Recent advances in the solution of Unit-Commitment problems

Antonio Frangioni

Dipartimento di Informatica, Università di Pisa

ROADEF 2008

February 27, 2008

Outline

Outline

Outline

Outline

Outline

Outline

Outline

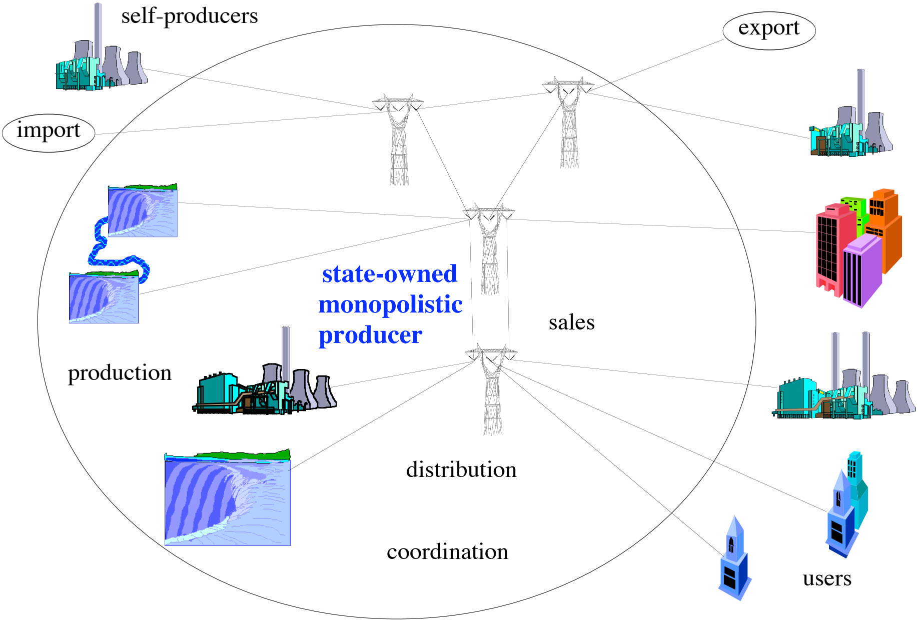
Outline

Outline

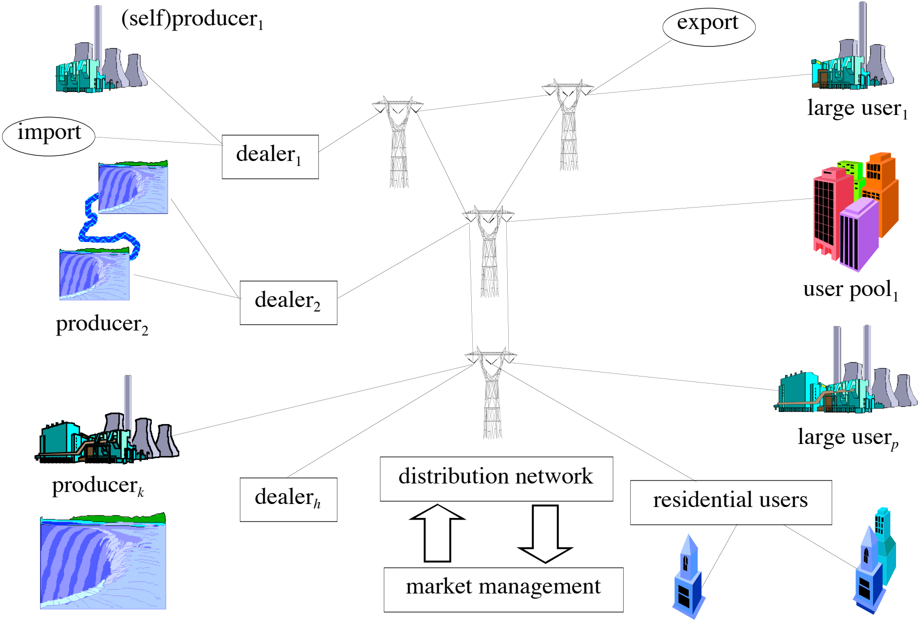
Outline

1. The Electrical system
2. The Hydro-Thermal Unit Commitment problem
3. A MIQP Formulation
4. Lagrangian Relaxation
5. Handling Ramp Constraints
6. Free Market versions
7. MILP Formulations
8. Small Detour: Portfolio Problems
9. A Hybrid Lagrangian-MILP Approach
10. Conclusions

The electrical system, monopolistic version



The electrical system, free-market version



Why the electrical system is complex

Two simple reasons:

- electricity cannot be stored $\Rightarrow$ must be produced exactly when needed

$$+$$

Why the electrical system is complex

Two simple reasons:

- electricity cannot be stored $\Rightarrow$ must be produced exactly when needed

$$+$$

- electricity does not go where you say, it goes where Kirchoff says

Why the electrical system is complex

Two simple reasons:

- electricity cannot be stored $\Rightarrow$ must be produced exactly when needed

$$+$$

- electricity does not go where you say, it goes where Kirchoff says

$$\Downarrow$$

The electrical system must be
constantly managed in real time to satisfy the demand while
respecting the complex technical constraints of
generating units and of the distribution network

Generating units

- Generating units (circa 2004):

type	no.	net installed power (MW)			avg. peak
		producers	self-producers	total	
hydro	1981	20.177	337	20.514	13.450
thermal	1818	50.069	4.545	54.614	34.750
geothermal	37	666		666	550
solar+wind	99+10	783		783	200
total	3945	71.695	4.881	76.576	48.950

  A large variety:
  - thermal units: different technologies (turbogas, combined cycle, ...) and sizes (10 – 500+ MW)
  - hydro units: different types (flowing water, reservoirs, cascaeds, ...) and sizes (1 – 200+ MW)
  - self-producers, co-generation (refineries, foundries, sugar, ...)
- Too few: daily imports for 3 – 6.5 GW

- Electrical network (multi-level):
  - 21.885 km VHV (9.880 km 380 kV + 12.005 km 220 kV)
  - 44.800 km HV (150 – 120 kV)
  - 21.700 km RTN + 23.100 km others
  - ?????? km MV + LV

- Too little capacity ⇒ zonal prices

The electrical market(s)

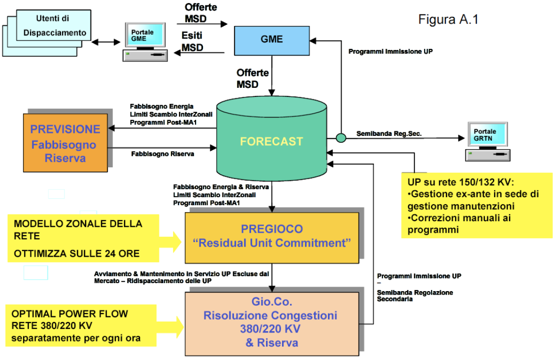
- Three markets, to be solved in sequence and influencing each other:
  - day-ahead market (main), network constrained (zonal prices)
  - adjustment market (power swap between units)
  - auxiliary services market (1/2/3–ary active reserve, network security)

The electrical market(s)

- Three markets, to be solved in sequence and influencing each other:
  - day-ahead market (main), network constrained (zonal prices)
  - adjustment market (power swap between units)
  - auxiliary services market (1/2/3–ary active reserve, network security)
- Plus bilateral contracts, misc stuff (CIP6, renewables, . . .)

The electrical market(s)

- Three markets, to be solved in sequence and influencing each other:
  - day-ahead market (main), network constrained (zonal prices)
  - adjustment market (power swap between units)
  - auxiliary services market (1/2/3–ary active reserve, network security)
- Plus bilateral contracts, misc stuff (CIP6, renewables, ...)
- To be cleared:
  - daily
  - in a few hours
  - with nontrivial models



The electrical market (cont.d)

- Worth about 40 M€ / day

The electrical market (cont.d)

- Worth about 40 M€ / day

  . . . the market value, but how much a black-out day costs?

The electrical market (cont.d)

- Worth about 40 M€ / day

  . . . the market value, but how much a black-out day costs?

- About 15 G€/ year (that's 1% of GNP)

The electrical market (cont.d)

- Worth about 40 M€ / day

  ...the market value, but how much a black-out day costs?

- About 15 G€/ year (that's 1% of GNP)

- About 5 times as much as the whole TLC sector

The electrical market (cont.d)

- Worth about 40 M€ / day

  . . . the market value, but how much a black-out day costs?

- About 15 G€/ year (that's 1% of GNP)

- About 5 times as much as the whole TLC sector

- Importance and complexity can only increase:
  - energy costs increasing
  - CO_2, emissions constraints
  - green/black certificates
  - small-scale, non-continuous production (solar, wind, hydrogen, . . .)

The electrical market (cont.d)

- Worth about 40 M€ / day

  . . . the market value, but how much a black-out day costs?

- About 15 G€/ year (that's 1% of GNP)
- About 5 times as much as the whole TLC sector
- Importance and complexity can only increase:
  - energy costs increasing
  - CO_2, emissions constraints
  - green/black certificates
  - small-scale, non-continuous production (solar, wind, hydrogen, . . .)
- All sorts of thorny political/technical issues (hydrogen, nuclear, . . .)

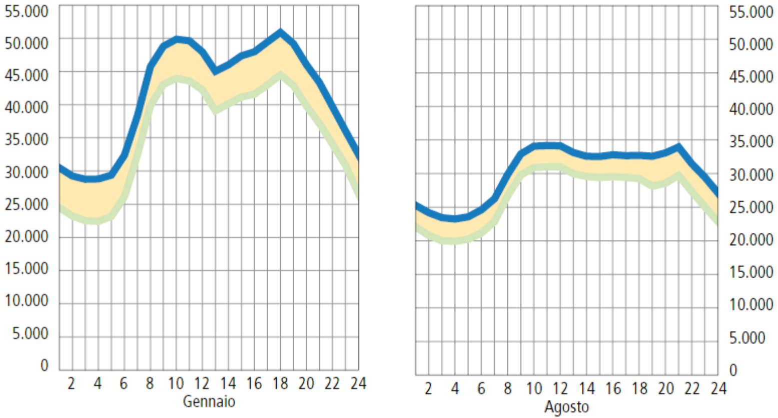
The Hydro-Thermal Unit Commitment problem

Given:

- a discretized (short) time horizon $\mathcal{T}$ (day/week, 60/15 minutes)

Given:

- a discretized (short) time horizon $\mathcal{T}$ (day/week, 60/15 minutes)
- a (forecasted) energy demand $\bar{d}_t$ for $t \in \mathcal{T}$



The Hydro-Thermal Unit Commitment problem (2)

- A set P of thermal units (coal, gas, oil, nuclear, ...)
- For each $i \in P$

The Hydro-Thermal Unit Commitment problem (2)

- A set P of thermal units (coal, gas, oil, nuclear, ...)
- For each $i \in P$
  - nonlinear energy cost (typically quadratic), startup cost
  - maximum and minimum (if committed) power output
  - constraints on how often the unit can be brought on/off line
  - constraints on maximum increase/decrease of power output
  - others (valve points, must run/must-not run, ...)

