Load Scheduling of Simple Temporal Networks Under Dynamic Resource Pricing

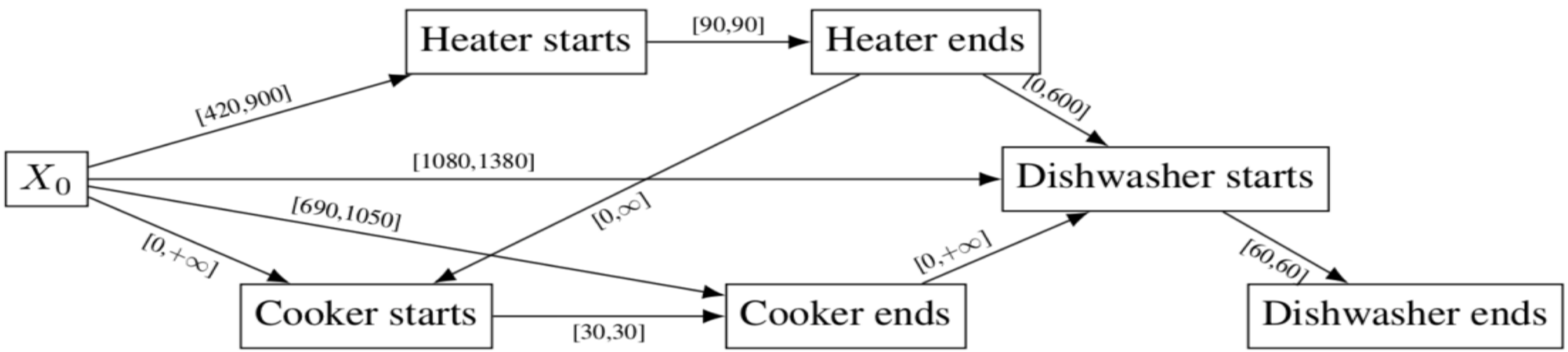
T. K. Satish Kumar, Zhi Wang, Anoop Kumar, Craig Rogers, Craig Knoblock

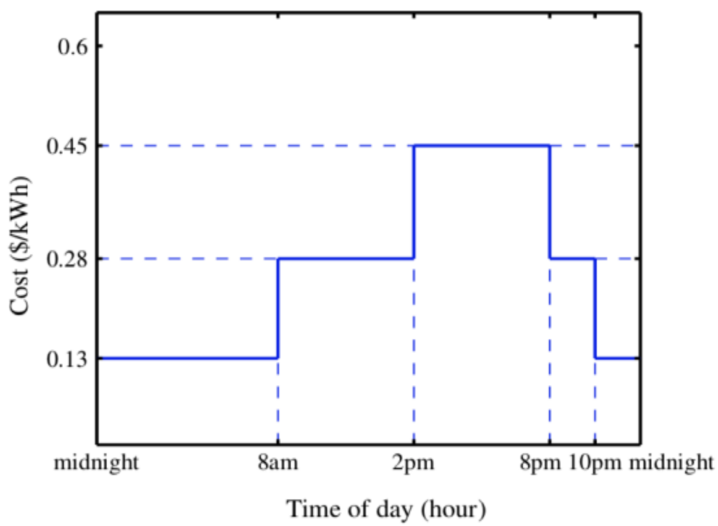
Information Sciences Institute, University of Southern California

Executive Summary in Keywords

- ***Simple Temporal Networks (STNs):*** temporal constraints between processes in scheduling problems.
- ***Resources:*** like electricity, consumed by processes.
- ***Dynamic Price:*** unit cost of electricity varies with time and total demand.
- ***Polynomial-time Algorithms:*** for cost minimization and optimal tradeoff against makespan in many important classes of such scheduling problems.

