Block coordination of nonlinear networks and discrete optimization

pump scheduling for load shifting in drinking water distribution systems

Sophie Demassey, Valentina Sessa, Amirhossein Tavakoli (CMA, Mines Paris – PSL)

MIP Europe, 3 july 2025

Content

potential-flow networks & monotropic programming

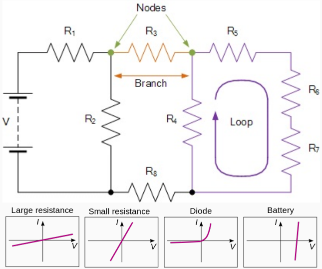
discrete bilevel models: design vs operation

variable splitting for pump scheduling

POTENTIAL-FLOW NETWORKS & MONOTROPIC PROGRAMMING

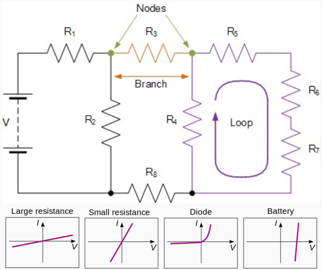
LINEAR POTENTIAL-FLOW NETWORKS: ELECTRIC CIRCUIT

- arcs A: conductors as resistors, batteries, diodes
- flow x_{ij}: signed current I
- flow conservation: Kirchhoff's current law
- potential loss $v_{ij} = u_i - u_j$: voltage V
- linear resistance $V = RI$ Ohm's law



LINEAR POTENTIAL-FLOW NETWORKS: ELECTRIC CIRCUIT

- arcs A: conductors as resistors, batteries, diodes
- flow x_{ij}: signed current I
- flow conservation: Kirchhoff's current law
- potential loss $v_{ij} = u_i - u_j$: voltage V
- linear resistance $V = RI$ Ohm's law

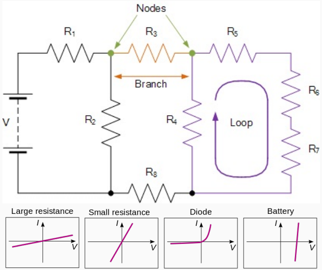


equilibrium

for a nodal demand d and incidence matrix E, find (x, u, v) s.t. $E^\top x = d$ and $v := -Eu = r^\top x$

LINEAR POTENTIAL-FLOW NETWORKS: ELECTRIC CIRCUIT

- arcs A: conductors as resistors, batteries, diodes
- flow x_{ij}: signed current I
- flow conservation: Kirchhoff's current law
- potential loss $v_{ij} = u_i - u_j$: voltage V
- linear resistance $V = RI$ Ohm's law



equilibrium

for a nodal demand d and incidence matrix E, find (x, u, v) s.t. $E^\top x = d$ and $v := -Eu = r^\top x$

KKT N&S conditions for **minimum energy dissipation**:

$$(P): \min_{x:E^\top x = d} \sum_{a \in A} \frac{r_a}{2} x_a^2 = \sum_{a \in A} f_a(x_a) \quad \text{with} \quad f_a'(x_a) = r_a x_a = v_a$$

GENERALIZATION TO NONLINEAR RESISTANCE

- transportation of newtonian fluids: water, gas, heat, blood, spring, traffic, telecom
- flow $x_a \in \mathbb{R}$: signed volume/rate on arc $a \in A$
- flow conservation/demand satisfaction $E_j^\top x = d_j$ on node $j \in N$
- potential loss $v_a := -E_a u$ due to resistance $v_a = \phi_a(x_a)$ of arc $a \in A$

***Equilibrium*(E, d_S, u_R)**

Given *boundary conditions* at nodes $N = S \cup R$, as fixed potential u_R or inflow d_S,
and a resistance function ϕ_a on each arc $a \in A$,
compute overall arc flows x, node potentials u (and arc potential losses $v = -Eu$) satisfying:

$$\text{demand } E_j^\top x = d_j \; \forall\, j \in S, \; \text{and resistance } v_a = \phi_a(x_a) \; \forall\, a \in A$$

STEADY-STATE EQUILIBRIUM

$Equilibrium(E, d_S, u_R)$

$$\begin{aligned}
\mathscr{E}(E, d_S, u_R) = \{(x_A, u_S) \in \mathbb{R}^A \times \mathbb{R}^S, & & & \text{(flows, potentials)} \\
x_j := \sum_a E_{aj} x_a = d_j & & \forall\, j \in S, & \text{demand} \\
v_a := \sum_j -E_{aj} u_j = \phi_a(x_a) & & \forall\, a \in A\} & \text{resistance}
\end{aligned}$$

STEADY-STATE EQUILIBRIUM

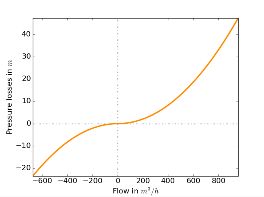
$Equilibrium(E, d_S, u_R)$

$$\mathcal{E}(E, d_S, u_R) = \{(x_A, u_S) \in \mathbb{R}^A \times \mathbb{R}^S, \qquad \text{(flows, potentials)}$$

$$x_j := \sum_a E_{aj} x_a = d_j \qquad \forall j \in S, \qquad \text{demand}$$

$$v_a := \sum_j -E_{aj} u_j = \phi_a(x_a) \qquad \forall a \in A\} \qquad \text{resistance}$$

comfy assumption: ϕ_a *continuous, strictly increasing, bijective* on $\mathbb{R}$