The Hydro-Thermal Unit Commitment problem (2)

- A set P of thermal units (coal, gas, oil, nuclear, ...)
- For each $i \in P$
  - nonlinear energy cost (typically quadratic), startup cost
  - maximum and minimum (if committed) power output
  - constraints on how often the unit can be brought on/off line
  - constraints on maximum increase/decrease of power output
  - others (valve points, must run/must-not run, ...)
- A set H of hydro (cascade) units
- For each $j \in H$

The Hydro-Thermal Unit Commitment problem (2)

- A set P of thermal units (coal, gas, oil, nuclear, ...)
- For each $i \in P$
  - nonlinear energy cost (typically quadratic), startup cost
  - maximum and minimum (if committed) power output
  - constraints on how often the unit can be brought on/off line
  - constraints on maximum increase/decrease of power output
  - others (valve points, must run/must-not run, ...)
- A set H of hydro (cascade) units
- For each $j \in H$
  - no energy cost (long-term cost of water incorporated as bounds)
  - inflows, maximum and minimum basin levels, cascade topology
  - maximum power output
  - others (nonlinear effects of water head, nonzero technical minima, cavitation points, pumping, ...)

The Hydro-Thermal Unit Commitment problem (3)

- Transmission constraints
  - simplest form: bus
  - actual topology: Direct Current (linear), Alternating Current (nonlinear)

The Hydro-Thermal Unit Commitment problem (3)

- Transmission constraints
  - simplest form: bus
  - actual topology: Direct Current (linear), Alternating Current (nonlinear)
- Reliability constraints
  - rotating reserve constraints
  - other (stochastic demand, ...)

The Hydro-Thermal Unit Commitment problem (3)

- Transmission constraints
  - simplest form: bus
  - actual topology: Direct Current (linear), Alternating Current (nonlinear)
- Reliability constraints
  - rotating reserve constraints
  - other (stochastic demand, ...)

Operate the available units over $\mathcal{T}$ to satisfy demand at minimal cost

The Hydro-Thermal Unit Commitment problem (3)

- Transmission constraints
  - simplest form: bus
  - actual topology: Direct Current (linear), Alternating Current (nonlinear)
- Reliability constraints
  - rotating reserve constraints
  - other (stochastic demand, ...)

Operate the available units over $\mathcal{T}$ to satisfy demand at minimal cost

- Typical of monopolistic regime

The Hydro-Thermal Unit Commitment problem (3)

- Transmission constraints
  - simplest form: bus
  - actual topology: Direct Current (linear), Alternating Current (nonlinear)
- Reliability constraints
  - rotating reserve constraints
  - other (stochastic demand, ...)

Operate the available units over $\mathcal{T}$ to satisfy demand at minimal cost

- Typical of monopolistic regime
- Useful in free markets
  - actual scheduling after market(s) clears
  - optimal bidding strategy

A MIQP Formulation

- Main variables:
  - $u^i_t \in \{0, 1\}$: ON/OFF state of thermal unit $i \in P$
  - $p^i_t \in \mathbb{R}_+$: power level of thermal unit $i \in P$
  - $q^j_t \in \mathbb{R}_+$: water discharge for hydro unit $j \in H(h)$ for cascade $h \in H$

[1] M.P. Nowak and W. Römisch "Stochastic Lagrangian Relaxation Applied to Power Scheduling in a Hydro-thermal System Under Uncertainty", *Annals of Operations Research*, 2000

A MIQP Formulation

- Main variables:
  - $u_t^i \in \{0,1\}$: ON/OFF state of thermal unit $i \in P$
  - $p_t^i \in \mathbb{R}_+$: power level of thermal unit $i \in P$
  - $q_t^j \in \mathbb{R}_+$: water discharge for hydro unit $j \in H(h)$ for cascade $h \in H$

- Objective function:

$$f(p,u) = \sum_{i \in P} c^i(p^i, u^i) = \sum_{i \in P} \Big(s^i(u^i) + \sum_{t \in \mathcal{T}} \big(f_t^i(p_t^i) + c_t^i u_t^i \big) \Big) \qquad (1)$$

  - convex energy cost, usually quadratic ($f_t^i(p_t^i) = a_t^i(p_t^i)^2 + b_t^i p_t^i$, $a_t^i > 0$)
  - time-dependent start-up costs $s^i(u^i)$ (only some extra constraints with nifty formulation[1])

[1] M.P. Nowak and W. Römisch "Stochastic Lagrangian Relaxation Applied to Power Scheduling in a Hydro-thermal System Under Uncertainty", *Annals of Operations Research*, 2000

A MIQP Formulation (2)

Thermal units:

- Maximum and minimum power output:

$$\bar{p}^i_{min} u^i_t \leq p^i_t \leq \bar{p}^i_{max} u^i_t \qquad t \in \mathcal{T} \tag{2}$$

A MIQP Formulation (2)

Thermal units:

- Maximum and minimum power output:

$$\bar{p}^i_{min} u^i_t \leq p^i_t \leq \bar{p}^i_{max} u^i_t \qquad t \in \mathcal{T} \qquad (2)$$

- Ramp-up constraints ($\Delta^i_+ =$ ramp-up threshold):

$$p^i_t \leq p^i_{t-1} + u^i_{t-1}\Delta^i_+ + (1 - u^i_{t-1})\bar{l}^i \qquad t \in \mathcal{T} \qquad (3)$$

A MIQP Formulation (2)

Thermal units:

- Maximum and minimum power output:

$$\bar{p}^i_{min} u^i_t \leq p^i_t \leq \bar{p}^i_{max} u^i_t \qquad t \in \mathcal{T} \tag{2}$$

- Ramp-up constraints ($\Delta^i_+ =$ ramp-up threshold):

$$p^i_t \leq p^i_{t-1} + u^i_{t-1}\Delta^i_+ + (1 - u^i_{t-1})\bar{l}^i \qquad t \in \mathcal{T} \tag{3}$$

- Ramp-down constraints ($\Delta^i_- =$ ramp-down threshold):

$$p^i_{t-1} \leq p^i_t + u^i_t\Delta^i_- + (1 - u^i_t)\bar{u}^i \qquad t \in \mathcal{T} \tag{4}$$

A MIQP Formulation (2)

Thermal units:

- Maximum and minimum power output:

$$\bar{p}^i_{min} u^i_t \leq p^i_t \leq \bar{p}^i_{max} u^i_t \qquad t \in \mathcal{T} \qquad (2)$$

- Ramp-up constraints (Δ^i_+ = ramp-up threshold):

$$p^i_t \leq p^i_{t-1} + u^i_{t-1}\Delta^i_+ + (1 - u^i_{t-1})\bar{l}^i \qquad t \in \mathcal{T} \qquad (3)$$

- Ramp-down constraints (Δ^i_- = ramp-down threshold):

$$p^i_{t-1} \leq p^i_t + u^i_t \Delta^i_- + (1 - u^i_t)\bar{u}^i \qquad t \in \mathcal{T} \qquad (4)$$

- Min up-time constraints (τ^i_+ = min up-time):

$$u^i_t \leq 1 - u^i_{r-1} + u^i_r \qquad t \in \mathcal{T},\ r \in [t - \tau^i_+, t-1] \qquad (5)$$

A MIQP Formulation (2)

Thermal units:

- Maximum and minimum power output:

$$\bar{p}^i_{min} u^i_t \leq p^i_t \leq \bar{p}^i_{max} u^i_t \qquad t \in \mathcal{T} \qquad (2)$$

- Ramp-up constraints (Δ^i_+ = ramp-up threshold):

$$p^i_t \leq p^i_{t-1} + u^i_{t-1}\Delta^i_+ + (1 - u^i_{t-1})\bar{l}^i \qquad t \in \mathcal{T} \qquad (3)$$

- Ramp-down constraints (Δ^i_- = ramp-down threshold):

$$p^i_{t-1} \leq p^i_t + u^i_t\Delta^i_- + (1 - u^i_t)\bar{u}^i \qquad t \in \mathcal{T} \qquad (4)$$

- Min up-time constraints (τ^i_+ = min up-time):

$$u^i_t \leq 1 - u^i_{r-1} + u^i_r \qquad t \in \mathcal{T},\ r \in [t - \tau^i_+, t-1] \qquad (5)$$

- Min down-time constraints (τ^i_- = min down-time):

$$u^i_t \geq 1 - u^i_{r-1} - u^i_r \qquad t \in \mathcal{T},\ r \in [t - \tau^i_-, t-1] \qquad (6)$$

A MIQP Formulation (3)

Hydro units:

A MIQP Formulation (3)

Hydro units:

- Maximum discharge:

$$0 \leq q_t^j \leq \bar{q}_{max}^j \qquad t \in \mathcal{T} \tag{7}$$

A MIQP Formulation (3)

Hydro units:

- Maximum discharge:

$$0 \leq q_t^j \leq \bar{q}_{max}^j \qquad t \in \mathcal{T} \qquad (7)$$

- Maximum and minimum reservoir volume:

$$\bar{v}_{min}^j \leq v_t^j \leq \bar{v}_{max}^j \qquad t \in \mathcal{T} \qquad (8)$$

A MIQP Formulation (3)

Hydro units:

- Maximum discharge:

$$0 \leq q_t^j \leq \bar{q}_{max}^j \qquad t \in \mathcal{T} \tag{7}$$

- Maximum and minimum reservoir volume:

$$\bar{v}_{min}^j \leq v_t^j \leq \bar{v}_{max}^j \qquad t \in \mathcal{T} \tag{8}$$

- Water conservation ($\bar{w}_t^j$ = inflow, w_t^j = spillage, t_{kj} = time delay):

$$v_t^j - v_{t-1}^j = \bar{w}_t^j - w_t^j - q_t^j + \sum_{k \in \mathcal{S}(j)} \left(q_{t-t_{kj}}^k + w_{t-t_{kj}}^k \right) \qquad t \in \mathcal{T} \tag{9}$$

A MIQP Formulation (3)

Hydro units:

- Maximum discharge:

$$0 \leq q^j_t \leq \bar{q}^j_{max} \qquad t \in \mathcal{T} \qquad (7)$$

- Maximum and minimum reservoir volume:

$$\bar{v}^j_{min} \leq v^j_t \leq \bar{v}^j_{max} \qquad t \in \mathcal{T} \qquad (8)$$

- Water conservation ($\bar{w}^j_t$ = inflow, w^j_t = spillage, t_{kj} = time delay):

$$v^j_t - v^j_{t-1} = \bar{w}^j_t - w^j_t - q^j_t + \sum_{k \in \mathcal{S}(j)} \left(q^k_{t-t_{kj}} + w^k_{t-t_{kj}} \right) \qquad t \in \mathcal{T} \quad (9)$$

System-wide constraints:

- Demand satisfaction (α^j = constant power-to-discharged water):

$$\sum_{i \in P} p^i_t + \sum_{h \in H} \sum_{j \in H(h)} \alpha^j q^j_t = \bar{d}_t \qquad t \in \mathcal{T} \qquad (10)$$

Results of the MIQP Formulation

Can you solve (1)—(10) (as it is) with `Cplex`? ...