Example: Smart Home



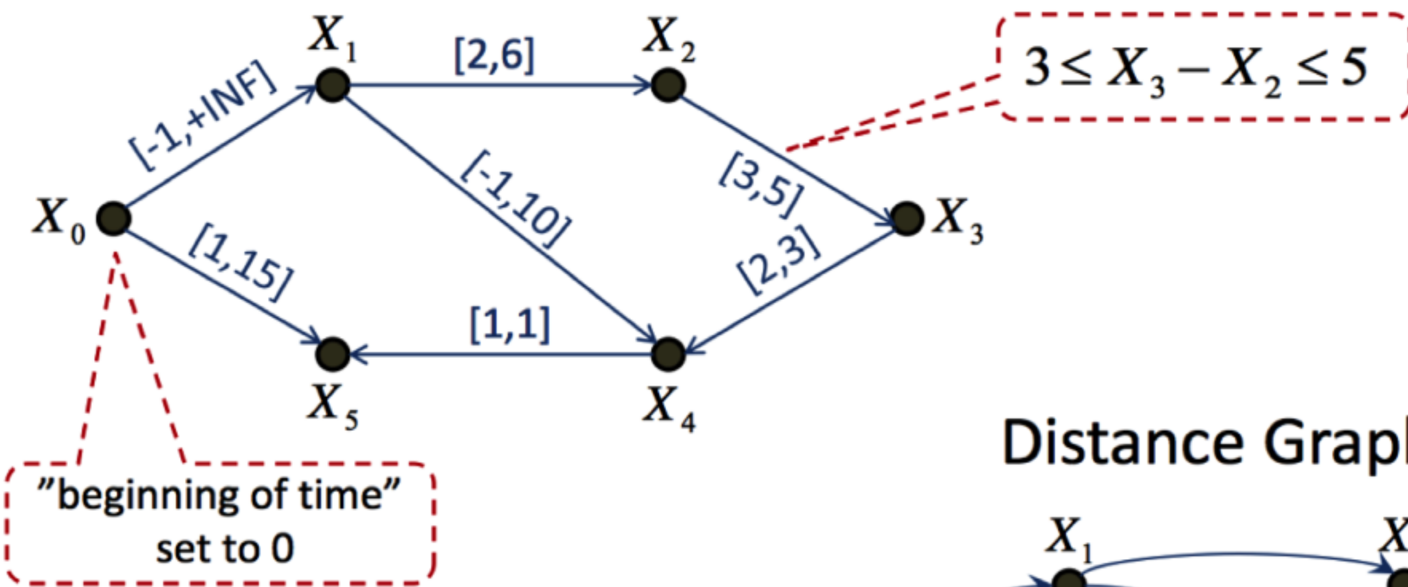


Find a consistent schedule of minimum total cost.

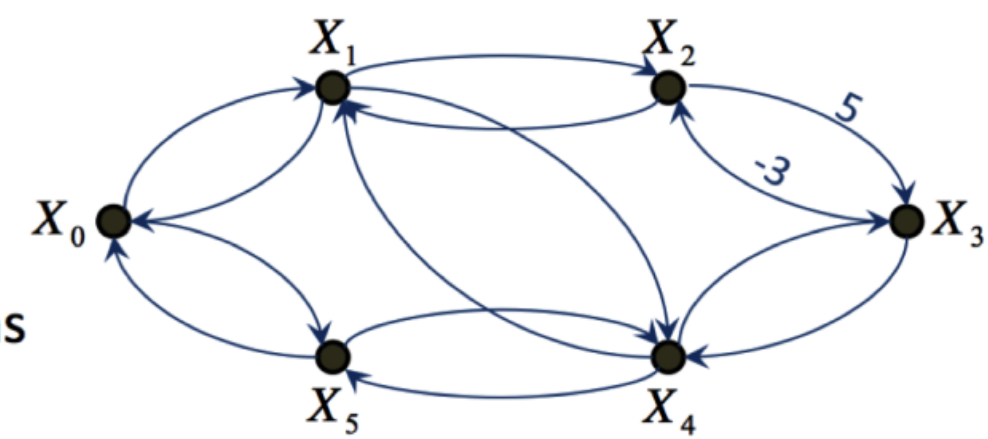
Two Models of Resource Consumption

- **Model A:** Each process P_i consumes electricity at the rate of w_i watts during execution.
- **Model B:** Each process P_i demands its entire energy requirement, that is, the total energy $W_i = w_i\, duration(P_i)$ at the beginning of its execution.
- Solving **Model B** is a little simpler. It also provides the critical combinatorial arguments useful for solving **Model A**.