STEADY-STATE EQUILIBRIUM

Equilibrium(E, d_S, u_R)

$$\mathscr{E}(E, d_S, u_R) = \{(x_A, u_S) \in \mathbb{R}^A \times \mathbb{R}^S, \quad \text{(flows, potentials)}$$

$$x_j := \sum_a E_{aj} x_a = d_j \quad \forall j \in S, \quad \text{demand}$$

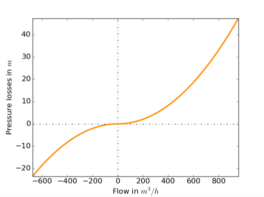
$$v_a := \sum_j -E_{aj} u_j = \phi_a(x_a) \quad \forall a \in A\} \quad \text{resistance}$$

comfy assumption: ϕ_a *continuous, strictly increasing, bijective* on $\mathbb{R}$

$\Rightarrow$ integral $f_a(x) = \int_0^x \phi_a(t)dt$ is *smooth, strictly convex, coercive*

$\Rightarrow \mathscr{E}$ is KKT N&S conditions for minimum energy loss $\sum_{a \in A} f_a(x_a)$

$\Rightarrow$ unique solution and strong duality



REFORMULATIONS

KKT (equilibrium): (x,u) solves

$$\mathscr{E}: E_S^\top x = d_S$$
$$v_a = \phi_a(x_a) \qquad \forall a \in A$$

primal (distribution): x solves

$$P: \min_x \sum_a f_a(x_a) + u_R^\top E_R^\top x$$
$$s.t.\ E_S^\top x = d_S$$

dual (differential): u solves

$$D: \min_u \sum_a f_a^*(v_a) + d_S^\top u_S$$
$$s.t.\ v := -Eu$$

strong duality: (x,u) solves

$$SD: E_S^\top x = d_S, v := -Eu$$
$$\sum_a \left(f_a(x_a) + f_a^*(v_a)\right) + u_R^\top E_R^\top x + d_S^\top u_S \leq 0.$$

with $f_a = \int \phi_a$ and $f_a^* = \int \phi_a^{-1}$: $f_a^*(v_a) = -f_a(\phi_a^{-1}(v_a)) + v_a\phi_a^{-1}(v_a)$ the convex conjugate*

*under our assumptions, f_a^* is also smooth, strictly convex and coercive.

Monotropic programming (Rockafellar, 1988)

additive convex objective
over linear constraints

f_j closed proper convex on $\mathbb{R}$
= l.s.c. possibly nonsmooth

$$P : \min_{x \in \mathbb{R}^J} \sum_{j \in J} f_j(x_j)$$

$$s.t. \sum_{j \in J} E_{ij} x_j = d_i \qquad \forall i \in I$$

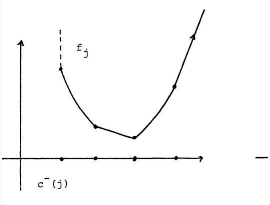


Figure 8.24

- monotropic aka "one-dimension convexity" extended to finite-dimension in [Bertsekas08]

Monotropic programming (Rockafellar, 1988)

additive convex objective
over linear constraints

$$P : \min_{x \in \mathbb{R}^J} \sum_{j \in J} f_j(x_j)$$

$$s.t. \sum_{j \in J} E_{ij} x_j = d_i \qquad \forall i \in I$$

f_j closed proper convex on $\mathbb{R}$
= l.s.c. possibly nonsmooth

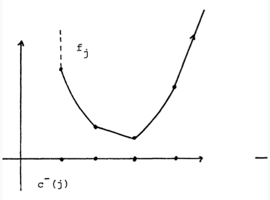


Figure 8.24

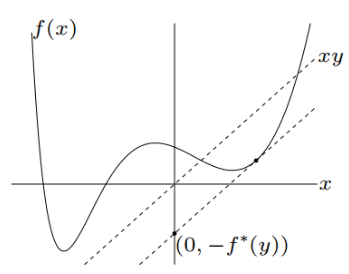
- monotropic aka "one-dimension convexity" extended to finite-dimension in [Bertsekas08]
- a class of convex programs behaving like linear programs:
  - combinatorial properties: finite set of descent directions (*elementary vectors*)
  - duality properties: strong duality, explicit symmetric dual
- other application: f_j piecewise linear/quad $\Longrightarrow$ same size dual

Fenchel duality

Let $f_j^* : v_j \in \mathbb{R} \mapsto \sup_{x_j}(x_j v_j - f_j(x_j))$ the *convex conjugate function* of $f_j \; \forall j \in J$

$$(P) : \min_{x \in \mathbb{R}^J} \sum_{j \in J} f_j(x_j)$$
$$s.t. \sum_{j \in J} E_{ij} x_j = d_i \qquad \forall i \in I$$

$$(D) : \min_{u \in \mathbb{R}^I} \sum_{i \in I} d_i u_i + \sum_{j \in J} f_j^*(v_j)$$
$$s.t. \; v_j := \sum_{i \in I} -E_{ij} u_i \qquad \forall j \in J$$