Results of the MIQP Formulation

Can you solve (1)—(10) (as it is) with `Cplex`? . . . Not really

p	first	best	gap	unsolved
20	24	2229	0.29	
50	249	1491	0.22	
75	447	1514	0.10	
100	940	2327	0.13	
150	2348	2483	0.24	1
200	3600	3600	*	5

Results of the MIQP Formulation

Can you solve (1)—(10) (as it is) with `Cplex`? ... Not really

p	first	best	gap	unsolved
20	24	2229	0.29	
50	249	1491	0.22	
75	447	1514	0.10	
100	940	2327	0.13	
150	2348	2483	0.24	1
200	3600	3600	*	5

- For larger problems, no feasible solutions found in 1h

Results of the MIQP Formulation

Can you solve (1)—(10) (as it is) with `Cplex`? ... Not really

p	first	best	gap	unsolved
20	24	2229	0.29	
50	249	1491	0.22	
75	447	1514	0.10	
100	940	2327	0.13	
150	2348	2483	0.24	1
200	3600	3600	*	5

- For larger problems, no feasible solutions found in 1h
- Gap w.r.t. tight Lagrangian bound, inherent gap vastly worse

Results of the MIQP Formulation

Can you solve (1)—(10) (as it is) with `Cplex`? . . . Not really

p	first	best	gap	unsolved
20	24	2229	0.29	
50	249	1491	0.22	
75	447	1514	0.10	
100	940	2327	0.13	
150	2348	2483	0.24	1
200	3600	3600	*	5

- For larger problems, no feasible solutions found in 1h
- Gap w.r.t. tight Lagrangian bound, inherent gap vastly worse
- Randomly-generated, realistic ramp-constrained UC instances

`http://www.di.unipi.it/optimize/Data`

Lagrangian Relaxation

- Denote (2)—(6) as $\mathcal{U}^i$, (7)—(9) as $\mathcal{H}^h$
- Lagrangian Relaxation of demand constraints (10), multipliers λ
- The problem decomposes by unit:

$$\phi(\lambda) = \sum_{i \in P} \phi_i^1(\lambda) + \sum_{h \in H} \phi_h^2(\lambda) + \sum_{t \in \mathcal{T}} \lambda_t \bar{d}_t$$

$$\phi_i^1(\lambda) \;=\; \min \left\{ \; c^i(p^i, u^i) - \lambda p^i \;:\; (p^i, u^i) \in \mathcal{U}^i \; \right\} \tag{11}$$

$$\phi_h^2(\lambda) \;=\; \min \left\{ -\lambda \textstyle\sum_{j \in H(h)} \alpha^j q^j \;:\; [q^j]_{j \in H(h)} \in \mathcal{H}^h \right\} \tag{12}$$

[2] F. "Generalized Bundle Methods" *SIAM Journal on Optimization*, 2002

- Denote (2)—(6) as $\mathcal{U}^i$, (7)—(9) as $\mathcal{H}^h$
- Lagrangian Relaxation of demand constraints (10), multipliers λ
- The problem decomposes by unit:

$$\phi(\lambda) = \sum_{i \in P} \phi_i^1(\lambda) + \sum_{h \in H} \phi_h^2(\lambda) + \sum_{t \in \mathcal{T}} \lambda_t \bar{d}_t$$

$$\phi_i^1(\lambda) = \min \{ c^i(p^i, u^i) - \lambda p^i \ : \ (p^i, u^i) \in \mathcal{U}^i \} \tag{11}$$

$$\phi_h^2(\lambda) = \min \left\{ -\lambda \textstyle\sum_{j \in H(h)} \alpha^j q^j \ : \ [q^j]_{j \in H(h)} \in \mathcal{H}^h \right\} \tag{12}$$

- Lagrangian Dual:

$$\max \{ \phi(\lambda) \ : \ \lambda \in \mathbb{R}^n \} \tag{13}$$

($n = |\mathcal{T}|$) efficiently solvable e.g. by Bundle methods[2] ...

[2] F. "Generalized Bundle Methods" *SIAM Journal on Optimization*, 2002

Lagrangian Relaxation

- Denote (2)—(6) as $\mathcal{U}^i$, (7)—(9) as $\mathcal{H}^h$
- Lagrangian Relaxation of demand constraints (10), multipliers λ
- The problem decomposes by unit:

$$\phi(\lambda) = \sum_{i \in P} \phi_i^1(\lambda) + \sum_{h \in H} \phi_h^2(\lambda) + \sum_{t \in \mathcal{T}} \lambda_t \bar{d}_t$$

$$\phi_i^1(\lambda) \;=\; \min \left\{ \, c^i(p^i, u^i) - \lambda p^i \;:\; (p^i, u^i) \in \mathcal{U}^i \, \right\} \tag{11}$$

$$\phi_h^2(\lambda) \;=\; \min \left\{ -\lambda \textstyle\sum_{j \in H(h)} \alpha^j q^j \;:\; [q^j]_{j \in H(h)} \in \mathcal{H}^h \right\} \tag{12}$$

- Lagrangian Dual:

$$\max \{ \, \phi(\lambda) \;:\; \lambda \in \mathbb{R}^n \, \} \tag{13}$$

($n = |\mathcal{T}|$) efficiently solvable e.g. by Bundle methods[2] ...
provided $\phi(\lambda)$ is efficiently computable

[2] F. "Generalized Bundle Methods" *SIAM Journal on Optimization*, 2002

Solving the subproblems

- Hydro Single-Unit Subproblems (12): Network Flows algorithms

Solving the subproblems

- Hydro Single-Unit Subproblems (12): Network Flows algorithms
- Thermal Single-Unit Subproblems (11): Dynamic Programming ...

Solving the subproblems

- Hydro Single-Unit Subproblems (12): Network Flows algorithms
- Thermal Single-Unit Subproblems (11): Dynamic Programming . . .
  if there are no ramping constraints (3)–(4)

Solving the subproblems

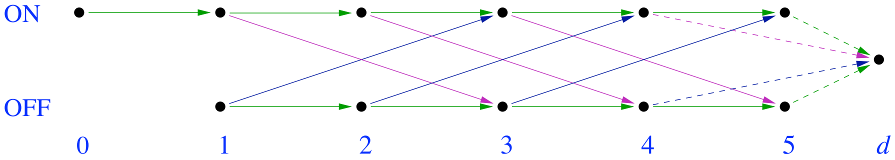
- Hydro Single-Unit Subproblems (12): Network Flows algorithms
- Thermal Single-Unit Subproblems (11): Dynamic Programming . . .
  if there are no ramping constraints (3)–(4)
- Trick: optimal power level (if unit on) computable a-priori:

$$\tilde{p}^i_t = \mathrm{argmin} \; \{ \; f^i_t(p) - \lambda_t p \; : \; p^{min}_i \leq p \leq p^{max}_i \; \}$$

Solving the subproblems

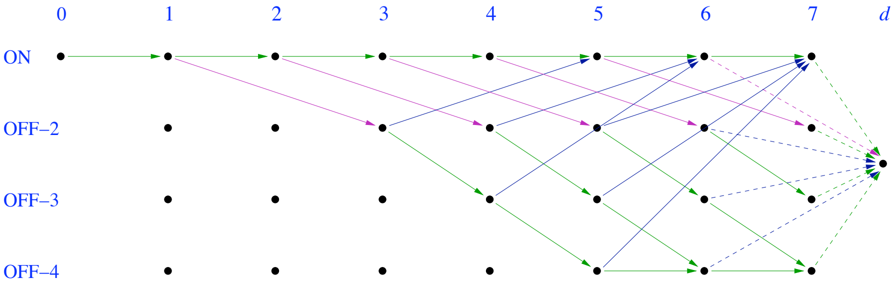
- Hydro Single-Unit Subproblems (12): Network Flows algorithms
- Thermal Single-Unit Subproblems (11): Dynamic Programming ...
  if there are no ramping constraints (3)–(4)
- Trick: optimal power level (if unit on) computable a-priori:

$$\tilde{p}^i_t = \text{argmin} \left\{ \, f^i_t(p) - \lambda_t p \; : \; p^{min}_i \leq p \leq p^{max}_i \, \right\}$$



- State-space graph $G = (V, A)$ for fixed start-up costs ($A = O(n)$):
  - start-up costs on $(t_{\text{OFF}}, t_{\text{ON}})$ arcs
  - cost $z_{it}(\lambda_t) = f^i_t(\tilde{p}^i_t) - \lambda_t \tilde{p}^i_t$ on t_{ON} nodes
  - set of initial arcs depending on initial conditions

Solving the (thermal) subproblems



- State-space graph for Time-dependent start-up costs:
  - Nodes OFF$-h$: unit is off, and has been for the last h hours
  - Time-dependent start-up costs on $(t_{\text{OFF}-h}, t_{\text{ON}})$ arcs
  - Cost $z_{it}(\lambda_t)$ on t_{ON} nodes as before
  - Complexity $O(nk)$, where k is maximum cooling time

Lagrangian Heuristic

- Solving (13) provides a lower bound on the original problem ...

[3] F. "About Lagrangian Methods in Integer Optimization" *Annals of Operations Research*, 2005

Lagrangian Heuristic

- Solving (13) provides a lower bound on the original problem ... but not only; also valuable primal information is generated [3]

- At every iteration:
  - Lagrangian multipliers $\bar{\lambda}$
  - primal solution $[\bar{p}, \bar{u}, \bar{q}]$ of (11)–(12), $\bar{u}$ integer
  - "convexified" primal solution $[\tilde{p}, \tilde{u}, \tilde{q}]$, almost feasible to (10)

- For fixed u, (2)—(10) is an easy convex quadratic program (called the Economic Dispatch (ED) problem)

- One can fix $u = \bar{u}$ and solve (ED); if it is feasible, a solution is obtained ...

