SPLiT: **T**ime-of-Use **P**ump **S**cheduling with Switch **Li**mit and/or Penalty

Yuzhen Feng

Abstract

We study a continuous-time pump scheduling problem for a flow transmission task. A finite table of empirical operating points of pump combinations is given, each point specifying a flow rate and power consumption. Electricity prices follow a time-of-use tariff, and combination changes are penalized or limited by per-shift switch caps. We prove a structural theorem: after partitioning time into maximal intervals on which both the tariff and the shift label are the same, called atoms, there exists an optimal schedule in which all switches occur at atom boundaries except possibly one internal switch in one atom. If the selected optimal schedule terminates strictly inside an atom, then no internal atom switch is needed at all. This structure yields an exact atom-level mixed integer linear program whose size depends on the number of atoms rather than on a fine time grid.

1 Introduction

Pumping systems are central to continuous-process operations such as crude-oil gathering, pipeline transmission, refinery transfers, and water distribution. This paper studies a practically common flow transmission setting. Table 1 gives a compact illustrative data set. A task starts at time zero and must deliver a prescribed volume. Exactly one pump combination k is active at any time, and idling is not allowed. Combination k has measured flow v_k and power q_k. Additivity across pumps is not assumed: a combination is treated as its own empirical operating point because hydraulic interactions can make the combined (v_k, q_k) pair different from the sum of the individual operating points. Operators must deliver the volume while facing time-varying electricity prices $c(t)$ and limits and/or penalties on how often the active pump combination may be changed. Time-of-Use (ToU) tariffs make the pump scheduling economically important: even when one pump combination has the lowest specific energy, it may be profitable to run a higher-throughput combination in low-price hours to avoid expensive pumping later.

The literature on pump scheduling is extensive. Early optimization models already considered electricity tariffs and discrete operational decisions. Little and McCrodden (1989) used a mixed integer linear program (MILP) for raw-water pumping under time-varying electricity costs. Jowitt and Germanopoulos (1992) developed a linear program for water-supply network pump scheduling. Lansey and Awumah (1994) explicitly included pump-switching considerations. McCormick and Powell (2003) studied maximum-demand electricity charges. Later work on water distribution networks addresses much richer hydraulic physics, typically leading to nonconvex mixed integer nonlinear programs. Representative approaches include Lagrangian decomposition (Ghaddar et al.,

Table 1: Illustrative ToU tariff and pump combinations.

(a) ToU Tariff

Start (h)	End (h)	Price $c(t)$
0	8	0.21
8	10	0.55
10	14	0.94
14	19	0.55
19	24	0.94

(b) Pump Combinations

Combination $k \in \mathcal{K}$	Flow v_k	Power q_k
1#	1055	1102
2#	1113	1120
1#&2#	1880	2352

2015), branch-and-bound methods (Costa et al., 2016; Menke et al., 2016b; Bonvin et al., 2021), convexification (Bonvin et al., 2017), linearization (Shao et al., 2024), and polyhedral relaxations (Tasseff et al., 2024). Studies on demand management further show that pumping systems can shift load in response to ToU and other market mechanisms (van Staden et al., 2011; Menke et al., 2016a). Another related stream studies pipeline and transmission scheduling with hybrid time representations, where event times induced by batches, station arrivals, or tariff changes are often more natural than a uniform time grid. Zhou et al. (2019) used a hybrid-time MILP to capture batch-induced hydraulic loss changes in multiproduct pipeline pump scheduling. Xu et al. (2021) combined breakpoints of electricity prices with event points of batch arrivals in a decomposition approach for straight multiproduct pipelines.

Our main contribution is a structural theorem for ToU pump scheduling. Partitioning the horizon into maximal intervals with constant tariff and shift, which we call atoms, we show that there exists an optimal schedule with at most one internal switch in at most one atom, with all remaining switches placed at atom boundaries. This result provides an exact continuous-time alternative to fine time discretization. Building on the theorem, we formulate an atom-level MILP that explicitly handles the unique possible internal switch. The remainder of this paper is organized as follows. Section 2 formulates the problem. Section 3 proves the structural theorem. Section 4 presents the exact atom-level MILP implied by the theorem. Section 5 reports numerical experiments comparing the atom-level formulation with a time-indexed baseline. Section 6 concludes.