Simple Temporal Networks/Problems

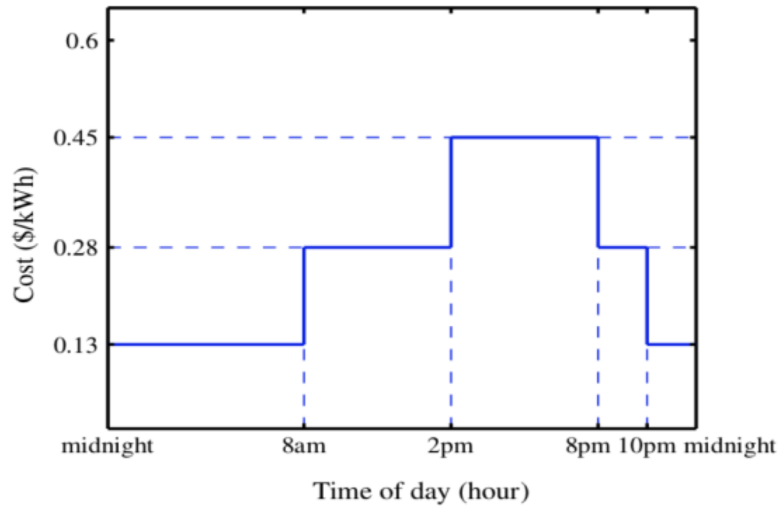


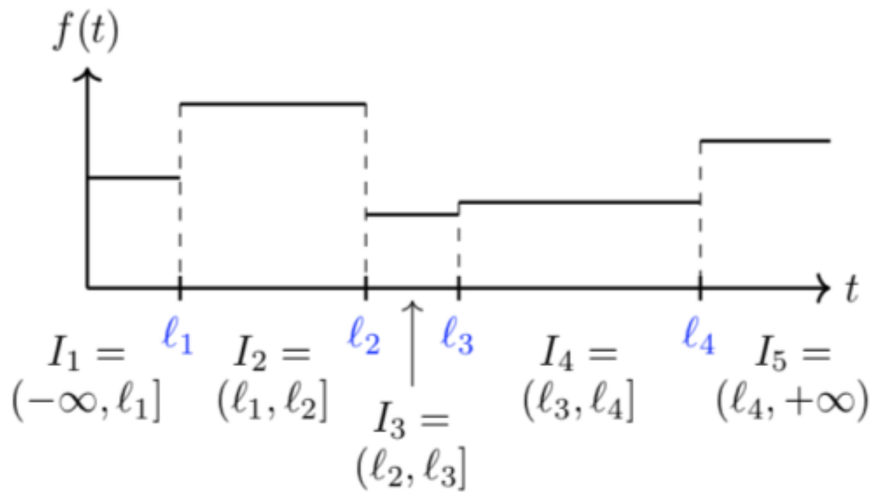
Distance Graph



STPs can be solved by shortest-path computations on their distance graphs

Core Combinatorial Problem





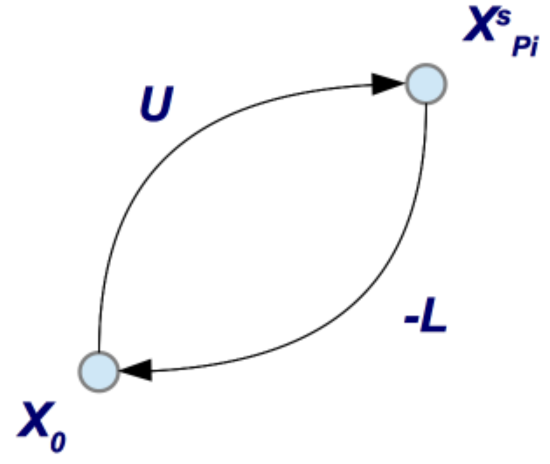
Each process P_i has to be ***activated*** in some interval I_j, that is, the starting point of P_i should be in I_j.