Fenchel duality

Let $f_j^* : v_j \in \mathbb{R} \mapsto \sup_{x_j}(x_j v_j - f_j(x_j))$ the *convex conjugate function* of $f_j \; \forall j \in J$

$$(P) : \min_{x \in \mathbb{R}^J} \sum_{j \in J} f_j(x_j) \qquad s.t. \sum_{j \in J} E_{ij} x_j = d_i \qquad \forall i \in I$$

$$(D) : \min_{u \in \mathbb{R}^I} \sum_{i \in I} d_i u_i + \sum_{j \in J} f_j^*(v_j) \qquad s.t. \; v_j := \sum_{i \in I} -E_{ij} u_i \qquad \forall j \in J$$

- *conjugate* f_j^* is l.s.c. convex: D is monotropic

Fenchel duality

Let $f_j^* : v_j \in \mathbb{R} \mapsto \sup_{x_j}(x_j v_j - f_j(x_j))$ the *convex conjugate function* of f_j $\forall j \in J$

$$(P) : \min_{x \in \mathbb{R}^J} \sum_{j \in J} f_j(x_j)$$
$$s.t. \sum_{j \in J} E_{ij} x_j = d_i \qquad \forall i \in I$$

$$(D) : \min_{u \in \mathbb{R}^I} \sum_{i \in I} d_i u_i + \sum_{j \in J} f_j^*(v_j)$$
$$s.t.\ v_j := \sum_{i \in I} -E_{ij} u_i \qquad \forall j \in J$$

- *conjugate* f_j^* is l.s.c. convex: D is monotropic
- *biconjugate* $f_j = f_j^{**}$ (as f_j l.s.c. convex): dual(dual)=primal

Fenchel duality

Let $f_j^* : v_j \in \mathbb{R} \mapsto \sup_{x_j}(x_j v_j - f_j(x_j))$ the *convex conjugate function* of f_j $\forall j \in J$

$$(P) : \min_{x \in \mathbb{R}^J} \sum_{j \in J} f_j(x_j) \quad s.t. \sum_{j \in J} E_{ij} x_j = d_i \qquad \forall i \in I$$

$$(D) : \min_{u \in \mathbb{R}^I} \sum_{i \in I} d_i u_i + \sum_{j \in J} f_j^*(v_j) \quad s.t.\ v_j := \sum_{i \in I} -E_{ij} u_i \qquad \forall j \in J$$

- *conjugate* f_j^* is l.s.c. convex: D is monotropic
- *biconjugate* $f_j = f_j^{**}$ (as f_j l.s.c. convex): dual(dual)=primal
- *Fenchel inequality*: $f_j(x_j) + f_j^*(v_j) \geq x_j v_j$ and equality holds iff $v_j \in \partial f_j(x_j)$ (*subgradient*)

Fenchel duality

Let $f_j^* : v_j \in \mathbb{R} \mapsto \sup_{x_j}(x_j v_j - f_j(x_j))$ the *convex conjugate function* of f_j $\forall j \in J$

$$(P): \min_{x \in \mathbb{R}^J} \sum_{j \in J} f_j(x_j) \quad s.t. \sum_{j \in J} E_{ij} x_j = d_i \qquad \forall i \in I$$

$$(D): \min_{u \in \mathbb{R}^I} \sum_{i \in I} d_i u_i + \sum_{j \in J} f_j^*(v_j) \quad s.t.\ v_j := \sum_{i \in I} -E_{ij} u_i \qquad \forall j \in J$$

- *conjugate* f_j^* is l.s.c. convex: D is monotropic
- *biconjugate* $f_j = f_j^{**}$ (as f_j l.s.c. convex): dual(dual)=primal
- *Fenchel inequality*: $f_j(x_j) + f_j^*(v_j) \geq x_j v_j$ and equality holds iff $v_j \in \partial f_j(x_j)$ (*subgradient*)
- *Fenchel equality*: strong duality $\iff$ stationarity:
  $0 = \sum_j \left(f_j(x_j) + f_j^*(v_j)\right) + \sum_i d_i u_i = \sum_j \left(f_j(x_j) + f_j^*(v_j) - x_j v_j\right) \iff v_j \in \partial f_j(x_j) \forall j$

General resistance as Subdifferential

generalize the equilibrium problem to $\mathscr{E} : \{(x, v) : Ex = d, v_j \in \partial f_j(x_j),\ \forall j\}$ for:

- different boundary conditions
- resistance as a maximal monotone relation $(x_j, v_j) \in \Gamma_j \subset \mathbb{R}^2$ (the *characteristic curve*):

$$\Gamma_j := \{(x_j, v_j) \,|\, v_j \in \partial f_j(x_j)\} \iff f_j(x_j) + f_j^*(v_j) = x_j v_j$$

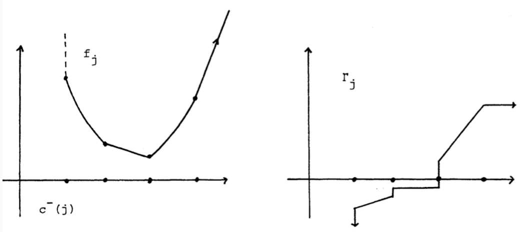


Figure 8.24

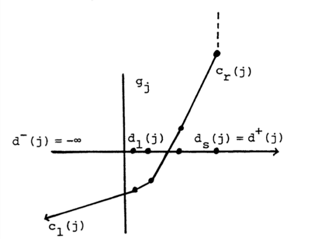
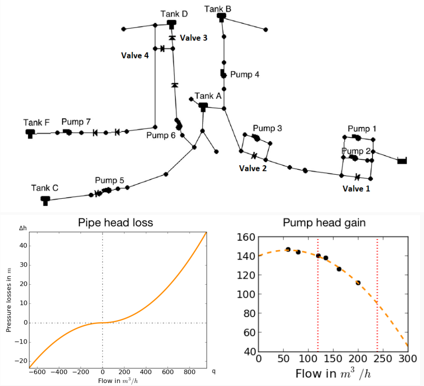