2 Problem Formulation

Let $\mathcal{K}$ be a finite set of pump combinations. Each $k \in \mathcal{K}$ has flow $v_k > 0$ and power $q_k > 0$. The ToU price $c(t) > 0$ and shift label $s(t)$ are right-continuous step functions. Since the task starts at time 0 and all flows are positive, every feasible schedule completes no later than $H := V/\min_{k\in\mathcal{K}} v_k$. We include H and all discontinuity times of c or s in the event set. Let $0 = t_0 < t_1 < \cdots < t_A = H$ be the resulting ordered event times, and define atom $a \in \mathcal{A} := \{0, \ldots, A-1\}$ as the interval $[t_a, t_{a+1})$. Its length is $l_a := t_{a+1} - t_a$, price is $c_a := c(t_a)$, and shift label is $s_a := s(t_a)$. By construction, both c and s are constant on each atom. The shift periods are denoted by $\mathcal{P} := \{0, \ldots, P-1\}$. Period p is a maximal interval on which the shift label is constant, and it has switch cap W_p. For readability, one may set $W_p = W$, although the formulation allows period-dependent caps. Let $\mathcal{A}_p \subseteq \mathcal{A}$ be the atoms contained in shift period p, let $a_p := \min \mathcal{A}_p$ be the first atom of period p, and let $\mathcal{B}_p := \mathcal{A}_p \setminus \{a_p\}$. Thus $\mathcal{B}_p$ indexes atom boundaries strictly inside period p.

A schedule is a right-continuous, piecewise-constant map $k : [0, T^*) \to \mathcal{K}$, where $T^* \in (0, H]$. A switch occurs at $t \in (0, T^*)$ if $k(t^-) \neq k(t^+)$. A switch strictly inside a shift period is attributed to that period. A switch exactly at a shift boundary may be attributed to either adjacent period; let η denote such an attribution rule. For a schedule k and attribution η, let $w_p(k, \eta)$ be the number of switches attributed to period p. The initial choice at $t = 0$ is not counted as a switch. The schedule must deliver exactly V, meaning $\int_0^{T^*} v_{k(t)} \mathrm{d}t = V$. The objective is to minimize the energy cost $\int_0^{T^*} c(t) q_{k(t)} \mathrm{d}t$, subject to $w_p(k, \eta) \leq W_p$ for all $p \in \mathcal{P}$. The structural analysis also applies when switch limits are replaced by nonnegative additive switch penalties. Specifically, for penalties $\lambda_p \geq 0$, one may minimize

$$\int_0^{T^*} c(t) q_{k(t)} \mathrm{d}t + \sum_{p \in \mathcal{P}} \lambda_p w_p(k, \eta).$$

3 Structural Theorem

For any feasible schedule, write the blocks of pump combinations with positive durations inside atom a as the ordered list $[k_{a1}, \ldots, k_{aR_a}]$, with corresponding durations $[d_{a1}, \ldots, d_{aR_a}]$. The *block skeleton* of a schedule is the collection of these ordered combination lists. In the argument below, the block skeleton is fixed, whereas their durations are reoptimized and may become zero. The next lemma shows that, after this duration reoptimization, zero-duration blocks can be deleted without increasing switch counts.

Lemma 1. *Fix the block skeleton of a feasible schedule. Delete all zero-duration blocks and merge adjacent positive-duration blocks with the same combination. The resulting schedule has the same delivered volume and energy cost as the original schedule, and its switch count in every shift period is no larger than that of the original skeleton.*

Proof. Volume and cost are unchanged because zero-duration blocks contribute neither volume nor cost, and merging adjacent blocks with the same combination preserves their total duration. Consider switches. Deleting a zero-duration block can only remove switches or replace a chain of skeleton switches by a single switch between the nearest positive-duration blocks on the two sides. If the replacement switch lies strictly inside a shift period, it is charged to the same period that contained the removed chain, so the count in that period cannot increase. If the replacement switch lies at a shift boundary, it can be attributed to the side from which the removed switch originated. Repeating this argument for all zero-duration blocks gives a compressed schedule whose per-period switch counts are componentwise no larger than those of the skeleton. □