The cost for activating P_i in the interval I_j is $W_i f(I_j)$.

Find the best combination of intervals in which each process should be activated such that: (a) the schedule is consistent; and (b) the total cost is minimized.

Activating Process P_i in Interval I_j

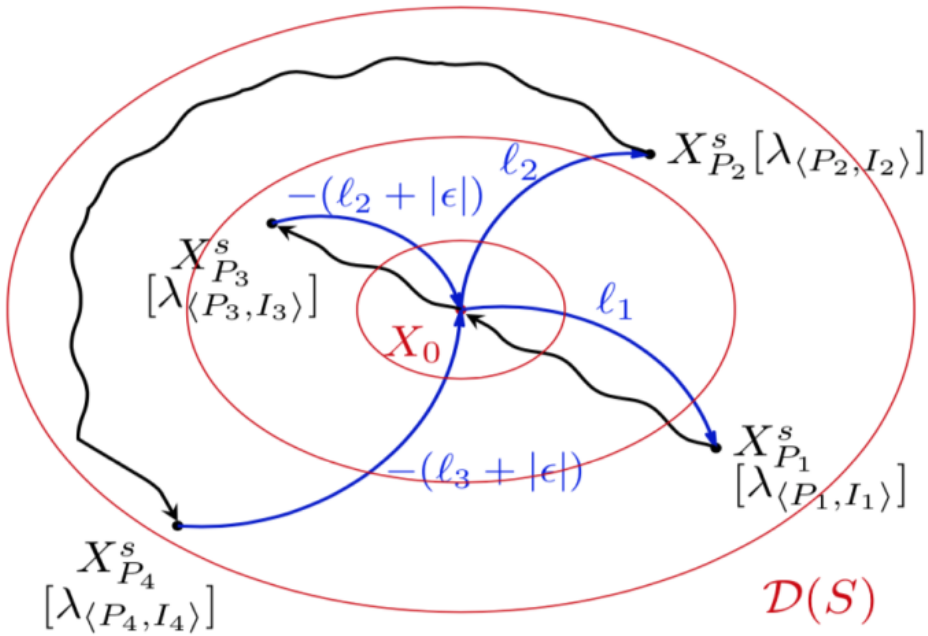
- The beginning point of P_i, i.e., X^s_{Pi}, should be scheduled after the left endpoint of I_j (say, L) and before the right endpoint of I_j (say, U).
- $X^s_{Pi} - X_0 \geq L$ and $X^s_{Pi} - X_0 \leq U$.



Conflicts and Minimal Conflicts

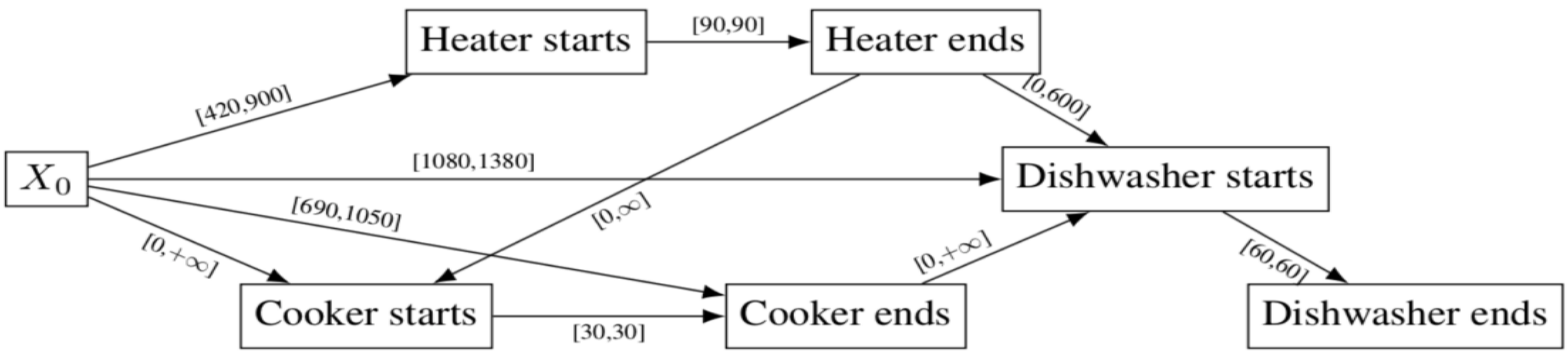
- A ***conflict*** is a set of activations $(P_1, I_{j1}), (P_2, I_{j2}) \ldots (P_K, I_{jK})$ that cannot be simultaneously achieved.
- A ***minimal conflict*** is a conflict no proper subset of which is also a conflict.
- A set of activations $(P_1, I_{j1}), (P_2, I_{j2}) \ldots (P_K, I_{jK})$ can be simultaneously achieved if and only if they do not contain a minimal conflict.