Figure 8.37

[Rockafellar88]

NONLINEAR NETWORK: DRINKING WATER DISTRIBUTION



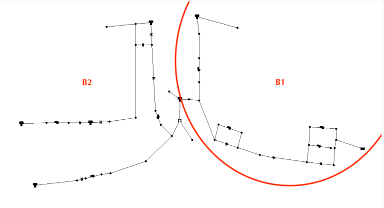
- A = { pipes, pumps, valves }
- $I = S \cup R$: Service nodes (junctions) and Reservoir (tanks, sources)
- x = water flow rate
- u = hydraulic head = pressure + elevation
- nonlinear resistance:
    - friction in pipes $\phi(x) = sgn(x)\alpha|x|^2$
    - discharge pressure in pumps $\phi(x) = |\beta x|x + \kappa$

hydraulic simulation (e.g. EPANET):

solve $\mathscr{E}$ aka *network analysis problem* with Todini-Pilati (Newton-Raphson) algorithm

SPATIAL DECOMPOSITION

$G = \cup_{b \in B}(N_b, A_b)$ a graph partition along the nodes R with fixed potential



equilibrium is separable along the potential nodes R

$$\mathscr{E}(E, d_S, u_R) = \bigcup_{b \in B} \mathscr{E}(E_b, d_{S_b}, u_{R_b})$$

$$= \bigcup_{b \in B} \{(x, u) \in \mathbb{R}^{A_b} \times \mathbb{R}^{S_b} : E_{S_b}^{\top} x = d_{S_b}, \; v_a = \phi_a(x_a) \; \forall \, a \in A_b\}$$

COMPLEXITY

is $\mathscr{E}(E, d_S, u_R)$ hard ?

- NO ! separable linearly constrained strictly convex pb, solved quickly with Newton algorithms

COMPLEXITY

is $\mathscr{E}(E, d_S, u_R)$ hard ?

- NO ! separable linearly constrained strictly convex pb, solved quickly with Newton algorithms
- for fixed (E, d_S, u_R)

COMPLEXITY

is $\mathscr{E}(E, d_S, u_R)$ hard ?

- NO ! separable linearly constrained strictly convex pb, solved quickly with Newton algorithms
- for fixed (E, d_S, u_R)
- **but** not always fixed: uncertain d, active E, dynamic u

DISCRETE BILEVEL MODELS:
DESIGN VS OPERATION

GRAVITY-FED NETWORK DESIGN

pipe layout (deterministic)

given a worst-case demand scenario d, select a pipe layout E in a discrete set such that $\mathscr{E}(E,d,u)$ is feasible and installation cost minimum

$$\min_{y0/1,x,u} \sum_a \sum_k c_{ak} y_{ak}$$

$$s.t. (x,u) \in \mathscr{E}(E(y), d_S, u_R)$$

$$y_{ak} = 0 \implies x_{ak} = v_{ak} = 0 \qquad \forall a \in A, k \in K$$

$$\sum_{k \in K} y_{ak} = 1, u_i - u_j = \sum_k v_{ak} \qquad \forall a = (i,j) \in A.$$

† B. Tasseff, R. Bent, M. Epelman, D. Pasqualini, P. Van Hentenryck (2020) Exact Mixed-integer Convex Programming Formulation for Optimal Water Network Design.

GRAVITY-FED NETWORK DESIGN

pipe layout (deterministic)

given a worst-case demand scenario d, select a pipe layout E in a discrete set such that $\mathscr{E}(E, d, u)$ is feasible and installation cost minimum

$$\min_{y0/1,x,u} \sum_a \sum_k c_{ak} y_{ak}$$
$$s.t. (x,u) \in \mathscr{E}(E(y), d_S, u_R)$$
$$y_{ak} = 0 \implies x_{ak} = v_{ak} = 0 \qquad \forall a \in A, k \in K$$
$$\sum_{k \in K} y_{ak} = 1, u_i - u_j = \sum_k v_{ak} \qquad \forall a = (i,j) \in A.$$

$$\min \sum_a \sum_k c_{ak} y_{ak}$$
$$s.t. \sum_{a \in A} \sum_{k \in K} \left(f_{ak}(x_{ak}) + f^*_{ak}(v_{ak}) \right) + u_R^\top x_R + d_S^\top u_S \leq 0$$
$$y_{ak} = 0 \implies x_{ak} = v_{ak} = 0$$
$$\sum_{k \in K} y_{ak} = 1, u_i - u_j = \sum_k v_{ak}.$$

High combinatorics but the strong duality reformulation is convex (although nonpolynomial)†

---

†B. Tasseff, R. Bent, M. Epelman, D. Pasqualini, P. Van Hentenryck (2020) Exact Mixed-integer Convex Programming Formulation for Optimal Water Network Design.