We now formulate the duration reoptimization for a fixed skeleton as a linear program (LP). Suppose the schedule has full atoms $\mathcal{F}$ and possibly one terminal atom b that is

only partially used. The fixed-skeleton LP is

$$\min \quad \sum_{a\in\mathcal{F}\cup\{b\}} \sum_{r=1}^{R_a} c_a q_{k_{ar}} d_{ar}, \tag{1}$$

$$\text{s.t.} \quad \sum_{a\in\mathcal{F}\cup\{b\}} \sum_{r=1}^{R_a} v_{k_{ar}} d_{ar} = V, \tag{2}$$

$$\sum_{r=1}^{R_a} d_{ar} = l_a, \quad \forall a \in \mathcal{F}, \tag{3}$$

$$\sum_{r=1}^{R_a} d_{br} \le l_b, \tag{4}$$

$$d_{ar} \ge 0, \quad \forall r = 1, \dots, R_a,\ a \in \mathcal{F}\cup\{b\}. \tag{5}$$

If the original schedule ends at an atom boundary, the inequality (4) is absent and all used atoms are in $\mathcal{F}$.

Lemma 2. *For the block skeleton induced by any optimal schedule, there exists an optimal duration assignment such that, if f atoms are fully used, then at most $f+1$ block durations are positive.*

Proof. The feasible region of the fixed-skeleton LP is a nonempty polytope because it contains the durations of the original optimal schedule. Since the objective is linear, an optimal extreme point exists. The LP has one volume equality (2) and f full-atom length equalities (3). Therefore the rank of the equality system is at most $f+1$, and an extreme point has at most $f+1$ positive nonnegative variables. □

Theorem 1. *There exists an optimal schedule satisfying one of the following two alternatives:*

(i) The schedule terminates strictly inside an atom. Then every fully used atom and the terminal partial atom use a single pump combination; hence all switches occur at atom boundaries.

(ii) The schedule terminates at an atom boundary. Then at most one fully used atom has two positive-duration blocks, and all other atoms use a single combination. Equivalently, all switches occur at atom boundaries except possibly one internal switch in one atom.

Proof. Take an optimal feasible schedule and fix its block skeleton. Solve the corresponding fixed-skeleton LP and choose an optimal extreme point. By Lemma 1, after deleting zero-duration blocks and merging equal adjacent labels, the resulting schedule is feasible for the original switch caps and has the same energy cost. First suppose the compressed schedule terminates strictly inside an atom b. Let f be the number of full atoms before b. Lemma 2 gives at most $f+1$ positive blocks in total, so every full atom has exactly one positive block and the terminal atom has exactly one positive block; no atom contains an internal switch. Now suppose the compressed schedule terminates at an atom boundary. Lemma 2 gives at most $f+1$ positive blocks. Hence at most one atom can have two positive blocks, and no atom can have three or more. Such an atom has at most one internal switch, while all other atoms have none. Therefore all other switches occur at atom boundaries. □

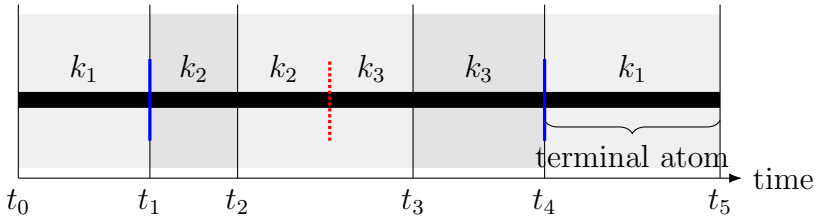

Figure 1: Theorem 1(ii) restricts switching times to atom boundaries (blue solid), except possibly one internal split (red dotted).

4 Continuous-Time Atom-Level MILP

Theorem 1 implies that it is sufficient to optimize over schedules with at most one internal split. The following MILP exploits this structure in continuous time, using atom-level variables rather than a fine time discretization.