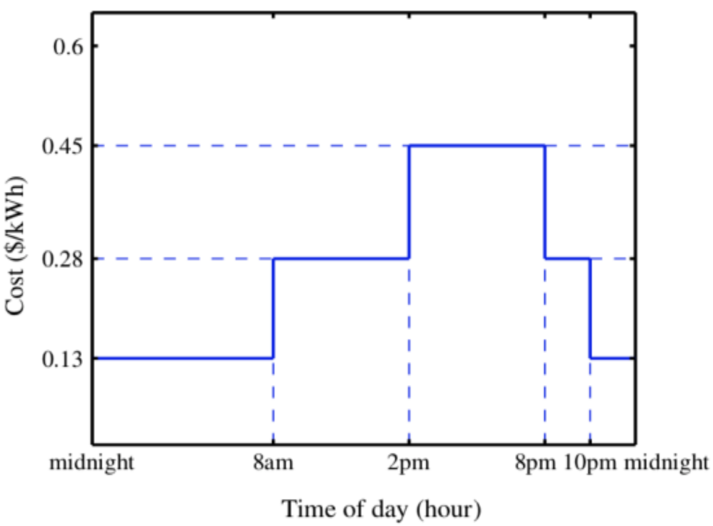
Bounded Minimal Conflicts



The size of a minimal conflict is ≤ 2.

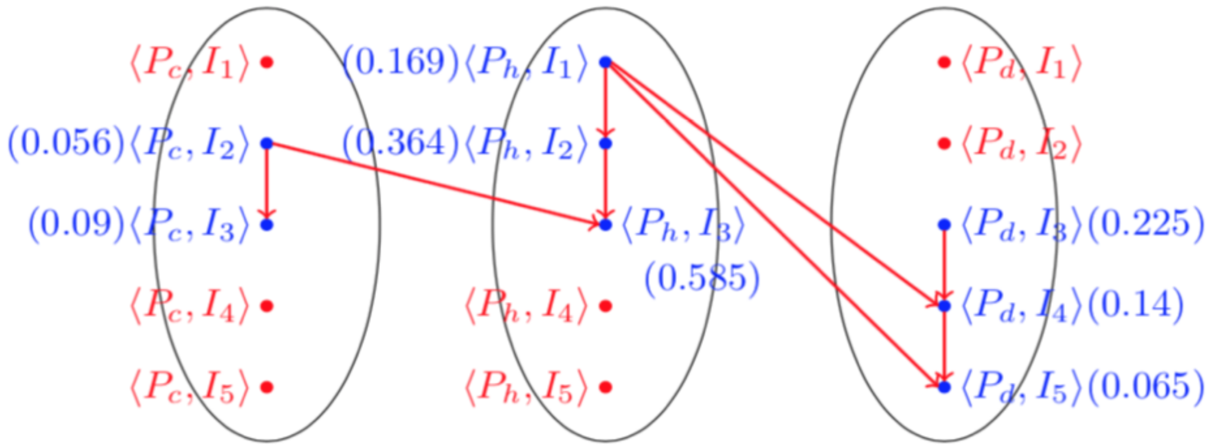
Example: Smart Home





Find a consistent schedule of minimum total cost.