[3] F. "About Lagrangian Methods in Integer Optimization" *Annals of Operations Research*, 2005

Lagrangian Heuristic

- Solving (13) provides a lower bound on the original problem ... but not only; also valuable primal information is generated [3]
- At every iteration:
  - Lagrangian multipliers $\bar{\lambda}$
  - primal solution $[\bar{p}, \bar{u}, \bar{q}]$ of (11)–(12), $\bar{u}$ integer
  - "convexified" primal solution $[\tilde{p}, \tilde{u}, \tilde{q}]$, almost feasible to (10)
- For fixed u, (2)—(10) is an easy convex quadratic program (called the Economic Dispatch (ED) problem)
- One can fix $u = \bar{u}$ and solve (ED); if it is feasible, a solution is obtained ... however, almost always $\bar{u}$ is undercommitted

[3] F. "About Lagrangian Methods in Integer Optimization" *Annals of Operations Research*, 2005

Lagrangian Heuristic (cont.d)

- Fix $q = \tilde{q}$, reduce demand $\tilde{d}_t = \bar{d}_t - \sum_{h \in H} \sum_{j \in H(h)} \alpha^j \tilde{q}_t^j$

Lagrangian Heuristic (cont.d)

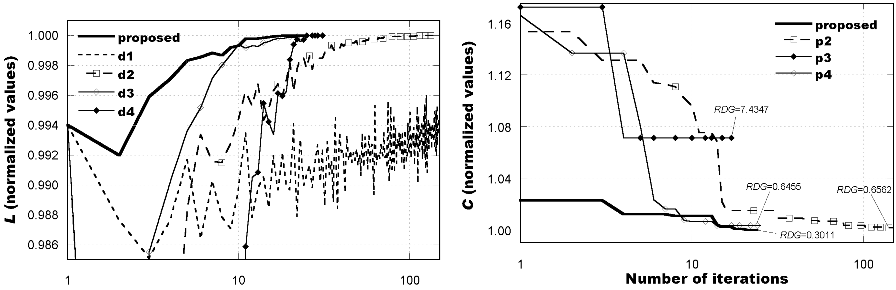
- Fix $q = \tilde{q}$, reduce demand $\tilde{d}_t = \bar{d}_t - \sum_{h \in H} \sum_{j \in H(h)} \alpha^j \tilde{q}^j_t$
- Greedy heuristic to find $\hat{u}_t$ feasible for residual demand $\tilde{d}_t$:
  - initialize $\hat{u} = \bar{u}$
  - for all time instants t, in increasing order
  - compute $\bar{u}^-_t = \sum_{i \in P} p^i_{min} \hat{u}^i_t \qquad \bar{u}^+_t = \sum_{i \in P} p^i_{max} \hat{u}^i_t$
    if $\tilde{d}_t > \bar{u}^+_t$ then turn on some units
    if $\tilde{d}_t < \bar{u}^-_t$ then turn off some units
    (check min up- and down-constraints from partial solution)

Lagrangian Heuristic (cont.d)

- Fix $q = \tilde{q}$, reduce demand $\tilde{d}_t = \bar{d}_t - \sum_{h \in H} \sum_{j \in H(h)} \alpha^j \tilde{q}^j_t$
- Greedy heuristic to find $\hat{u}_t$ feasible for residual demand $\tilde{d}_t$:
  - initialize $\hat{u} = \bar{u}$
  - for all time instants t, in increasing order
  - compute $\bar{u}^-_t = \sum_{i \in P} p^i_{min} \hat{u}^i_t \qquad \bar{u}^+_t = \sum_{i \in P} p^i_{max} \hat{u}^i_t$
    if $\tilde{d}_t > \bar{u}^+_t$ then turn on some units
    if $\tilde{d}_t < \bar{u}^-_t$ then turn off some units
    (check min up- and down-constraints from partial solution)
- Fix $u = \hat{u}$, solve (ED) to find $\hat{p}$, $\hat{q}$

Lagrangian Heuristic (cont.d)

- Fix $q = \tilde{q}$, reduce demand $\tilde{d}_t = \bar{d}_t - \sum_{h \in H} \sum_{j \in H(h)} \alpha^j \tilde{q}^j_t$
- Greedy heuristic to find $\hat{u}_t$ feasible for residual demand $\tilde{d}_t$:
  - initialize $\hat{u} = \bar{u}$
  - for all time instants t, in increasing order
  - compute $\bar{u}^-_t = \sum_{i \in P} p^i_{min} \hat{u}^i_t \qquad \bar{u}^+_t = \sum_{i \in P} p^i_{max} \hat{u}^i_t$
    if $\tilde{d}_t > \bar{u}^+_t$ then turn on some units
    if $\tilde{d}_t < \bar{u}^-_t$ then turn off some units
    (check min up- and down-constraints from partial solution)
- Fix $u = \hat{u}$, solve (ED) to find $\hat{p}$, $\hat{q}$
- Lagrangian information used:
  - $\tilde{q}$ to scale demand (modifying hydro schedule difficult)
  - $\bar{u}$ as the "backbone" of the feasible solution
  - $\tilde{u}$ and $\bar{\lambda}$ to define the order for turning on/off units



- Good dual convergence, good primal solution = small gap ($<< 1\%$)[4]
- Fast computing time (few minutes, AMPL code, 100+ units)
- All pieces need to fit together (dual convergence, primal solutions)

[4] A. Borghetti, F., F. Lacalandra, C.A. Nucci "Lagrangian Heuristics Based on Disaggregated Bundle Methods for Hydrothermal Unit Commitment", *IEEE Transactions on Power Systems*, 2003

Results (ramp constraints)

- The approach can be used even in presence of ramp constraints (lower bound valid, ramp constraints easily inserted in the (ED))
- Is it effective?

Results (ramp constraints)

- The approach can be used even in presence of ramp constraints (lower bound valid, ramp constraints easily inserted in the (ED))
- Is it effective? Not really

p	h	time	iter	sol	gap
20	0	6	202	1	11.30(3)
50	0	16	247	1	5.25 (3)
75	0	22	278	1	9.25
100	0	29	285	1	8.69
150	0	54	341	1	7.66
200	0	78	369	1	8.53
20	10	7	206	3	3.80
50	20	16	231	6	0.63
75	35	28	274	5	1.73
100	50	38	301	1	1.86
150	75	71	318	1	4.10 (1)
200	100	90	305	2	4.38

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p_t^i variables for different t together

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p_t^i variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p_t^i variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure huge state-space graph $\Rightarrow$ costly, approximate solution, invalid bound

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p_t^i variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure huge state-space graph ⇒ costly, approximate solution, invalid bound
  - "Lagrangianize" ramp rate constraints (possibly two-level methods)

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p_t^i variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure huge state-space graph ⇒ costly, approximate solution, invalid bound
  - "Lagrangianize" ramp rate constraints (possibly two-level methods) many multipliers ⇒ slow, weak bound, no feasible solutions

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p_t^i variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure
    huge state-space graph ⇒ costly, approximate solution, invalid bound
  - "Lagrangianize" ramp rate constraints (possibly two-level methods)
    many multipliers ⇒ slow, weak bound, no feasible solutions
  - using an off-the-shelf (MIQP) solver (works for general models)

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p_t^i variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure
    huge state-space graph ⇒ costly, approximate solution, invalid bound
  - "Lagrangianize" ramp rate constraints (possibly two-level methods)
    many multipliers ⇒ slow, weak bound, no feasible solutions
  - using an off-the-shelf (MIQP) solver (works for general models)
    impractical for large n (as we will see)

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p^i_t variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure
    huge state-space graph ⇒ costly, approximate solution, invalid bound
  - "Lagrangianize" ramp rate constraints (possibly two-level methods)
    many multipliers ⇒ slow, weak bound, no feasible solutions
  - using an off-the-shelf (MIQP) solver (works for general models)
    impractical for large n (as we will see)
  - Piecewise-linearizing the objective function[6]

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p^i_t variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure
    huge state-space graph $\Rightarrow$ costly, approximate solution, invalid bound
  - "Lagrangianize" ramp rate constraints (possibly two-level methods)
    many multipliers $\Rightarrow$ slow, weak bound, no feasible solutions
  - using an off-the-shelf (MIQP) solver (works for general models)
    impractical for large n (as we will see)
  - Piecewise-linearizing the objective function[6]
    approximate solution, cost growing as approximation improves

[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

Extension to the Ramp-Constrained Case

- With ramp constraints, the basic trick just don't work any longer: ramp rate limits link p^i_t variables for different t together
- Several attempts have been made to extend the original approach[5]:
  - discretizing the power space + using the standard DP procedure
    huge state-space graph ⇒ costly, approximate solution, invalid bound
  - "Lagrangianize" ramp rate constraints (possibly two-level methods)
    many multipliers ⇒ slow, weak bound, no feasible solutions
  - using an off-the-shelf (MIQP) solver (works for general models)
    impractical for large n (as we will see)
  - Piecewise-linearizing the objective function[6]
    approximate solution, cost growing as approximation improves
- The problem still looks easy, should be solvable . . . but how?

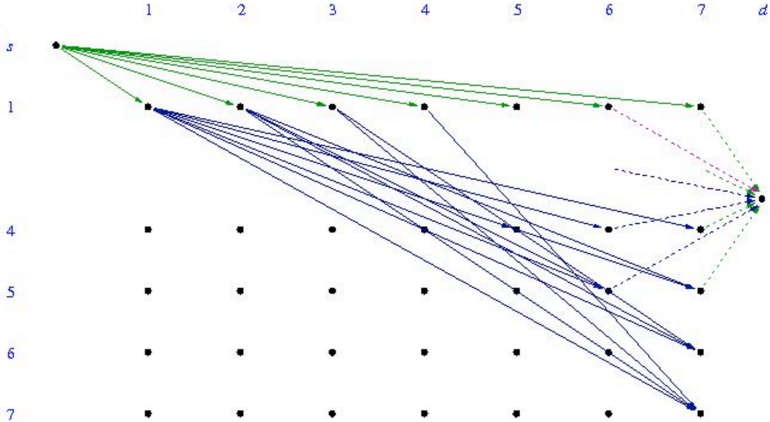
[5] F. Zhuang, F.D. Galiana "Towards a more rigorous and practical unit commitment by Lagrangian relaxation" *IEEE Transactions on Power Systems*, 1988

[6] W. Fan, X. Guan, Q. Zhai "A new method for unit commitment with ramping constraints" *Electric Power Systems Research*, 2002

A Dynamic Programming Algorithm for (1UC)

- First step: redefine the state-space graph
- Node (h, k) denotes unit ON from h to k (endpoints included)
- Not all nodes exist ($k - h + 1 \geq \tau^+$)
- Arcs between nodes (h, k) and (r, q) with $r \geq k + \tau^- + 1$
- Arcs from s to $(1, k)$ if unit ON at time 0
- Arcs from s to (h, k) with $h + \tau^0 - 1 \geq \tau^-$ if unit OFF at time 0
- Start-up cost on arcs, depending on the OFF time
- On nodes (h, k), optimal dispatching cost z^*_{hk} plus $(h - k + 1)c_i$
- Additional arcs with null cost from all nodes to d

The new space-state graph



- $O(n^4)$ arcs, but structured into levels $V_k = \{\ (h, k)\ :\ 1 \leq h \leq k\ \}$
- All nodes in V_k have the same set of adjacent nodes
- The cost of the arc between (h, k) and (r, q) only depends on k and r
- Increasing k, select best node of $V_k \Rightarrow O(n^3)$ if z^*_{hk} known

The Restricted Economic Dispatch Problem (ED_{hk})

- Convex problem with specially-structured linear constraints

$$
\begin{aligned}
z^*_{hk} = \min \ \textstyle\sum_{t=h}^{k} f^t(p_t) && (14)\\
p_{min} \leq p_t \leq p_{max} \qquad h \leq t \leq k && (15)\\
p_h \leq \bar{l} && (16)\\
p_{t+1} \leq p_t + \Delta_+ \qquad t = h, \ldots, k-1 && (17)\\
p_t \leq p_{t+1} + \Delta_- \qquad t = h, \ldots, k-1 && (18)\\
p_k \leq \bar{u} && (19)
\end{aligned}
$$

- Solving it should be easy . . .