For each atom a and combination k, binary variables y_a and $\bar{y}_a$ indicate whether atom a is reached and whether it is the terminal atom. The variable $\rho_a \in [0,1]$ is the used fraction of the terminal atom. The binary z_a indicates whether atom a is internally split. The binaries x^l_{ak} and x^r_{ak} select the left-block and right-block combinations in atom a. The continuous variables $\theta^l_{ak} \geq 0$ and $\theta^r_{ak} \geq 0$ are their run times. For atom boundary t_a, the binary σ_a indicates a boundary switch, with $\sigma_0 = 0$. For each shift boundary p, $\eta_p = 1$ attributes the boundary switch at t_{a_p} to the left shift $p-1$, while $\eta_p = 0$ attributes it to the right shift p. The auxiliary variable α_p linearizes $\eta_p \sigma_{a_p}$. The objective is

$$\min \quad \sum_{a \in \mathcal{A}} c_a \sum_{k \in \mathcal{K}} q_k (\theta^l_{ak} + \theta^r_{ak}). \tag{6}$$

The contiguity and terminality constraints are

$$y_0 = 1, \quad y_a \leq y_{a-1}, \quad y_a + \bar{y}_{a-1} \leq 1, \quad \forall a \in \mathcal{A} \setminus \{0\},$$
$$\sum_{a \in \mathcal{A}} \bar{y}_a = 1, \quad \rho_a \leq \bar{y}_a \leq y_a, \quad \forall a \in \mathcal{A}.$$

The split and endpoint combination-selection constraints are

$$\sum_{a \in \mathcal{A}} z_a \leq 1, \quad z_a \leq y_a, \quad \forall a \in \mathcal{A},$$
$$\sum_{k \in \mathcal{K}} x^l_{ak} = y_a, \quad \sum_{k \in \mathcal{K}} x^r_{ak} = y_a, \quad \forall a \in \mathcal{A},$$
$$x^r_{ak} - x^l_{ak} \leq z_a, \quad \forall a \in \mathcal{A},\ k \in \mathcal{K}.$$

The last inequality forces $x^l_{ak} = x^r_{ak}$ when $z_a = 0$, while allowing different endpoint combinations only when $z_a = 1$. Run times are linked by

$$\sum_{k \in \mathcal{K}} (\theta^l_{ak} + \theta^r_{ak}) = l_a (y_a - \bar{y}_a) + l_a \rho_a, \quad \forall a \in \mathcal{A},$$
$$0 \leq \theta^l_{ak} \leq l_a x^l_{ak}, \quad 0 \leq \theta^r_{ak} \leq l_a x^r_{ak}, \quad \forall a \in \mathcal{A},\ k \in \mathcal{K},$$
$$\sum_{k \in \mathcal{K}} \theta^r_{ak} \leq l_a z_a, \quad \forall a \in \mathcal{A}.$$

The exact delivery requirement is

$$\sum_{a\in\mathcal{A}}\sum_{k\in\mathcal{K}} v_k(\theta^l_{ak}+\theta^r_{ak}) = V.$$

Switches are detected by

$$\sigma_0 = 0, \quad \sigma_a \le y_a, \quad \sigma_a \ge x^l_{ak} - x^r_{a-1,k}, \quad \forall a \in \mathcal{A}\setminus\{0\},\ k\in\mathcal{K}.$$

The McCormick envelopes for $\alpha_p = \eta_p\sigma_{a_p}$, $p \in \mathcal{P}\setminus\{0\}$, are

$$\alpha_p \le \eta_p, \quad \alpha_p \le \sigma_{a_p}, \quad \alpha_p \ge \eta_p + \sigma_{a_p} - 1, \quad \alpha_p \ge 0.$$

Finally, the shift switch caps are

$$\sum_{a\in\mathcal{A}_p} z_a + \sum_{a\in\mathcal{B}_p} \sigma_a + \mathbf{1}_{\{p\ge 1\}}(\sigma_{a_p} - \alpha_p) + \mathbf{1}_{\{p\le P-2\}}\alpha_{p+1} \le W_p, \quad \forall p\in\mathcal{P}.$$

The first term counts internal splits in period p, the second counts atom-boundary switches strictly inside the shift period, the third counts the incoming shift-boundary switch if it is not attributed left, and the fourth counts the outgoing shift-boundary switch if it is attributed left. The domains are $y_a, \bar{y}_a, z_a, \sigma_a \in \{0,1\}$, $x^l_{ak}, x^r_{ak} \in \{0,1\}$, $\eta_p \in \{0,1\}$, $\rho_a \in [0,1]$, $\alpha_p \in [0,1]$, and $\theta^l_{ak}, \theta^r_{ak} \ge 0$.