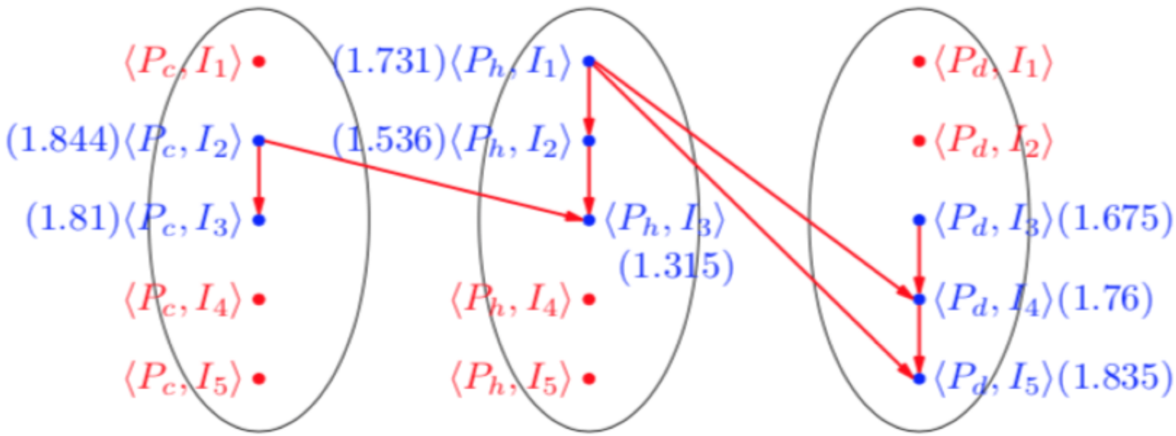
Conflict Graph



The ***minimum weighted independent set*** that includes ***exactly one*** interval activation for each process corresponds to the optimal solution.

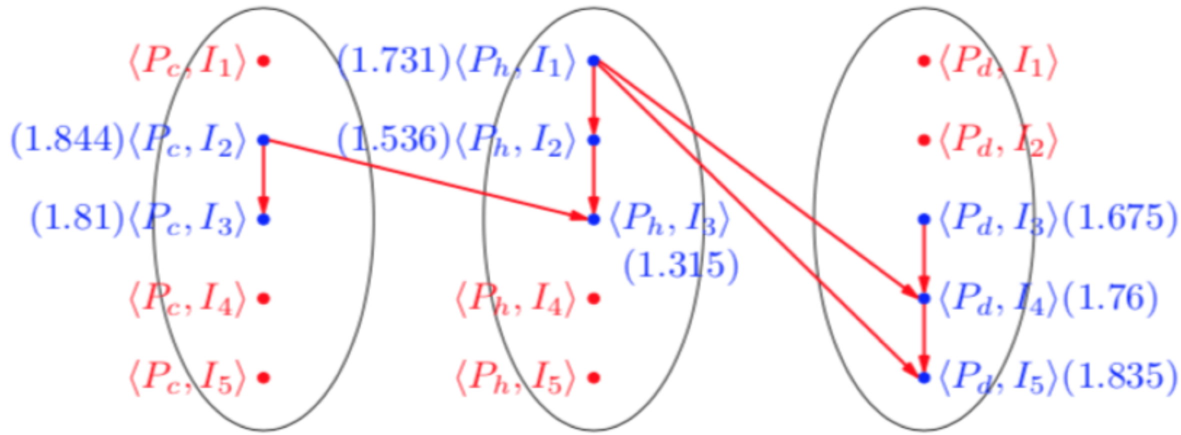
Issue 1: Different from the maximum weighted independent set.
Issue 2: Computing the maximum weighted independent set is NP-hard.

Solving Issue 1



A simple readjustment of weights converts the problem to a regular ***maximum weighted independent set*** problem.