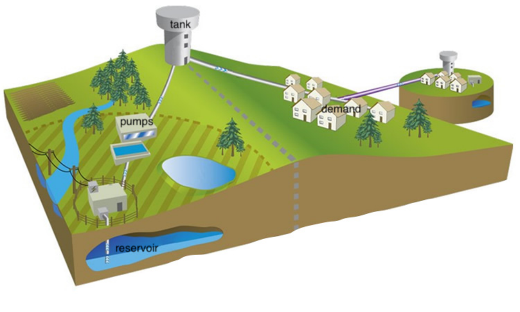
LOAD SHIFTING IN DRINKING WATER DISTRIBUTION

pumping is energy-intensive

pump in advance of demand to save energy,
+ to reduce bill + to support power grid

Opportunities:

- water tanks for energy storage
- nonlinear efficiency
- dynamic electricity tariff



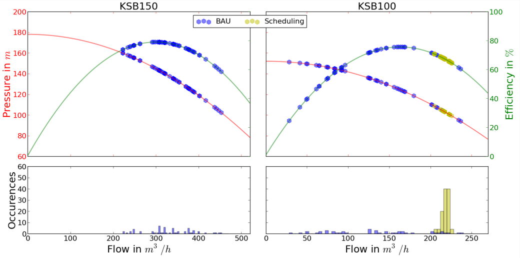
LOAD SHIFTING IN DRINKING WATER DISTRIBUTION

pumping is energy-intensive

pump in advance of demand to save energy,
+ to reduce bill + to support power grid

Opportunities:

- water tanks for energy storage
- nonlinear efficiency
- dynamic electricity tariff



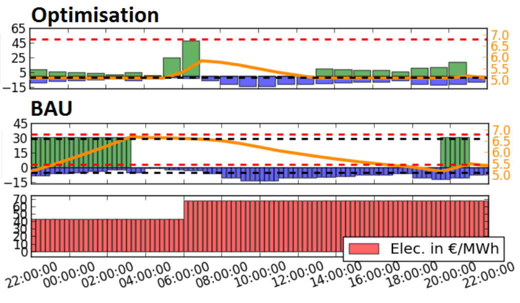
LOAD SHIFTING IN DRINKING WATER DISTRIBUTION

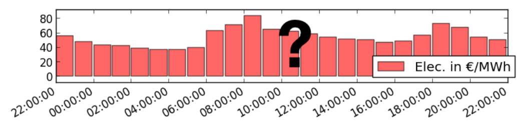
pumping is energy-intensive

pump in advance of demand to save energy,
+ to reduce bill + to support power grid

Opportunities:

- water tanks for energy storage
- nonlinear efficiency
- dynamic electricity tariff





LOAD SHIFTING IN PRESSURIZED NETWORKS

pump scheduling

given a demand profile $(d_t)_t$ on a discrete time horizon, reconfigure $(E_t)_t$ such that $\mathscr{E}(E_t, d_t, u_t)$ and **tank conservation+bounds** are feasible and **operation costs** minimum

$$\min_{y0/1,x,u} \sum_{t,a} c^0_{ta} y_{ta} + c^1_{ta} x_{ta}$$

$$s.t. (x_t, u_{tS}) \in \mathscr{E}(E(y_t), d_{tS}, u_{tR}) \qquad \forall t \in T$$

$$y_{ta} = 0 \implies x_{ta} = 0 \qquad \forall a \in A, t \in T$$

$$u_{(t+1)R} = u_{tR} + x_{tR} \qquad \forall t \in T$$

$$\underline{U}_{tR} \leq u_{tR} \leq \overline{U}_{tR} \qquad \forall t \in T.$$

pump scheduling

given a demand profile $(d_t)_t$ on a discrete time horizon, reconfigure $(E_t)_t$ such that $\mathscr{E}(E_t, d_t, u_t)$ and **tank conservation+bounds** are feasible and **operation costs** minimum

- multiple sequence-dependent followers

$$\min_{y0/1,x,u} \sum_{t,a} c^0_{ta} y_{ta} + c^1_{ta} x_{ta}$$

$$
\begin{aligned}
s.t. (x_t, u_{tS}) &\in \mathscr{E}(E(y_t), d_{tS}, u_{tR}) && \forall t \in T \\
y_{ta} = 0 &\implies x_{ta} = 0 && \forall a \in A, t \in T \\
u_{(t+1)R} &= u_{tR} + x_{tR} && \forall t \in T \\
\underline{U}_{tR} \le u_{tR} &\le \overline{U}_{tR} && \forall t \in T.
\end{aligned}
$$

LOAD SHIFTING IN PRESSURIZED NETWORKS

pump scheduling

given a demand profile $(d_t)_t$ on a discrete time horizon, reconfigure $(E_t)_t$ such that $\mathscr{E}(E_t, d_t, u_t)$ and **tank conservation+bounds** are feasible and **operation costs** minimum

$$\min_{y0/1,x,u} \sum_{t,a} c^0_{ta} y_{ta} + c^1_{ta} x_{ta}$$

$$\begin{aligned}
s.t.(x_t, u_{tS}) &\in \mathscr{E}(E(y_t), d_{tS}, u_{tR}) && \forall t \in T \\
y_{ta} = 0 &\implies x_{ta} = 0 && \forall a \in A, t \in T \\
u_{(t+1)R} &= u_{tR} + x_{tR} && \forall t \in T \\
\underline{U}_{tR} \leq u_{tR} &\leq \overline{U}_{tR} && \forall t \in T.
\end{aligned}$$

