheuristic. Then, we highlight the linearization of transient stability constraints before discussing the importance of tuning the CPLEX solver's setting.

4.2 $\mathcal{RSFC}$ Matheuristic

We summarize the $\mathcal{RFSC}$ matheuristic in Algorithm 2. It is inspired by successful matheuristics in the literature (Er Raqabi et al., 2023). It consists of decomposing the time horizon into smaller time intervals, called hereafter, windows. Then, we optimize by rolling through these windows. The number of windows (Φ) is obtained by dividing the time horizon length by the number of windows. The matheuristic has four phases: *Relax*, *Solve*, *Fix*, and *Correct*. In the *Relax* phase, we pick the first window (ϕ), and we keep all

corresponding $\mathbf{x}_k^p$ variables binary. All other binary variables in the MILP model are relaxed (integrality). A variable $\mathbf{x}_k^p$ corresponds to window ϕ if its index p belongs to the time interval of this window. In the *Solve* phase, we solve the MILP. In the *Fix* phase, we fix the $\mathbf{x}_k^p$ corresponding to window ϕ to their solution values. We move into the second window, and the process continues until rolling through the whole time horizon. Finally, in the *Correct* phase, we maintain all the fixed $\mathbf{x}_k^p$ variables, restore all the other binary variables of the MILP model to their binary state, and solve the MILP model. If the model is feasible, we return the solution x^*. Otherwise, we correct by tackling unfeasibilities. This is done using function $\textsc{CorrectMILP}(x, \alpha)$. In this function, we relax conflicting $\mathbf{x}_k^p$ and solve again. The process continues until reaching feasibility. We return solution x^*. It is worth mentioning that instead of partitioning the time horizon, we keep the overlapping between consecutive windows. Such a strategy transfers information between windows and consequently ensures better results.

Algorithm 2: $\mathcal{RSFC}$ **Matheuristic**

   Input: x, α, Φ, $\phi \leftarrow 1$
1   **while** $\phi < \Phi$ **do**
2   Relax $\mathbf{x}_k^p$ variables out of the current window ϕ
3   Relax all other binary variables
4      $x^* \leftarrow \textsc{SolveMILP}(x, \alpha)$
5   Fix the $\mathbf{x}_k^p$ variables in the current window to their values
6      $\phi \leftarrow \phi + 1$
7   **end**
8   Restore all other binary variables to Binary
9   $x^* \leftarrow \textsc{SolveMILP}(x, \alpha)$
10   $x^* \leftarrow \textsc{CorrectMILP}(x, \alpha)$
11   **Return** x^*

While we discussed a time-based implementation of the $\mathcal{RFSC}$ matheuristic, it is worth mentioning that there are other decomposition strategies, such as the space-based decomposition. The latter consists of decomposing $\mathbf{x}_k^p$ based on their corresponding hydro plant.

4.3 Transient stability constraints

For the transient stability constraints (see Example 1 below), it is not necessary to linearize them exactly. When linearizing them exactly, we add complicating binary variables and big-M constraints into the model, as shown in the classic linearization discussed in the previous section (Model1 of Example 1). Furthermore, the latter does not consider the changing objective function in the model. With a changing objective function, it is possible to approximate the transient stability constraints by adding real variables into the objective function (Model2 of Example 1). Such a strategy eliminates all the binary variables and big-M constraints coming from the classic linearization, thus providing a better model that ensures close, if not similar, results. We highlight this observation in the example below.

Example 1. Let us consider the following model:

$$\textbf{Min} \quad c^T x \qquad\qquad\qquad\qquad (\text{Model})$$
$$s.t.: \ Ax + By + Cz \geq e$$
$$w \geq min(y, z)$$
$$x \in X, \ \ y \in Y, \ \ z \in Z, \ \ w \in W$$

The classic linearization of Model is:

$$\textbf{Min} \quad c^T x \qquad\qquad\qquad\qquad (\text{Model1})$$
$$s.t.: \ Ax + By + Cz \geq e$$
$$w \geq y - M + Mb$$
$$w \geq z - Mb$$

$$z - y \leq Mb$$
$$y - z \leq M(1 - b)$$
$$x \in X, \;\; y \in Y, \;\; z \in Z, \;\; w \in W$$
$$b \in \{0, 1\}^{|w|}$$

where M is a sufficiently large scalar and b is the binary vector of size $|w|$ (the size of vector variable w), such that element $b_i = 1$ if $y_i \leq z_i$, and 0 otherwise (y_i and z_i being the elements of vectors y and z, respectively).