Solving Issue 2



The directed graph is a POSET, that is, it is acyclic and transitive.

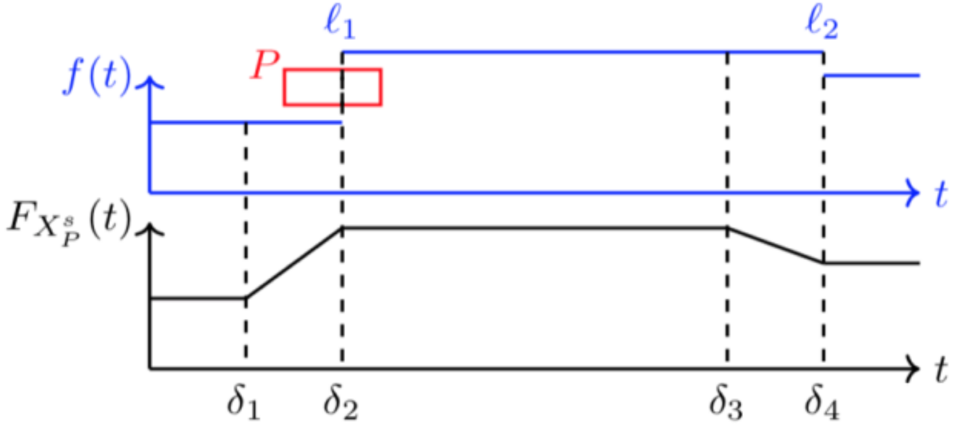
The maximum weighted independent set is the ***maximum weighted antichain*** in a POSET.

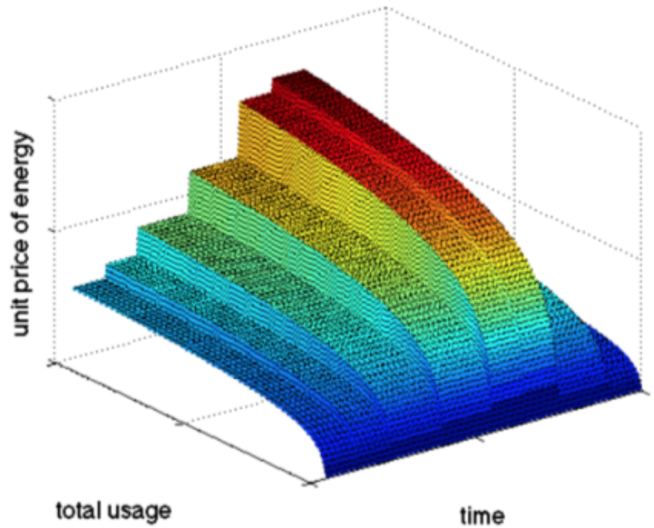
The maximum weighted antichain can be computed in polynomial time using a maxflow algorithm.

Tradeoff Against Makespan

- Find a schedule that is of minimum makespan among all schedules with cost ≤ γ optimal cost.
  - γ is a given suboptimality factor ≥ 1.
- **Key Observation:** makespan constraints are also simple temporal.
  - Do a Binary Search on makespan in the outer loop.
  - Solve the minimization of cost problem in the inner loop.
- Optimizations lead to Quasi Binary Search.

Conjectured Tractable Classes and Negative Results





Conjectured to be tractable for **Model A** and for ***concave*** dependency of unit price on total demand.

But provably NP-hard for ***convex*** dependency of unit price on total demand.

Conclusions and Future Work

- We presented a polynomial-time maxflow-based algorithm for optimally scheduling STNs with dynamic resource pricing.
  - Unit prices change with time but according to a piecewise constant function.
  - Processes demand energy requirements upfront.
- Conjectured tractable classes
  - Unit prices have a concave dependency on total demand.
  - Processes consume energy at a uniform wattage.
- Some NP-hard results
  - Unit prices have a convex dependency on total demand.
- **Future Work:** resolve conjectures; and apply algorithms to real-world engineering domains.