- multiple sequence-dependent followers
- nonconvex strong duality reformulation

$$\sum_a \left(f_a(x_{ta}) + f^*_a(v_{ta})\right) + u^\top_{tR} x_{tR} + d^\top_{tS} u_{tS} \leq 0$$

LOAD SHIFTING IN PRESSURIZED NETWORKS

pump scheduling

given a demand profile $(d_t)_t$ on a discrete time horizon, reconfigure $(E_t)_t$ such that $\mathscr{E}(E_t, d_t, u_t)$ and **tank conservation+bounds** are feasible and **operation costs** minimum

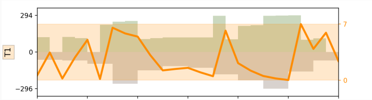
$$\min_{y0/1,x,u} \sum_{t,a} c^0_{ta} y_{ta} + c^1_{ta} x_{ta}$$

$$\begin{aligned}
s.t.(x_t, u_{tS}) &\in \mathscr{E}(E(y_t), d_{tS}, u_{tR}) && \forall t \in T \\
y_{ta} = 0 &\implies x_{ta} = 0 && \forall a \in A, t \in T \\
u_{(t+1)R} &= u_{tR} + x_{tR} && \forall t \in T \\
\underline{U}_{tR} \leq u_{tR} &\leq \overline{U}_{tR} && \forall t \in T.
\end{aligned}$$

- multiple sequence-dependent followers
- nonconvex strong duality reformulation

$$\sum_a \left(f_a(x_{ta}) + f^*_a(v_{ta})\right) + u^\top_{tR} x_{tR} + d^\top_{tS} u_{tS} \leq 0$$

- tight tank limits, long time steps:



sensitivity to Δy: feasibility alone is an issue

VARIABLE SPLITTING FOR PUMP SCHEDULING

SOLUTIONS FOR PUMP SCHEDULING

$$z(y^*, x^*, u^*) = \min_{y0/1,x,u} \sum_{t,a} c^0_{ta} y_{ta} + c^1_{ta} x_{ta}$$

$$\begin{aligned} & s.t. (x_t, u_{tS}) \in \mathscr{E}(E(y_t), d_{tS}, u_{tR}) && \forall t \in T \\ & y_{ta} = 0 \implies x_{ta} - v_{ta} = 0 && \forall a \in A, t \in T \\ & u_{(t+1)R} = u_{tR} + x_{tR} && \forall t \in T \\ & \underline{U}_{tR} \leq u_{tR} \leq \overline{U}_{tR} && \forall t \in T. \end{aligned}$$

1. **approximation or relaxation**
   - PWL approx [Morsi12,...]
   - LP relax [Burgschweiger09]
   - lag relax, ADMM [Ghaddar15, Ulusoy25]
   - convex relax + global search [Bonvin21]

   → *complexity/accuracy trade-off*
2. **simulation-optimization**
   - metaheuristics e.g. GA [Mackle95,...]
   - Benders decomposition [NaoumSawaya15]
   - LP approx [Bonvin&Demassey19]

   → *slow convergence, infeasibilities*

SOLUTIONS FOR PUMP SCHEDULING

$$z(y^*, x^*, u^*) = \min_{y0/1,x,u} \sum_{t,a} c^0_{ta} y_{ta} + c^1_{ta} x_{ta}$$

$$\begin{aligned}
s.t. (x_t, u_{tS}) &\in \mathscr{E}(E(y_t), d_{tS}, u_{tR}) && \forall t \in T \\
y_{ta} = 0 &\implies x_{ta} - v_{ta} = 0 && \forall a \in A, t \in T \\
u_{(t+1)R} &= u_{tR} + x_{tR} && \forall t \in T \\
\underline{U}_{tR} \leq u_{tR} &\leq \overline{U}_{tR} && \forall t \in T.
\end{aligned}$$

1. **approximation or relaxation**
   - PWL approx [Morsi12,...]
   - LP relax [Burgschweiger09]
   - lag relax, ADMM [Ghaddar15, Ulusoy25]
   - convex relax + global search [Bonvin21]

   → *complexity/accuracy trade-off*
2. **simulation-optimization**
   - metaheuristics e.g. GA [Mackle95,...]
   - Benders decomposition [NaoumSawaya15]
   - LP approx [Bonvin&Demassey19]

   → *slow convergence, infeasibilities*

- most optimize the value function $g(y) = z(y, x(y), u(y))$
- search the discrete upper-level y-space but feasible solutions are scarce/sparse

SOLUTIONS FOR PUMP SCHEDULING

$$z(y^*, x^*, u^*) = \min_{y0/1,x,u} \sum_{t,a} c^0_{ta} y_{ta} + c^1_{ta} x_{ta}$$

$$\begin{aligned}
s.t. (x_t, u_{tS}) &\in \mathscr{E}(E(y_t), d_{tS}, u_{tR}) && \forall t \in T \\
y_{ta} = 0 &\implies x_{ta} - v_{ta} = 0 && \forall a \in A, t \in T \\
u_{(t+1)R} &= u_{tR} + x_{tR} && \forall t \in T \\
\underline{U}_{tR} &\leq u_{tR} \leq \overline{U}_{tR} && \forall t \in T.
\end{aligned}$$