By introducing a real vector t, we obtain the following model:

$$\textbf{Min} \;\; c^T x - \epsilon^T t \qquad\qquad\qquad\qquad\qquad \text{(Model2)}$$
$$s.t. : \;\; Ax + By + Cz \geq e$$
$$w \geq t$$
$$t \leq y$$
$$t \leq z$$
$$x \in X, \;\; y \in Y, \;\; z \in Z, \;\; w \in W$$

Model2 seems better than Model1 since it does not contain any binary variable and big-M constraint, so that we can expect to solve it more efficiently.

We can show theoretically that, under certain conditions, there exists ϵ such that Model1 and Model2 are equivalent, i.e., they have the same solution x even if the objectives are different. In practice, experiments on the TSCUC confirmed this observation. An approximate value (found by tuning) of weights in vector ϵ is enough. We conducted a few simulations to tune the value of the vector ϵ. With the same intuition presented in the example above, we linearized the complex nested min-max bounds of the transient stability constraints. We refer to the linearizations in Model1 and Model2 forms as the classic and new linearizations, respectively.

4.4   Solver's tuning

An important aspect to highlight is the solver's setting used to solve the MILP problem. The default CPLEX setting was designed based on artificial instances from the literature, such as the ones from the MIPLIB library (Gleixner et al., 2021). This default setting does not work well on real-life large-scale instances, such as the HQ's case. Thus, we conducted manual troubleshooting by exploring the log files of these instances, which highlights that few parameters impact the solver's performance (Himmich et al., 2023). For the MILP problem tackled in this paper, these parameters are *emphasis*, *heuristic effort*, and *CutsFactor* (IBM ILOG CPLEX, 2009). Then, manual troubleshooting constructs an initial configuration. This configuration is provided to the irace tuner (López-Ibáñez et al., 2016) to improve it.

Automatic tuning has practical applications in many real-life large-scale MILP problems, whose solving requires many hours of configuration testing. For these problems, one should expect several days/weeks of tuning without interruption when considering multiple instances. Thus, the idea was to consider clustering available instances from space based on the industrial (demand volume, the season of the year, etc.) and mathematical (variables, constraints, etc.) structures, select one instance from each cluster, we hereafter call representative instance, and separately tune on each of these representative instances as a single training instance. It should result in a time reduction compared to tuning on all instances that are generated or will be generated in the future, especially considering that MILP solvers are exact algorithms known to have stable behavior in MILP instances with the same mathematical and industrial structures.

Tuning on a single instance is not appropriate for all cases. It is known, in general, that the better a metaheuristic performs over the tuning instances, the worse it might do over a different instance of the same class, and therefore, on average, over the whole class (Birattari and Kacprzyk, 2009). It is also the case when

instances change significantly from one year to another, as in the forestry context, where we do not cut the same trees yearly. Still, as initial experiments on the HQ problem have shown, there is a high chance that the best configuration obtained while tuning on a given instance performs well on same-cluster instances. We also expect this is the case for large-scale industrial optimization problems with fixed installations, standard processes, and repetitive trends. For such MILP problems, instances show periodicity, similarity, and repetitiveness from one year to another.

5   Computational study

We highlight in this section the matheuristic results in the worst-case, i.e. in instances that contain a large number of possible configurations. We also provide some managerial insights.

5.1   Test plan

We distinguish two classes of instances, I^1 and I^2. The first class is for the winter season, while the second class is for the summer season. We present the features of these instances, including the number of time steps, the number of variables, the number of binary variables, and the number of constraints in Table 3.