5 Numerical Experiments

We evaluate the exact atom-level MILP against a time-indexed MILP baseline formulated in Appendix A. The instances are synthetic and are grouped into four scale classes, S1–S4, as summarized in Table 2. Each day uses the same ToU tariff as in Table 1. The shift calendar has three 8-hour shifts, 00:00–08:00, 08:00–16:00, and 16:00–24:00. Tariff and shift boundaries jointly define six atoms per day. The time-indexed MILP uses slot lengths $\Delta \in \{60, 30, 15, 5\}$ minutes for S1–S2 and $\Delta \in \{60, 15\}$ minutes for S3–S4. All models were implemented in Python 3.11.5 and solved with Gurobi Optimizer 11.0.0. Experiments were run on a Mac mini with an Apple M2 chip, 8 CPU cores, and 24 GB RAM. We set the random seed to 1. The time limit was 600 seconds for the atom-level MILP, 600 seconds for the time-indexed MILP on S1–S2, and 900 seconds for the time-indexed MILP on S3–S4. All other solver parameters were left at their default values.

Table 2: Instance classes.

Class	#Cases	#Pumps	K	Days	W	V (10^3·m^3)
S1	20	3	7	2–5	0–3	39.2–142.5
S2	25	4	15	7–14	1–3	126.0–431.8
S3	25	5	31	21–45	1–4	350.8–1262.3
S4	30	6	63	45–90	1–4	765.4–2595.8

Table 3 summarizes the exact atom-level MILP. All 100 atom-level runs are solved to optimality. The model remains small because its binary variables scale with the number of atoms and pump combinations, not with minute-level time slots. Even in S4, where instances have six pumps, $K = 63$ combinations, and up to 510 atoms, the atom-level

MILP solves in at most 38.94 seconds. Table 4 compares the time-indexed MILP with the atom-level formulation. The column "Opt./TL" gives the number of instances solved to optimality and the number hitting the time limit. The objective gap is computed only on time-indexed runs solved to optimality. Thus the reported gap measures discretization loss rather than the quality of a time-limit incumbent. The binary ratio reports the average ratio between the number of time-indexed binary variables and the number of atom-level binary variables on the same instances.

Table 3: Atom-level MILP performance.

Class	#Binary vars. mean	Time med.	Time max
S1	320	0.014	0.056
S2	2,040	0.259	1.685
S3	11,141	3.125	6.212
S4	41,860	12.588	38.936

Table 4: Time-indexed MILP performance.

Class	Δ	Opt./TL	Time med.	Binary ratio	Max gap (%)
S1	60	20/0	0.06	2.2	0.2859
S1	30	20/0	0.24	4.3	0.0957
S1	15	20/0	1.02	8.5	0.0808
S1	5	19/1	3.92	25.5	0.0172
S2	60	24/1	2.98	2.1	0.0615
S2	30	23/2	4.19	4.2	0.0340
S2	15	25/0	7.50	8.3	0.0130
S2	5	25/0	33.99	25.0	0.0096
S3	60	21/4	11.22	2.1	0.0381
S3	15	25/0	45.56	8.2	0.0096
S4	60	28/2	45.79	2.0	0.0147
S4	15	21/9	309.65	8.1	0.0083

The results support three observations. First, the atom-level MILP is exact and robust across all tested scales. Second, finer time discretization does reduce the objective loss of the baseline, but at the expected cost of many more binary variables. For example, in S2 the average binary ratio rises from 2.1 at $\Delta = 60$ minutes to 25.0 at $\Delta = 5$ minutes. Third, the large-scale cases expose the computational risk of time-indexed formulations. In S4, the 15-minute baseline hits the 900-second time limit in 9 out of 30 instances, whereas the atom-level MILP solves all 30 instances in under 39 seconds. These experiments therefore confirm the value of Theorem 1. By optimizing directly over atoms and allowing the single possible internal split, the model preserves continuous-time exactness while avoiding the growth of binary variables induced by uniform time slots.