The Restricted Economic Dispatch Problem (ED_{hk})

- Convex problem with specially-structured linear constraints

$$
\begin{array}{rlr}
z^*_{hk} = \min & \sum_{t=h}^{k} f^t(p_t) & (14) \\
p_{min} \leq p_t \leq p_{max} & h \leq t \leq k & (15) \\
p_h \leq \bar{l} & & (16) \\
p_{t+1} \leq p_t + \Delta_+ & t = h, \ldots, k-1 & (17) \\
p_t \leq p_{t+1} + \Delta_- & t = h, \ldots, k-1 & (18) \\
p_k \leq \bar{u} & & (19)
\end{array}
$$

- Solving it should be easy . . . but how, exactly?

The Restricted Economic Dispatch Problem (ED_{hk})

- Convex problem with specially-structured linear constraints

$$
\begin{aligned}
z^*_{hk} = \min \;\; & \textstyle\sum_{t=h}^{k} f^t(p_t) && (14)\\
p_{min} \leq p_t \leq p_{max} & \qquad h \leq t \leq k && (15)\\
p_h \leq \bar{l} & && (16)\\
p_{t+1} \leq p_t + \Delta_+ & \qquad t = h, \dots, k-1 && (17)\\
p_t \leq p_{t+1} + \Delta_- & \qquad t = h, \dots, k-1 && (18)\\
p_k \leq \bar{u} & && (19)
\end{aligned}
$$

- Solving it should be easy . . . but how, exactly?
- Simple idea: parametric problem on the power at time k

$$z_{hk}(\bar{p}) = \min \{ \textstyle\sum_{t=h}^{k} f^t(p_t) \; : \; (15)\,,\, (16)\,,\, (17)\,,\, (18)\,,\; p_k = \bar{p} \; \}$$

- Slightly simpler for $h = k$ (base case)

$$z_{hk}(\bar{p}) = \min\{ \; f^h(p_h) \; : \; (15)\,,\, (16)\,,\; p_h = \bar{p} \; \}$$

Solving (ED_{hk})

- Well-known general result: $z_{hk}(\bar{p})$ convex (a value function)

[7] F., C. Gentile "Solving Nonlinear Single-Unit Commitment Problems with Ramping Constraints" *Operations Research*, 2006

Solving (ED_{hk})

- Well-known general result: $z_{hk}(\bar{p})$ convex (a value function)
- Something more can be proven[7]: a compact representation exists

Proposition

$\exists v \leq 2(k-h)$, $l^k \leq m_0 \leq \ldots \leq m_{v+1} \leq u^k$ *s.t.* $dom(z_{hk}) = [m_0, m_{v+1}]$,

$$z_{hk}(p) = z^i(p) \qquad \text{if } p \in [m_i, m_{i+1}]$$

where each z^i is the sum of at most $k-h+1$ functions f^t for $t \in [h,k]$.

- Furthermore, $z_{h(k+1)}$ can be efficiently constructed given ...

[7] F., C. Gentile "Solving Nonlinear Single-Unit Commitment Problems with Ramping Constraints" *Operations Research*, 2006

Solving (ED_{hk})

- Well-known general result: $z_{hk}(\bar{p})$ convex (a value function)
- Something more can be proven[7]: a compact representation exists

Proposition

$\exists v \leq 2(k-h)$, $l^k \leq m_0 \leq \ldots \leq m_{v+1} \leq u^k$ s.t. $dom(z_{hk}) = [m_0, m_{v+1}]$,

$$z_{hk}(p) = z^i(p) \qquad \text{if } p \in [m_i, m_{i+1}]$$

where each z^i is the sum of at most $k-h+1$ functions f^t for $t \in [h,k]$.

- Furthermore, $z_{h(k+1)}$ can be efficiently constructed given . . .

  z_{hk} and $p^*_{hk} = \text{argmin}\{\ z_{hk}(p)\ :\ p \in [m_0, m_{v+1}]\ \}$ solving (ED_{hk})

[7] F., C. Gentile "Solving Nonlinear Single-Unit Commitment Problems with Ramping Constraints" *Operations Research*, 2006

Solving (ED_{hk})

- Well-known general result: $z_{hk}(\bar{p})$ convex (a value function)
- Something more can be proven[7]: a compact representation exists

Proposition

$\exists v \leq 2(k-h)$, $l^k \leq m_0 \leq \ldots \leq m_{v+1} \leq u^k$ s.t. $dom(z_{hk}) = [m_0, m_{v+1}]$,

$$z_{hk}(p) = z^i(p) \qquad \text{if } p \in [m_i, m_{i+1}]$$

where each z^i is the sum of at most $k-h+1$ functions f^t for $t \in [h,k]$.

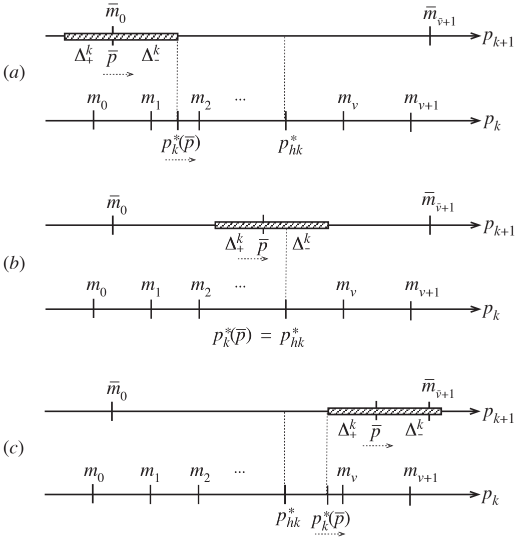
- Furthermore, $z_{h(k+1)}$ can be efficiently constructed given ...

  z_{hk} and $p^*_{hk} = \text{argmin}\{\ z_{hk}(p)\ :\ p \in [m_0, m_{v+1}]\ \}$ solving (ED_{hk})

  ... solving $(ED_{h(k+1)})$ (finding $p^*_{h(k+1)}$) while you are at that

[7] F., C. Gentile "Solving Nonlinear Single-Unit Commitment Problems with Ramping Constraints" *Operations Research*, 2006

Constructive proof



$$z_{hh}(\bar{p}) = f^h(\bar{p})$$

$$z_{h,k+1}(\bar{p}) = f^{k+1}(\bar{p}) + \left\{ \begin{array}{l} \min z_{hk}(p) \\ p \in [m_0, m_{v+1}] \\ p \in \bar{p} + [-\Delta^k_+, \Delta^k_-] \end{array} \right.$$

$$p^*_k(\bar{p}) = Proj(p^*_{hk}, \bar{p}+[-\Delta^k_+, \Delta^k_-])$$

$$z_{h,k+1}(\bar{p}) = f^{k+1}(\bar{p}) + z_{hk}(p^*_k(\bar{p}))$$

A Dynamic Programming Algorithm for (ED_{hk})

- Actually a Dynamic Programming Algorithm for (ED_{hk})

A Dynamic Programming Algorithm for (ED_{hk})

- Actually a Dynamic Programming Algorithm for (ED_{hk})
- Complexity depends on $min\{\ z_{hk}(p)\ :\ p \in [a, b]\ \}$ (ultimately on f^t)

A Dynamic Programming Algorithm for (ED_{hk})

- Actually a Dynamic Programming Algorithm for (ED_{hk})
- Complexity depends on $min\{\ z_{hk}(p)\ :\ p \in [a, b]\ \}$ (ultimately on f^t)
- $O(1)$ if f^t quadratic (sum of quadratic functions is quadratic)
- $O(k)$ to solve (ED_{hk}) having solved $(ED_{h(k-1)})$
- $O(k^2)$ to solve all (ED_{hk}) for $h \leq k$
- $O(n^3)$ for solving the overall (1UC) subproblem, after which

  $O(n)$ backward visit computes optimal dual solutions

A Dynamic Programming Algorithm for (ED_{hk})

- Actually a Dynamic Programming Algorithm for (ED_{hk})
- Complexity depends on $min\{\ z_{hk}(p)\ :\ p \in [a, b]\ \}$ (ultimately on f^t)
- $O(1)$ if f^t quadratic (sum of quadratic functions is quadratic)
- $O(k)$ to solve (ED_{hk}) having solved $(ED_{h(k-1)})$
- $O(k^2)$ to solve all (ED_{hk}) for $h \leq k$
- $O(n^3)$ for solving the overall (1UC) subproblem, after which

  $O(n)$ backward visit computes optimal dual solutions
- Easily extended to more complex situations:
  - Any fancy startup cost formula depending on (h, k) and (r, q)
  - Unit data changing every time period (e.g., external temperature)
  - Power level clock faster than ON/OFF one (e.g., 15m vs. 1h)

Results — (1UC) only

- 100 thermal units, 4 representative iterations of the Lagrangian
- Our DP algorithm vs. Cplex to solve (1UC) (time limit to 300 sec.)