Table 3: Instances

Type	Instance	Horizon	Variables	Binaries	Constraints
I^1	I_1^1	3	25728	4429	28397
	I_2^1	8	69401	11879	76078
	I_3^1	10	86867	14861	95150
	I_4^1	15	130367	22316	142756
	I_5^1	21	182699	31262	199942
	I_6^1	35	267335	12453	448087
	Avg		**127066**	**16200**	**165068**
I^2	I_1^2	3	23664	4636	29409
	I_2^2	8	67206	12557	80069
	I_3^2	10	84604	15714	100326
	I_4^2	15	127986	23573	150922
	I_5^2	21	179736	32915	211288
	I_6^2	35	300602	54385	351914
	Avg		**157997**	**23963**	**153988**

We compare the following methods:

- **MILP$_1$**: CPLEX under its default setting for the classic linearization.
- **MILP$_1^+$**: CPLEX with a tuned setting for the classic linearization.
- **MILP$_2$**: CPLEX under its default setting for the new linearization.
- **MILP$_2^+$**: CPLEX with a tuned setting for the new linearization.
- **RSFC$_1$**: $\mathcal{RSFC}$ matheuristic with the default setting for the classic linearization.
- **RSFC$_1^+$**: $\mathcal{RSFC}$ matheuristic with a tuned setting for the classic linearization.
- **RSFC$_2$**: $\mathcal{RSFC}$ matheuristic with the default setting for the new linearization.
- **RSFC$_2^+$**: $\mathcal{RSFC}$ matheuristic with a tuned setting for the new linearization.

The coding language is Python, and tests are conducted using version 20.1.0.0 of the IBM ILOG CPLEX solver. All experiments were carried out on a 3.20GHz Intel^R Core^{TM} i7-8700 processor, with 64GiB System memory, running on Oracle Linux Server release 7.7. We use real-time to measure runtime.

5.2 Computational results

We compare various methods by reporting the execution time (Time) in seconds and the optimality gap (Gap) in percentage. The time limit for each execution is 3 hours. When the method fails to reach a feasible solution, we report a gap of 100% gap.

We compare in Table 4 the default and tuned CPLEX results using the classic linearization. While the tuned CPLEX outperforms the default CPLEX on the small (few time steps) instances, the default and tuned CPLEX cannot reach feasible solutions for large instances.

Table 4: Default and Tuned CPLEX Results using the Classic Linearization

| Scenario | Instances | $\mathbf{MILP_1}$ | | $\mathbf{MILP_1^+}$ | |
		Time	Gap(%)	Time	Gap(%)
$\mathbf{I^1}$	I_1^1	324.96	0.00	58.75	0.00
	I_2^1	10800.00	100.00	77.88	0.00
	I_3^1	10800.00	100.00	91.42	0.00
	I_4^1	10800.00	100.00	10800.00	100.00
	I_5^1	10800.00	100.00	10800.00	100.00
	I_6^1	10800.00	100.00	10800.00	100.00
	Avg	**9054.16**	**83.33**	**5438.01**	**50.00**
$\mathbf{I^2}$	I_1^2	117.54	0.00	21.25	0.00
	I_2^2	10800.00	93.25	28.17	0.00
	I_3^2	10800.00	93.82	33.07	0.00
	I_4^2	10800.00	100.00	10800.00	100.00
	I_5^2	10800.00	100.00	10800.00	100.00
	I_6^2	10800.00	100.00	10800.00	100.00
	Avg	**9019.59**	**81.18**	**5413.75**	**50.00**

Table 5 shows the default and tuned CPLEX results using the new linearization. The new linearization outperforms the classic linearization. In particular, $MILP_2^+$ reaches a gap below 5% on several instances. The results also confirm the role of tuning in reaching better solutions quicker than the default CPLEX setting.