1. **approximation or relaxation**
   - PWL approx [Morsi12,...]
   - LP relax [Burgschweiger09]
   - lag relax, ADMM [Ghaddar15, Ulusoy25]
   - convex relax + global search [Bonvin21]

   → *complexity/accuracy trade-off*
2. **simulation-optimization**
   - metaheuristics e.g. GA [Mackle95,...]
   - Benders decomposition [NaoumSawaya15]
   - LP approx [Bonvin&Demassey19]

   → *slow convergence, infeasibilities*

- most optimize the value function $g(y) = z(y, x(y), u(y))$
- search the discrete upper-level y-space but feasible solutions are scarce/sparse
- time decomposition: fix coupling variables u_R, not just relax coupling constraints

PROPOSED APPROACH

search the continuous u_R-space:

- solutions are less sensitive to Δu_R
- near-optimal storage profiles u_R are more predictable

PROPOSED APPROACH

search the continuous u_R-space:

- solutions are less sensitive to Δu_R
- near-optimal storage profiles u_R are more predictable
- fix u_R then split time
- fix u_R then split space, then enumerate y solutions on each component
- fix u_R + SD reformulation then split primal/flow and dual/potential

PROPOSED APPROACH

search the continuous u_R-space:

- solutions are less sensitive to Δu_R
- near-optimal storage profiles u_R are more predictable
- fix u_R then split time
- fix u_R then split space, then enumerate y solutions on each component
- fix u_R + SD reformulation then split primal/flow and dual/potential

- however the value function $g(u_R) = z(y(u_R), x(u_R), u(u_R))$ is nonconvex nonsmooth everywhere
- variable split methods (e.g. ADMM) do not converge a priori

PROPOSED APPROACH

search the continuous u_R-space:

- solutions are less sensitive to Δu_R
- near-optimal storage profiles u_R are more predictable
- fix u_R then split time
- fix u_R then split space, then enumerate y solutions on each component
- fix u_R + SD reformulation then split primal/flow and dual/potential

- however the value function $g(u_R) = z(y(u_R), x(u_R), u(u_R))$ is nonconvex nonsmooth everywhere
- variable split methods (e.g. ADMM) do not converge a priori
- adapt as a heuristic to reconcile the inflow profile $(x_{tR})_t$ and the storage profile $(u_{t+1,R} - u_{tR})_t$ and reach strictly-feasible near-optimal solutions

OPTION 0: DUALIZE THE TIME-COUPLING CONSTRAINTS

ex: lagrangian subproblem

$$(P) : \min_{x,y,u} \sum_t c_t(x_t, y_t) + \mu_t^\top (u_{t+1,R} - u_{tR} - x_{tR})$$

$$s.t. : x_t \in \mathscr{E}(E(y_t), d_t, u_{tR}) \qquad \forall t \in T$$

- the model becomes separable in time
- but each static component remains hard (and poor) as the initial state u_{Rt} **is unknown**

OPTION 1: FULL VARIABLE-SPLIT

1: fix storage u_R, then compute (x, y)

$$P(u_R) : \min_{(x,y)} \sum_t c_t(x_t, y_t) + \mu_t^\top (u_{t+1,R} - u_{tR} - x_{tR}) + \lambda_t^\top (\mathscr{E}(E(y_t), d_t, u_{tR}))$$

$\downarrow \quad \uparrow \quad$ update μ, λ

2: fix command (x, y), then compute u_R

$$P(x, y) : \min_{u} \sum_t c_t(x_t, y_t)) + \mu_t^\top (u_{t+1,R} - u_{tR} - x_{tR}) + \lambda_t^\top (\mathscr{E}(E(y_t), d_t, u_{tR}))$$

- no theoretical convergence of ADMM with nonconvex coupling constraints
- $P(u_R)$ is too poor, $P(x, y)$ too hard (inverse problem)

OPTION 2: PARTIAL SPLIT AND ADM-LIKE

no theory ? be practical: keep $\mathscr{E}$ in $P(u_R)$, but **drop it** from $P(x, y)$

1: fix storage u_R, then compute (x, y)

$$P(u_R) : \min_{(x,y)} \sum_t c_t(x_t, y_t) + \mu_t^\top (u_{t+1,R} - u_{tR} - x_{tR})$$

$$s.t. : x_t \in \mathscr{E}(E(y_t), d_t, u_{tR}) \qquad \forall t \in T$$

simulate $\mathscr{E}$, $\forall$ time t, graph component b, 0/1 vector y_{tb}

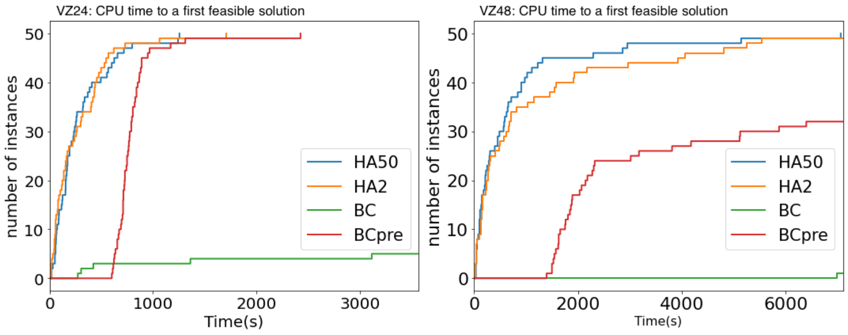
$\downarrow$ $\uparrow$ stop when $\|u_{t+1,R} - u_{tR} - x_{tR}\| < \epsilon$