		DP		CPLEX			
n	iter.	time	st.dev.	time	st.dev.	gap%	fail
24	1	.001	3e-3	0.05	0.05		0
	12	.002	4e-3	0.08	0.05		0
	16	.002	4e-3	0.08	0.05		0
	23	.002	4e-3	0.08	0.05		0
96	1	0.04	2e-3	10.74	41.99		1
	12	0.04	3e-3	17.57	50.93	0.06	2
	16	0.04	2e-3	32.64	76.87	0.02	6
	23	0.04	3e-3	32.21	76.12	0.03	6
168	1	0.20	6e-3	47.73	103.68	1.09	13
	12	0.20	6e-3	117.94	142.61	1.20	35
	16	0.20	5e-3	117.49	142.11	0.50	35
	23	0.20	6e-3	117.46	141.87	1.23	35

Results — the whole Lagrangian Heuristic

		RCDP				UDP				
p	*h*	time	iter	sol	gap	time	iter	sol	gap	Δlb
20	0	8	189	34	0.44	6	202	1	11.30(3)	2.49
50	0	17	195	33	0.26	16	247	1	5.25 (3)	1.48
75	0	30	206	33	0.38	22	278	1	9.25	2.38
100	0	46	213	21	0.48	29	285	1	8.69	2.21
150	0	72	277	23	0.20	54	341	1	7.66	2.31
200	0	134	317	67	0.06	78	369	1	8.53	2.46
20	10	16	162	159	0.22	7	206	3	3.80	1.50
50	20	41	165	146	0.07	16	231	6	0.63	1.19
75	35	89	209	166	0.02	28	274	5	1.73	1.19
100	50	135	218	143	0.04	38	301	1	1.86	1.27
150	75	222	223	164	0.01	71	318	1	4.10 (1)	1.20
200	100	353	244	192	0.05	90	305	2	4.38	1.25

- Actually quite good[8] (modern PC, C++ code)
- Can be improved with more sophisticated logic for greedy choice[9]

[8] F., C. Gentile, F. Lacalandra "Solving Unit Commitment Problems with General Ramp Contraints", *IJEPES*, 2008

[9] F., C. Gentile, F. Lacalandra "New Lagrangian Heuristics for Ramp-Constrained Unit Commitment Problems" *Proceedings ORMMES 2006*

The Day-ahead market

- Organized by the Market Operator (MO)
- Revolves around bids = (price, quantity) pairs
- Day-ahead market: for each of the 24 hours of tomorrow:
  - each generator submits to the MO selling bids (sp_j, sq_j), $j \in S$
  - each buyer submits to the MO buying bids (bp_i, bq_i), $i \in B$
  - the MO solves the Market Clearing Problem

$$\max \sum_{i \in B} bp_i b_i - \sum_{j \in S} sp_j s_j \tag{20}$$

$$0 \leq b_i \leq bq_i \qquad i \in B \tag{21}$$

$$0 \leq s_j \leq sq_j \qquad j \in S \tag{22}$$

$$\sum_{i \in B} b_i = \sum_{j \in S} s_j \tag{23}$$

The (dual) Market Clearing Problem

...or equivalently its dual

$$\min \sum_{i \in B} bq_i \mu_i \; + \; \sum_{j \in S} sq_j \eta_j \tag{24}$$

$$\mu_i + \pi \geq bp_i \qquad \mu_i \geq 0 \qquad i \in B \tag{25}$$

$$\eta_j - \pi \geq -sp_j \qquad \eta_j \geq 0 \qquad j \in S \tag{26}$$

The (dual) Market Clearing Problem

... or equivalently its dual

$$\min \sum_{i \in B} bq_i \mu_i + \sum_{j \in S} sq_j \eta_j \tag{24}$$

$$\mu_i + \pi \geq bp_i \qquad \mu_i \geq 0 \qquad i \in B \tag{25}$$

$$\eta_j - \pi \geq -sp_j \qquad \eta_j \geq 0 \qquad j \in S \tag{26}$$

which reads

$$\min_{\pi} \sum_{i \in B} bq_i \max\{bp_i - \pi, 0\} + \sum_{j \in S} sq_j \max\{\pi - sp_j, 0\} \tag{27}$$

The (dual) Market Clearing Problem

...or equivalently its dual

$$\min \sum_{i \in B} bq_i \mu_i \ + \sum_{j \in S} sq_j \eta_j \tag{24}$$

$$\mu_i + \pi \geq bp_i \qquad \mu_i \geq 0 \qquad i \in B \tag{25}$$

$$\eta_j - \pi \geq -sp_j \qquad \eta_j \geq 0 \qquad j \in S \tag{26}$$

which reads

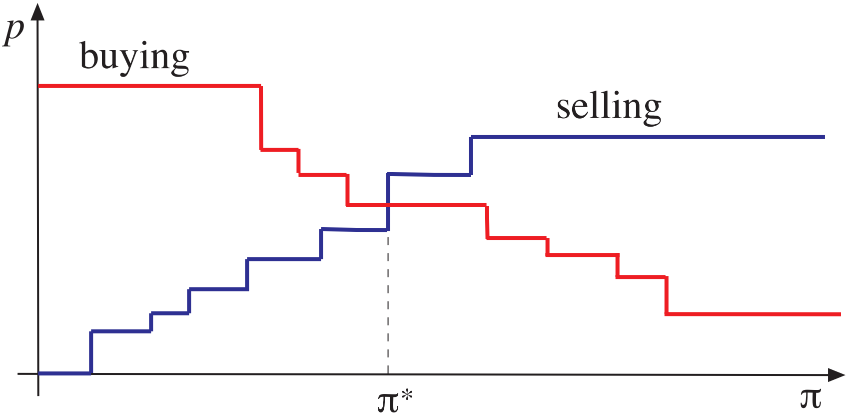
$$\min_{\pi} \sum_{i \in B} bq_i \max\{bp_i - \pi, 0\} \ + \sum_{j \in S} sq_j \max\{\pi - sp_j, 0\} \tag{27}$$

- $\pi^* =$ market clearing price
- Complementary slackness $\Rightarrow$

$$sp_j > \pi^* \Rightarrow s_j = sq_j$$

$$bp_i < \pi^* \Rightarrow b_i = bq_i$$

Graphical interpretation



The "X" marks the spot . . .

Complications in the Electrical Market

- Minor complications:
  - anelastic demand: just a fixed RHS in (23)

$$\bar{b} + \sum_{i \in B} b_i \ = \ \sum_{j \in S} s_j$$

    but this may cause $\pi^* = +\infty$ (need a price cap)

Complications in the Electrical Market

- Minor complications:
  - anelastic demand: just a fixed RHS in (23)

$$\bar{b} + \sum_{i \in B} b_i \;=\; \sum_{j \in S} s_j$$

    but this may cause $\pi^* = +\infty$ (need a price cap)

  - network constraints (DC version): $\mathcal{K}$ zones, $\mathcal{L}$ link between zones

$$m_l \leq \sum_{k \in \mathcal{K}} S_l^k \Big(\sum_{i \in I(k)} b_i - \sum_{j \in J(k)} s_j \Big) \leq M_l \qquad l \in \mathcal{L}$$

    - m_l and M_l: maximum and minimum current on link l
    - $I(k)/J(k)$: buying/selling bids on zone k
    - S_l^k: sensitivity of link l to injection in zone k

    $\Rightarrow$ zonal prices π_k^*

Complications in the Electrical Market

- Minor complications:
  - anelastic demand: just a fixed RHS in (23)

    $$\bar{b} + \sum_{i \in B} b_i \; = \; \sum_{j \in S} s_j$$

    but this may cause $\pi^* = +\infty$ (need a price cap)
  - network constraints (DC version): $\mathcal{K}$ zones, $\mathcal{L}$ link between zones

    $$m_l \leq \sum_{k \in \mathcal{K}} S_l^k \Big(\sum_{i \in I(k)} b_i - \sum_{j \in J(k)} s_j \Big) \leq M_l \qquad l \in \mathcal{L}$$

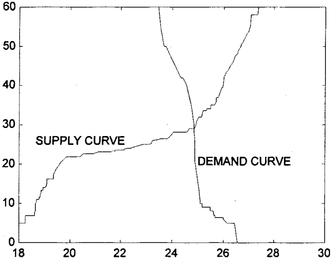
    - m_l and M_l: maximum and minimum current on link l
    - $I(k)/J(k)$: buying/selling bids on zone k
    - S_l^k: sensitivity of link l to injection in zone k

    $\Rightarrow$ zonal prices π_k^*

- Major complications:
  - AC network constraints (highly nonlinear, nonconvex)
  - PUN: unique buying price for all zones (an ugly mess)

Unit Commitment in the Electrical Market

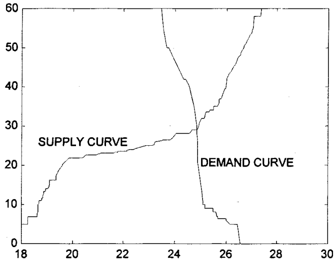
- Major simplifying assumptions:
  1. all demand is anelastic
  2. no network constraints
  3. competitors' supply curve known (estimate from past data works)



[10] A. Borghetti, F., F. Lacalandra, C.A. Nucci, P. Pelacchi "Using of a Cost-based Unit Commitment Algorithm to Assist Bidding Strategy Decisions" *Proceedings IEEE 2003 Powerteck Bologna Conference*, 2003

Unit Commitment in the Electrical Market

- Major simplifying assumptions:
  1. all demand is anelastic
  2. no network constraints
  3. competitors' supply curve known (estimate from past data works)



- Optimal bidding strategy[10]: modify model as

$$\max \quad \sum_{t\in\mathcal{T}} \mathcal{I}_t(po_t)(\bar{d}_t - po_t) - \sum_{i\in P} c^i(p^i, u^i)$$

$$\vdots \qquad \vdots$$

$$\sum_{i\in P} p^i_t + \sum_{h\in H} \sum_{j\in H(h)} \alpha^j q^j_t + po_t = \bar{d}_t \quad t \in \mathcal{T} \qquad (28)$$

where $\mathcal{I}_t$ = inverse of (estimate) competitors' supply function

[10] A. Borghetti, F., F. Lacalandra, C.A. Nucci, P. Pelacchi "Using of a Cost-based Unit Commitment Algorithm to Assist Bidding Strategy Decisions" *Proceedings IEEE 2003 Powerteck Bologna Conference*, 2003

Lagrangian Approach to Optimal Bidding Problem

- Relax (28) $\Rightarrow$ (UC) $+$ one "competitors' problem" for each $t \in \mathcal{T}$
  highly nonconvex but univariate

- $\mathcal{I}_t(po_t)$ piecewise-linear, increasing $\Rightarrow$
  $\mathcal{I}_t(po_t)(\bar{d}_t - po_t)$ piecewise-quadratic, concave $\Rightarrow$ easy in practice