Table 5: Default and Tuned CPLEX Results using the New Linearization

| Scenario | Instances | $\mathbf{MILP_2}$ | | $\mathbf{MILP_2^+}$ | |
		Time	Gap(%)	Time	Gap(%)
$\mathbf{I^1}$	I_1^1	3.33	0.00	2.21	0.00
	I_2^1	23.71	0.00	15.74	0.00
	I_3^1	80.31	0.00	53.30	0.00
	I_4^1	606.52	0.00	81.59	0.00
	I_5^1	10800.00	0.03	453.05	0.58
	I_6^1	10800.00	100.00	5336.01	4.43
	Avg	**3718.98**	**16.67**	**990.32**	**0.84**
$\mathbf{I^2}$	I_1^2	6.17	0.00	3.89	0.00
	I_2^2	10.76	0.00	6.78	0.00
	I_3^2	59.50	0.00	37.51	0.00
	I_4^2	682.44	0.00	67.69	0.00
	I_5^2	3327.04	0.00	261.53	0.00
	I_6^2	10800.00	100.00	2234.76	1.44
	Avg	**2480.99**	**16.67**	**435.36**	**0.24**

Table 6 presents the matheuristic results using the classic linearization. Both the default and tuned settings do not significantly outperform the default and tuned CPLEX using the new linearization. Still, some instances are solved to optimality using the matheuristic.

Table 6: Default and Tuned Matheuristic Results using the Classic Linearization

Scenario	Instances	RSFC$_1$ Time	RSFC$_1$ Gap(%)	RSFC$_1^+$ Time	RSFC$_1^+$ Gap(%)
I^1	I_1^1	6.70	0.00	3.15	0.00
	I_2^1	18.02	0.00	12.33	0.00
	I_3^1	76.34	0.00	53.35	0.00
	I_4^1	430.10	0.00	117.45	0.00
	I_5^1	2292.22	0.00	625.95	0.00
	I_6^1	18531.57	0.00	5367.77	0.00
	Avg	**3559.16**	**0.00**	**1030.00**	**0.00**
I^2	I_1^2	4.89	0.00	3.22	0.00
	I_2^2	13.15	0.00	12.60	0.00
	I_3^2	55.72	0.00	54.54	0.00
	I_4^2	608.34	0.00	112.34	0.00
	I_5^2	2101.06	0.00	378.72	0.00
	I_6^2	13449.10	0.00	2649.54	0.00
	Avg	**2705.38**	**0.00**	**535.16**	**0.00**

Table 7 shows the default and tuned matheuristic results using the new linearization. The results are significantly better, especially for large instances. Several instances are solved to optimality within less than 10 minutes.

Table 7: Default and Tuned Matheuristic Results using the New Linearization

Scenario	Instances	RSFC$_2$ Time	RSFC$_2$ Gap(%)	RSFC$_2^+$ Time	RSFC$_2^+$ Gap(%)
I^1	I_1^1	5.39	0.00	2.68	0.00
	I_2^1	33.45	0.00	13.71	0.00
	I_3^1	49.36	0.00	35.88	0.00
	I_4^1	538.60	0.00	79.77	0.00
	I_5^1	844.85	0.00	366.48	0.00
	I_6^1	1299.71	0.00	177.05	0.00
	Avg	**461.89**	**0.00**	**177.05**	**0.00**
I^2	I_1^2	6.61	0.00	6.55	0.00
	I_2^2	41.02	0.00	15.96	0.00
	I_3^2	60.53	0.00	41.77	0.00
	I_4^2	450.02	0.00	55.02	0.00
	I_5^2	705.90	0.00	322.07	0.00
	I_6^2	1085.95	0.00	495.47	0.00
	Avg	**391.67**	**0.00**	**155.57**	**0.00**

The results confirm the following. First, for real problems that are more customized than instances from the literature, the CPLEX parameter tuning plays a significant role in reducing execution time and finding better solutions. Second, the linearization of the power stability constraints significantly impacts the solving process. While the classic linearization increases the number of binary variables and big-M constraints, making the problem quite complex, the new linearization does not add any binary variable or big-M constraints. Thus, it makes the solving significantly quicker. Third, the matheuristic partially and smartly decomposes the problem, thus allowing another gain in the execution time. All these features combined allow tackling almost all instances optimally in less than 10 minutes.

5.3 Managerial insights

From a system standpoint, the quick optimization (gained from the shift to less than 10 minutes) boosts HQ planning capabilities and supports decision-making by stakeholders. In particular, the short execution time allows for running several what-if scenarios and recovering quickly after perturbations and disruptions.