6 Conclusion

We studied a continuous-time pump scheduling problem with ToU electricity prices, empirical flow–power operating points, and switch limits and/or penalties. The main result is structural: after partitioning time into atoms on which both the tariff and shift label are constant, there exists an optimal schedule with all switches at atom boundaries except possibly one internal split. This result leads directly to an exact atom-level MILP that avoids uniform time discretization while preserving continuous-time optimality. Computational results on synthetic instances show that the atom-level formulation solves all tested cases to optimality and remains smaller and faster than a time-indexed MILP baseline, especially as the slot length is refined or the horizon grows. A future direction is to extend the model to network settings with system-wide hydraulic interactions.

Disclosure

During the preparation of this work, the author used ChatGPT for language polishing. The author reviewed and edited the output as needed and takes full responsibility for the final manuscript.

References

Bonvin, G., Demassey, S., Le Pape, C., Maïzi, N., Mazauric, V., Samperio, A., 2017. A convex mathematical program for pump scheduling in a class of branched water networks. Applied Energy 185, 1702–1711.

Bonvin, G., Demassey, S., Lodi, A., 2021. Pump scheduling in drinking water distribution networks with an LP/NLP-based branch and bound. Optimization and Engineering 22, 1275–1313.

Costa, L.H.M., De Athayde Prata, B., Ramos, H.M., De Castro, M.A.H., 2016. A branch-and-bound algorithm for optimal pump scheduling in water distribution networks. Water Resources Management 30, 1037–1052.

Ghaddar, B., Naoum-Sawaya, J., Kishimoto, A., Taheri, N., Eck, B., 2015. A Lagrangian decomposition approach for the pump scheduling problem in water networks. European Journal of Operational Research 241, 490–501.

Jowitt, P.W., Germanopoulos, G., 1992. Optimal pump scheduling in water-supply networks. Journal of Water Resources Planning and Management 118, 406–422.

Lansey, K.E., Awumah, K., 1994. Optimal pump operations considering pump switches. Journal of Water Resources Planning and Management 120, 17–35.

Little, K.W., McCrodden, B.J., 1989. Minimization of raw water pumping costs using MILP. Journal of Water Resources Planning and Management 115, 511–522.

McCormick, G., Powell, R.S., 2003. Optimal pump scheduling in water supply systems with maximum demand charges. Journal of Water Resources Planning and Management 129, 372–379.

Menke, R., Abraham, E., Parpas, P., Stoianov, I., 2016a. Demonstrating demand response from water distribution system through pump scheduling. Applied Energy 170, 377–387.

Menke, R., Abraham, E., Parpas, P., Stoianov, I., 2016b. Exploring optimal pump scheduling in water distribution networks with branch and bound methods. Water Resources Management 30, 5333–5349.

Shao, Y., Zhou, X., Yu, T., Zhang, T., Chu, S., 2024. Pump scheduling optimization in water distribution system based on mixed integer linear programming. European Journal of Operational Research 313, 1140–1151.

van Staden, A.J., Zhang, J., Xia, X., 2011. A model predictive control strategy for load shifting in a water pumping scheme with maximum demand charges. Applied Energy 88, 4785–4794.

Tasseff, B., Bent, R., Coffrin, C., Barrows, C., Sigler, D., Stickel, J., Zamzam, A.S., Liu, Y., Van Hentenryck, P., 2024. Polyhedral relaxations for optimal pump scheduling of potable water distribution networks. INFORMS Journal on Computing 36, 1040–1063.

Xu, N., Liao, Q., Li, Z., Liang, Y., Qiu, R., Zhang, H., 2021. An efficient decomposition approach for the low-energy scheduling of a straight multiproduct pipeline. Chemical Engineering Research and Design 165, 341–360.

Zhou, X., Zhang, H., Qiu, R., Liang, Y., Wu, G., Xiang, C., Yan, X., 2019. A hybrid time MILP model for the pump scheduling of multi-product pipelines based on the rigorous description of the pipeline hydraulic loss changes. Computers & Chemical Engineering 121, 174–199.