2: fix command (x, y), then compute u_R

$$P(x, y) : \min_u \sum_t c_t(x_t, y_t) + \mu_t^\top (u_{t+1,R} - u_{tR} - x_{tR})$$

EXPERIMENTS: LEARNING PROFILES u_R + PARTIAL SPLIT

- **HA**: partial split $\mu \in \{50, 2\}$ from multiple learned storage profiles [ISCO 2024]
- **BC**: SOA Branch-and-Check [Opt&Eng 2021] + **BCpre** advanced preprocessing [ICAE 2022]
- run algorithms on 50 instances within 2 hours; stop at the first feasible solution



OPTION 3: FULL VARIABLE-SPLIT ON SD MODEL

step 1: fix storage u_R, then compute schedule and flow (y, x)

$$w(u_R): \min_{(y,x)} \sum_{t \in \mathcal{T}} c_t^0 y_t + c_t^1 x_t + \mu_t^\top (u_{t+1,R} - u_{tR} - x_{tR}) + \rho_t SD_t(x_t, u_{tS}) : (1 - y_t) x_t = 0,\ x_{tS} = d_{tS} \forall t$$

with $SD_t(x_t, u_t) = f(x_t) + f^*(v_t) + u_{tR}^\top x_{tR} + d_{tS}^\top u_{tS}$

$w(u_R)$ is separable in time and space (graph partition on tank nodes)

OPTION 3: FULL VARIABLE-SPLIT ON SD MODEL

step 1: fix storage u_R, then compute schedule and flow (y, x)

$$w(u_R) : \min_{(y,x)} \sum_{t\in\mathcal{T}} c_t^0 y_t + c_t^1 x_t + \mu_t^\top (u_{t+1,R} - u_{tR} - x_{tR}) + \rho_t SD_t(x_t, u_{tS}) : (1 - y_t)x_t = 0, \; x_{tS} = d_{tS} \forall t$$

with $SD_t(x_t, u_t) = f(x_t) + f^*(v_t) + u_{tR}^\top x_{tR} + d_{tS}^\top u_{tS}$

$w(u_R)$ is separable in time and space (graph partition on tank nodes)

moreover, each $w_t(u_{tR})$ is **separable in primal (x) /dual (u_S) parts** each corresponding to an equilibrium problem perturbed by the UL cost c_t and penalties μ, ρ:

perturbed primal

$$P_t(y_t, u_{tR}) : \min_{x_t} \rho_t f(x_t) + (\rho_t u_{tR} - \mu_t + c_t^1)^\top x_t$$

$$s.t. : x_{tS} = d_{tS}, \; (1 - y_t)^\top x_t = 0.$$

perturbed dual

$$D_t(y_t, u_{tR}) : \min_{u_{tS}} \rho_t f^*(v_t) + \rho_t d_t^\top u_{tS}$$

$$s.t. : v_t = -E u_t.$$

CONCLUSION AND PERSPECTIVE

- storage/control variable split leads to chain decomposition: time $\longrightarrow$ space $\longrightarrow$ primal/dual
- integrate the economic and penalty costs within the physical equilibrium problem

CONCLUSION AND PERSPECTIVE

- storage/control variable split leads to chain decomposition: time ⟶ space ⟶ primal/dual
- integrate the economic and penalty costs within the physical equilibrium problem
- alternative bilevel view: (leader) implied continuous storage variables (follower) discrete decisions
- alternative ML/MIP hybrid: ML for optimality, MIP for feasibility

CONCLUSION AND PERSPECTIVE

- storage/control variable split leads to chain decomposition: time $\longrightarrow$ space $\longrightarrow$ primal/dual
- integrate the economic and penalty costs within the physical equilibrium problem
- alternative bilevel view: (leader) implied continuous storage variables (follower) discrete decisions
- alternative ML/MIP hybrid: ML for optimality, MIP for feasibility
- ongoing works: convergence and application to traffic network design, and energy system expansion planning

CONCLUSION AND PERSPECTIVE

- storage/control variable split leads to chain decomposition: time $\longrightarrow$ space $\longrightarrow$ primal/dual
- integrate the economic and penalty costs within the physical equilibrium problem
- alternative bilevel view: (leader) implied continuous storage variables (follower) discrete decisions
- alternative ML/MIP hybrid: ML for optimality, MIP for feasibility
- ongoing works: convergence and application to traffic network design, and energy system expansion planning
- save a polar bear, optimize load shifting

References

- nonlinear flows: **Rockafellar**
- ADMM: **Boyd** on proximal algorithms; **Rockafellar**, **Eckstein** on monotone operators; apps in ML and OPF
- **G. Bonvin, S. Demassey** Extended linear formulation of the pump scheduling problem in water distribution networks. INOC 2019.
- **G. Bonvin, S. Demassey, A. Lodi** Pump scheduling in drinking water distribution networks with an LP/NLP-based branch and bound. Opt&Eng 2021.
- **A. Tavakoli, V. Sessa, S. Demassey** Strengthening mathematical models for pump scheduling in water distribution. ICAE 2022.
- **S.Demassey, V. Sessa, A. Tavakoli** Deep learning and alternating direction method for discrete control with storage. ISCO 2024.
- papers available at `https://sofdem.github.io/`
- code available at `https://github.com/sofdem/gopslpnlpbb`