Lagrangian Approach to Optimal Bidding Problem

- Relax (28) $\Rightarrow$ (UC) + one "competitors' problem" for each $t \in \mathcal{T}$
  highly nonconvex but univariate
- $\mathcal{I}_t(po_t)$ piecewise-linear, increasing $\Rightarrow$
  $\mathcal{I}_t(po_t)(\bar{d}_t - po_t)$ piecewise-quadratic, concave $\Rightarrow$ easy in practice
- No problem with lower bound, no problem with UC heuristic ...
  but dire problems with (ED) (large-scale, highly nonconvex)
- Trick: (ED) easy if po_t kept in the neighborhood of $\tilde{po}_t$ where
  $\mathcal{I}_t(po_t)(\bar{d}_t - po_t)$ quadratic, concave

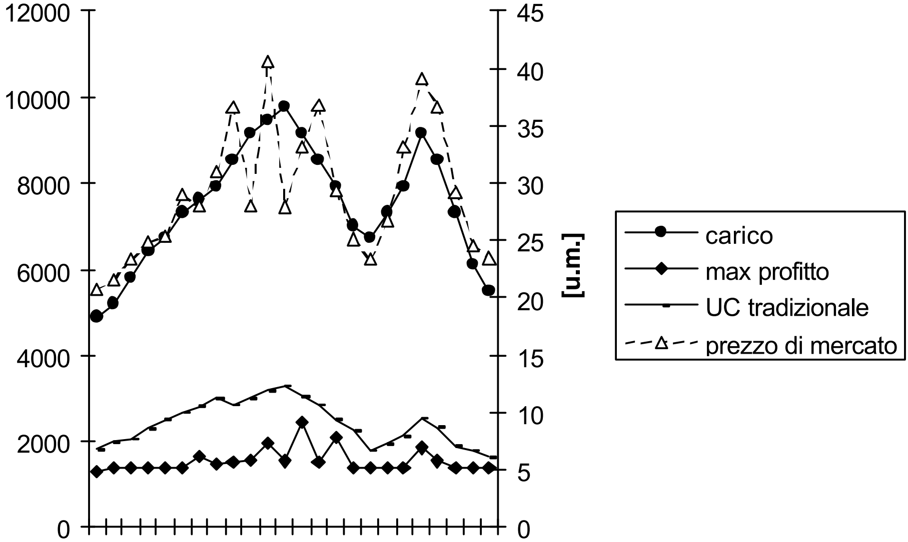
Lagrangian Approach to Optimal Bidding Problem

- Relax (28) $\Rightarrow$ (UC) + one "competitors' problem" for each $t \in \mathcal{T}$
  highly nonconvex but univariate
- $\mathcal{I}_t(po_t)$ piecewise-linear, increasing $\Rightarrow$
  $\mathcal{I}_t(po_t)(\bar{d}_t - po_t)$ piecewise-quadratic, concave $\Rightarrow$ easy in practice
- No problem with lower bound, no problem with UC heuristic ...
  but dire problems with (ED) (large-scale, highly nonconvex)
- Trick: (ED) easy if po_t kept in the neighborhood of $\tilde{po}_t$ where
  $\mathcal{I}_t(po_t)(\bar{d}_t - po_t)$ quadratic, concave
- As fast as (UC), good gaps ($\approx 0.5\%$) despite higher nonconvexity
- Relaxing (1) possible (non entirely trivial, work in progress)
- Relaxing (2) hard: strategic bidding with zones (work in progress)

Lagrangian Approach to Optimal Bidding Problem

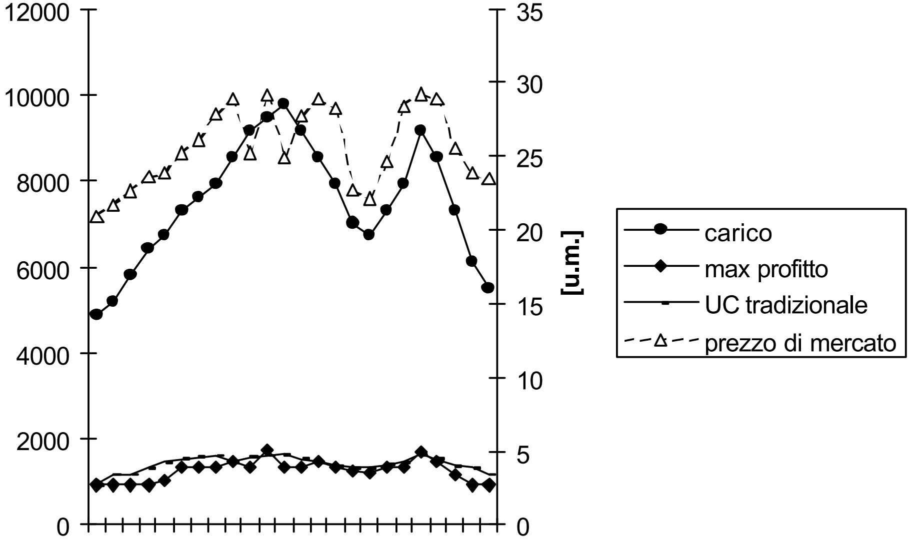
- Relax (28) $\Rightarrow$ (UC) + one "competitors' problem" for each $t \in \mathcal{T}$ highly nonconvex but univariate
- $\mathcal{I}_t(po_t)$ piecewise-linear, increasing $\Rightarrow$ $\mathcal{I}_t(po_t)(\bar{d}_t - po_t)$ piecewise-quadratic, concave $\Rightarrow$ easy in practice
- No problem with lower bound, no problem with UC heuristic ... but dire problems with (ED) (large-scale, highly nonconvex)
- Trick: (ED) easy if po_t kept in the neighborhood of $\tilde{po}_t$ where $\mathcal{I}_t(po_t)(\bar{d}_t - po_t)$ quadratic, concave
- As fast as (UC), good gaps ($\approx 0.5\%$) despite higher nonconvexity
- Relaxing (1) possible (non entirely trivial, work in progress)
- Relaxing (2) hard: strategic bidding with zones (work in progress)
- Some artificial (but realistic) results:

Results (producer has 35% of power)



If you are large, you can game the market

Results (producer has 15% of power)



If you are small, the market rules

Lagrangian Approaches: the Good and the Bad

- The Good:
  - efficient, effective, elegant
  - allow to incorporate fancy constraints, even difficult to model
  - decomposition + specialized algorithms = scale to very large size

[11] L. Dubost, R. Gonzalez, C. Lemaréchal "A Primal-proximal Heuristic Applied to the French Unit Commitment Problem" *Mathematical Programming*, 2005

Lagrangian Approaches: the Good and the Bad

- The Good:
  - efficient, effective, elegant
  - allow to incorporate fancy constraints, even difficult to model
  - decomposition + specialized algorithms = scale to very large size
- The Bad:
  - heuristics need be changed whenever the model changes (not so bad[11])
  - subproblems need be changed whenever the model changes (bad, especially if it takes 20 years)
  - in general, requires continuous work from OR specialists

[11] L. Dubost, R. Gonzalez, C. Lemaréchal "A Primal-proximal Heuristic Applied to the French Unit Commitment Problem" *Mathematical Programming*, 2005

Lagrangian Approaches: the Good and the Bad

- The Good:
  - efficient, effective, elegant
  - allow to incorporate fancy constraints, even difficult to model
  - decomposition + specialized algorithms = scale to very large size
- The Bad:
  - heuristics need be changed whenever the model changes (not so bad[11])
  - subproblems need be changed whenever the model changes (bad, especially if it takes 20 years)
  - in general, requires continuous work from OR specialists

$$\Downarrow$$

The Ugly: very hard to sell in a real-world environment

[11] L. Dubost, R. Gonzalez, C. Lemaréchal "A Primal-proximal Heuristic Applied to the French Unit Commitment Problem" *Mathematical Programming*, 2005

Lagrangian Approaches: the Good and the Bad

- The Good:
  - efficient, effective, elegant
  - allow to incorporate fancy constraints, even difficult to model
  - decomposition + specialized algorithms = scale to very large size
- The Bad:
  - heuristics need be changed whenever the model changes (not so bad[11])
  - subproblems need be changed whenever the model changes (bad, especially if it takes 20 years)
  - in general, requires continuous work from OR specialists

$$\Downarrow$$

The Ugly: very hard to sell in a real-world environment

- What are the alternatives?

[11] L. Dubost, R. Gonzalez, C. Lemaréchal "A Primal-proximal Heuristic Applied to the French Unit Commitment Problem" *Mathematical Programming*, 2005

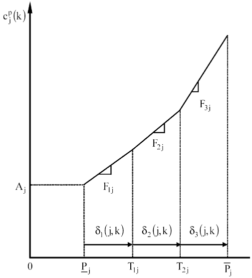
A MILP Formulation

- May the problem be the Quadratic part? If so, piecewise-linearize f[12]

[12] M. Carrión, J.M. Arroyo "A Computationally Efficient Mixed-integer Linear Formulation for the Thermal Unit Commitment Problem" *IEEE Transactions on Power Systems*, 2006

A MILP Formulation

- May the problem be the Quadratic part? If so, piecewise-linearize f[12]



- k new variables (i, t fixed), redefine p

$$
\begin{aligned}
& p = \sum_{l=1}^{k} \delta_l + \bar{p}_{min} u \\
& 0 \leq \delta_l \leq \bar{p}^l - \bar{p}^{l-1} \qquad l = 1, \ldots, k
\end{aligned}
\tag{29}
$$

- cost coefficient of u set to $f(\bar{p}_{min})$

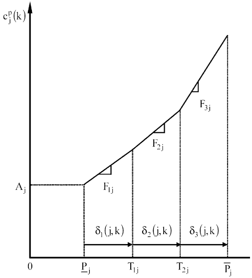
- cost coefficient of each δ_l set to

$$
F_l = \frac{f(\bar{p}^l) - f(\bar{p}^{l-1})}{\bar{p}^l - \bar{p}^{l-1}} = a(\bar{p}^l + \bar{p}^{l-1}) + b
$$

[12] M. Carrión, J.M. Arroyo "A Computationally Efficient Mixed-integer Linear Formulation for the Thermal Unit Commitment Problem" *IEEE Transactions on Power Systems*, 2006

A MILP Formulation

- May the problem be the Quadratic part? If so, piecewise-linearize f[12]



- k new variables (i, t fixed), redefine p

$$
\begin{aligned}
& p = \textstyle\sum_{l=1}^{k} \delta_l + \bar{p}_{min} u \\
& 0 \leq \delta_l \leq \bar{p}^l - \bar{p}^{l-1} \qquad l = 1, \dots, k
\end{aligned}
\tag{29}
$$

- cost coefficient of u set to $f(\bar{p}_{min})$

- cost coefficient of each δ_l set to

$$F_l = \frac{f(\bar{p}^l) - f(\bar{p}^{l-1})}{\bar{p}^l - \bar{p}^{l-1}} = a(\bar{p}^l + \bar{p}^{l-1}) + b$$

- Should this work? On the outset, I don't see why ...