A Time-Indexed MILP Baseline

This appendix gives the time-indexed MILP as a baseline. Unlike the atom-level formulation, this model restricts combination switches to a uniform time grid. It is therefore not an exact continuous-time formulation, although we allow a fractional terminal slot to avoid overdelivery.

Fix a slot length $\Delta > 0$, and let $\mathcal{T} := \{0, \dots, T-1\}$, where $T := \lceil H/\Delta \rceil$. Slot t represents the interval $[t\Delta, (t+1)\Delta)$. Let $c_t := c(t\Delta)$, and let $p(t)$ be the shift period containing the start of slot t. Let $\mathcal{T}_p := \{t \in \mathcal{T} \mid p(t) = p\}$, let $t_p := \min \mathcal{T}_p$, and let $\mathcal{D}_p := \mathcal{T}_p \setminus \{t_p\}$. We assume that the grid aligns with all tariff and shift boundaries. The decision variables are as follows. Binary variable x_{tk} equals one if combination k is selected in slot t. Binary variable y_t indicates whether slot t is reached, binary variable $\bar{y}_t$ marks the terminal slot, and $\rho_t \in [0,1]$ is the fraction of the terminal slot that is actually used. Continuous variable $\tau_{tk} \geq 0$ is the runtime of combination k in slot t. Binary variable σ_t indicates a switch at the beginning of slot t, with $\sigma_0 = 0$. As in the atom-level formulation, binary variable η_p attributes a switch at shift boundary $t_p\Delta$ to the left period when $\eta_p = 1$, and α_p linearizes $\eta_p \sigma_{t_p}$. The objective is

$$\min \quad \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} c_t q_k \tau_{tk}.$$

The contiguity and terminality constraints are

$$y_0 = 1, \quad y_t \le y_{t-1}, \quad y_t + \bar{y}_{t-1} \le 1, \quad \forall t \in \mathcal{T} \setminus \{0\},$$
$$\sum_{t \in \mathcal{T}} \bar{y}_t = 1, \quad 0 \le \rho_t \le \bar{y}_t \le y_t, \quad \forall t \in \mathcal{T}.$$

The combination-selection and runtime constraints are

$$\sum_{k \in \mathcal{K}} x_{tk} = y_t, \quad \forall t \in \mathcal{T},$$
$$\sum_{k \in \mathcal{K}} \tau_{tk} = \Delta(y_t - \bar{y}_t) + \Delta\rho_t, \quad \forall t \in \mathcal{T},$$
$$0 \le \tau_{tk} \le \Delta x_{tk}, \quad \forall t \in \mathcal{T},\ k \in \mathcal{K}.$$

The exact delivery requirement is

$$\sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} v_k \tau_{tk} = V.$$

Switches are detected by

$$\sigma_0 = 0, \quad \sigma_t \le y_t, \quad \sigma_t \ge x_{tk} - x_{t-1,k}, \quad \forall t \in \mathcal{T} \setminus \{0\},\ k \in \mathcal{K}.$$

The McCormick envelopes for $\alpha_p = \eta_p \sigma_{a_p}$, $p \in \mathcal{P} \setminus \{0\}$, are

$$\alpha_p \le \eta_p, \quad \alpha_p \le \sigma_{a_p}, \quad \alpha_p \ge \eta_p + \sigma_{a_p} - 1, \quad \alpha_p \ge 0.$$

Finally, the shift switch caps are

$$\sum_{t \in \mathcal{D}_p} \sigma_t + \mathbf{1}_{\{p \ge 1\}}(\sigma_{t_p} - \alpha_p) + \mathbf{1}_{\{p \le P-2\}} \alpha_{p+1} \le W_p, \quad p \in \mathcal{P}.$$

The domains are $y_t, \bar{y}_t, \sigma_t \in \{0,1\}$, $x_{tk} \in \{0,1\}$, $\eta_p \in \{0,1\}$, $\rho_t \in [0,1]$, $\alpha_p \in [0,1]$, and $\tau_{tk} \ge 0$. In this formulation, its binary variables scale with the number of slots T. As Δ decreases, it can become a closer approximation to continuous-time switching but grows rapidly in size.