As a result, the optimizer becomes an efficient decision-making tool to check, control, simulate, and re-optimize electricity production. The mathematical optimization solution initiated the establishment of new management rules in the company, thus enhancing operations planning. To sum up, quickly solving the TSCUC supports HQ in handling the future more resiliently, allowing it to manage its water resources more efficiently.

The designed approach is generic and scalable to real world large-scale optimization problems. This approach consists of mathematically formulating the problem (modeling), conducting an exploratory analysis (diagnostic) of complexity sources, and then developing a solution methodology (prescriptions) that tackles the problem effectively.

6   Conclusion

While addressing HQ's complex UC problem efficiently, this work offers general insights that apply to other large-scale challenges. Following the presentation of the TSCUC model, the exploratory analysis reveals that the binary variables governing the groups' activities and the transient stability constraints are the primary sources of the problem's complexity. We design the $\mathcal{RSFC}$ matheuristic and put into practice a workable version, for which time serves as the decomposition criterion. Compared to the default CPLEX, the latter is highly beneficial in enhancing the caliber and speed of the problem-resolution procedure. From an industrial standpoint, it is possible to obtain good (gap $\leq 2\%$) practical solutions within less than 10 minutes. Moreover, the solution provided enhances HQ's planning capabilities and supports decision-making within the company.

References

Anjos, M., Conejo, A., and Publishers, N. (2017). Unit Commitment in Electric Energy Systems. Now Publishers.

Besner, A., Massé, A. B., Bani, A., Morabit, M., Charest, L., Ialongo, D., Couture-Gagnon, S., and Fournier, J. (2024). Generation planning and operation under power stability constraints: A Hydro-Quebec use case.

Birattari, M. and Kacprzyk, J. (2009). Tuning metaheuristics: a machine learning perspective, volume 197. Springer.

Borghetti, A., D'Ambrosio, C., Lodi, A., and Martello, S. (2008). An milp approach for short-term hydro scheduling and unit commitment with head-dependent reservoir. IEEE Transactions on power systems, 23(3):1115–1124.

Burks, T. M. and Sakallah, K. (1993). Min-max linear programming and the timing analysis of digital circuits. In Proceedings of 1993 International Conference on Computer Aided Design (ICCAD), pages 152–155. IEEE.

Er Raqabi, E. M., Himmich, I., El Hachemi, N., El Hallaoui, I., and Soumis, F. (2023). Incremental LNS Framework for Integrated Production, Inventory, and Vessel Scheduling: Application to a Global Supply Chain. Omega, 116:102821.

Ginocchio, R. and Viollet, P.-L. (2012). L'énergie hydraulique. Lavoisier.

Gleixner, A., Hendel, G., Gamrath, G., Achterberg, T., Bastubbe, M., Berthold, T., Christophel, P., Jarck, K., Koch, T., Linderoth, J., et al. (2021). MIPLIB 2017: data-driven compilation of the 6th mixed-integer programming library. Mathematical Programming Computation, 13(3):443–490.

Himmich, I., Er Raqabi, E. M., El Hachemi, N., El Hallaoui, I., Metrane, A., and Soumis, F. (2023). MPILS: An Automatic Tuner for MILP Solvers. Computers & Operations Research, 159:106344.

Huang, J. A., Loud, L., Vanier, G., Lambert, B., and Guillon, S. (2012). Experiences and challenges in contingency analysis at Hydro-Quebec. In 2012 IEEE Power and Energy Society General Meeting, pages 1–9.

Hui, H. (2013). Reliability unit commitment in ERCOT nodal market. PhD thesis, University of Texas at Arlington.

IBM ILOG CPLEX (2009). V12.1: User's Manual for CPLEX. International Business Machines Corporation, 46(53):157.

López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L. P., Birattari, M., and Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. Operations Research Perspectives, 3:43–58.

Montero, L., Bello, A., and Reneses, J. (2022). A review on the unit commitment problem: Approaches, techniques, and resolution methods. Energies, 15(4):1296.

Musto, M. (2020). Day Ahead Network Constrained Unit Commitment Performance. Presented as the 11th Annual FERC Software Conference, Washington, DC.