[12] M. Carrión, J.M. Arroyo "A Computationally Efficient Mixed-integer Linear Formulation for the Thermal Unit Commitment Problem" *IEEE Transactions on Power Systems*, 2006

Results of the MILP Formulation

...but it does, big times!

Results of the MILP Formulation

. . . but it does, big times!

	MIQP			MILP				
p	first	best	gap	time	gap	ftime	fgap	nodes
20	24	2229	0.29	3.72	0.36		1.00	0
50	249	1491	0.22	21.93	0.21	15.98	0.36	0
75	447	1514	0.10	56.31	0.20	47.08	1.62	10
100	940	2327	0.13	94.09	0.17	69.75	2.18	16
150	2348	2483	0.24(1)	218.69	0.12	177.35	6.58	16
200	3600	3600	* (5)	267.78	0.09	247.12	1.85	6

Results of the MILP Formulation

. . . but it does, big times!

	MIQP			MILP				
p	first	best	gap	time	gap	ftime	fgap	nodes
20	24	2229	0.29	3.72	0.36		1.00	0
50	249	1491	0.22	21.93	0.21	15.98	0.36	0
75	447	1514	0.10	56.31	0.20	47.08	1.62	10
100	940	2327	0.13	94.09	0.17	69.75	2.18	16
150	2348	2483	0.24(1)	218.69	0.12	177.35	6.58	16
200	3600	3600	* (5)	267.78	0.09	247.12	1.85	6

- Stopping tolerance at 0.5% (and invalid lower bound)

Results of the MILP Formulation

... but it does, big times!

	MIQP			MILP				
p	first	best	gap	time	gap	ftime	fgap	nodes
20	24	2229	0.29	3.72	0.36		1.00	0
50	249	1491	0.22	21.93	0.21	15.98	0.36	0
75	447	1514	0.10	56.31	0.20	47.08	1.62	10
100	940	2327	0.13	94.09	0.17	69.75	2.18	16
150	2348	2483	0.24(1)	218.69	0.12	177.35	6.58	16
200	3600	3600	* (5)	267.78	0.09	247.12	1.85	6

- Stopping tolerance at 0.5% (and invalid lower bound)
- Again, inherent gap vastly worse (and invalid anyway)

Results of the MILP Formulation

...but it does, big times!

	MIQP			MILP				
p	first	best	gap	time	gap	ftime	fgap	nodes
20	24	2229	0.29	3.72	0.36		1.00	0
50	249	1491	0.22	21.93	0.21	15.98	0.36	0
75	447	1514	0.10	56.31	0.20	47.08	1.62	10
100	940	2327	0.13	94.09	0.17	69.75	2.18	16
150	2348	2483	0.24(1)	218.69	0.12	177.35	6.58	16
200	3600	3600	* (5)	267.78	0.09	247.12	1.85	6

- Stopping tolerance at 0.5% (and invalid lower bound)
- Again, inherent gap vastly worse (and invalid anyway)
- All the difference is in the heuristic

Comparing MILP and LR

		RCDP			`Cplex MILP`					
p	h	time	gap	iter	time	gap	ftime	fgap	nodes	LPs
10	0	0.75	0.99	187	0.95	0.33		1.18	0	23
20	0	1.83	0.46	189	3.72	0.36		1.00	0	23
50	0	4.84	0.28	195	21.93	0.21	15.98	0.36	0	25
75	0	9.41	0.34	206	56.31	0.20	47.08	1.62	10	59
100	0	14.74	0.33	213	94.09	0.17	69.75	2.18	16	76
150	0	21.20	0.17	277	218.69	0.12	177.35	6.58	16	115
200	0	34.80	0.09	317	267.78	0.09	247.12	1.85	6	87
20	10	1.76	0.39	170	93.53	0.21		0.59	140	258
50	20	6.36	0.06	160	17.98	0.06	17.98	0.06	0	60
75	35	15.01	0.04	198	96.86	0.11	96.86	0.11	170	300
100	50	24.74	0.04	209	130.86	0.06	130.86	0.06	180	266
150	75	37.41	0.02	189	467.62	0.06	467.62	0.06	300	554
200	100	50.91	0.01	175	427.71	0.05	427.71	0.05	205	321

- Faster version of RDCP (better (ED) solver)
- Overall, Cplex primal heuristic impressively effective

Perspective Cuts

- Convex function f, Mixed-Integer NonLinear Program fragment

$$\min \left\{ \, f(p) + cu \; : \; Ap \leq bu \; , \; u \in \{0, 1\} \, \right\} \tag{30}$$

$p \in \mathcal{P} = \{p \in \mathbb{R}^n : Ap \leq b\}$, $\{p : Ap \leq 0\} = \{0\}$ (think (2))

[13] F. and C. Gentile "Perspective Cuts for a Class of Convex 0-1 Mixed Integer Programs", *Mathematical Programming*, 2006

Perspective Cuts

- Convex function f, Mixed-Integer NonLinear Program fragment

$$\min \{ \; f(p) + cu \;\; : \;\; Ap \leq bu \; , \;\; u \in \{0, 1\} \; \} \tag{30}$$

$p \in \mathcal{P} = \{p \in \mathbb{R}^n : Ap \leq b\}$, $\{p : Ap \leq 0\} = \{0\}$ (think (2))

- Equivalently, minimize the nonconvex function

$$f(p, u) = \begin{cases} 0 & \text{if } u = 0 \text{ and } p = 0 \\ f(p) + c & \text{if } u = 1 \text{ and } Ap \leq b \\ +\infty & \text{otherwise} \end{cases} \tag{31}$$

[13] F. and C. Gentile "Perspective Cuts for a Class of Convex 0-1 Mixed Integer Programs", *Mathematical Programming*, 2006

Perspective Cuts

- Convex function f, Mixed-Integer NonLinear Program fragment

$$\min \left\{ \ f(p) + cu \ : \ Ap \leq bu \ , \ u \in \{0,1\} \ \right\} \tag{30}$$

$p \in \mathcal{P} = \{ p \in \mathbb{R}^n : Ap \leq b \}$, $\{ p : Ap \leq 0 \} = \{0\}$ (think (2))

- Equivalently, minimize the nonconvex function

$$f(p,u) = \begin{cases} 0 & \text{if } u = 0 \text{ and } p = 0 \\ f(p) + c & \text{if } u = 1 \text{ and } Ap \leq b \\ +\infty & \text{otherwise} \end{cases} \tag{31}$$

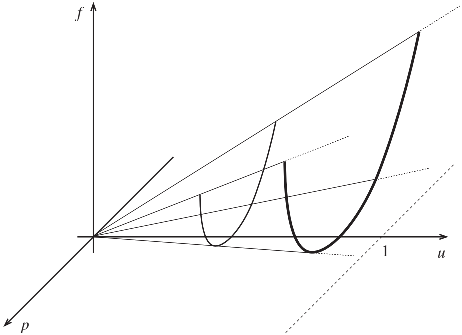
- Best possible convex relaxation of (30): use the convex envelope[13]

$$\overline{co} f(p,u) = \begin{cases} 0 & \text{if } p = 0 \text{ and } u = 0, \\ uf(p/u) + cu & \text{if } Ap \leq bu, \ \ u \in (0,1], \\ +\infty & \text{otherwise.} \end{cases} \tag{32}$$

(convex function minorizing $f(p,u)$ with smallest possible epigraph)

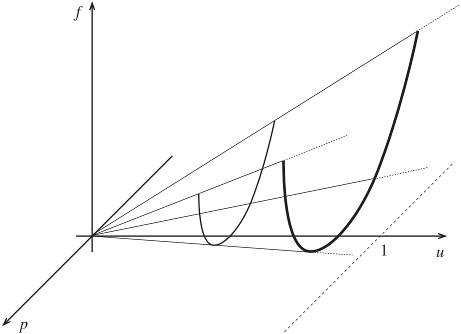
[13] F. and C. Gentile "Perspective Cuts for a Class of Convex 0-1 Mixed Integer Programs", *Mathematical Programming*, 2006

Perspective Cuts (2)



- Related with well-known perspective function of f
- $g(p, u) = u\, f(p/u)$
- despite the look, convex for $u > 0$ if f is

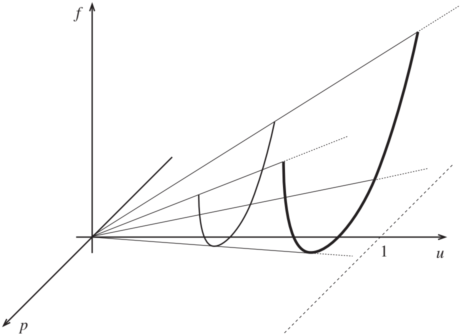
Perspective Cuts (2)



- Related with well-known perspective function of f
- $g(p, u) = u\, f(p/u)$
- despite the look, convex for $u > 0$ if f is

- Interesting examples:
  - linear: $f(p) = bp \Rightarrow \overline{co}f(p, u) = bp + cu$ (nothing happens!)

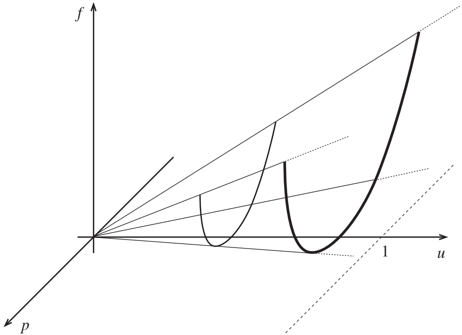
Perspective Cuts (2)



- Related with well-known perspective function of f
- $g(p, u) = u\, f(p/u)$
- despite the look, convex for $u > 0$ if f is

- Interesting examples:
  - linear: $f(p) = bp \Rightarrow \overline{co}f(p, u) = bp + cu$ (nothing happens!)
  - quadratic: $f(p) = ap^2 + bp \Rightarrow \overline{co}f(p, u) = ap^2/u + bp + cu$
    better than continuous relaxation $ap^2 + bp$ $(u \leq 1)$

Perspective Cuts (2)



- Related with well-known perspective function of f
- $g(p, u) = u\, f(p/u)$
- despite the look, convex for $u > 0$ if f is

- Interesting examples:
  - linear: $f(p) = bp \Rightarrow \overline{co}f(p, u) = bp + cu$ (nothing happens!)
  - quadratic: $f(p) = ap^2 + bp \Rightarrow \overline{co}f(p, u) = ap^2/u + bp + cu$ better than continuous relaxation $ap^2 + bp$ $(u \leq 1)$
  - ... but very nonlinear

Perspective Cuts (3)

- But every convex function is the supremum of its affine minorants

Perspective Cuts (3)

- But every convex function is the supremum of its affine minorants
- $(v, p, u) \in epi\ \overline{co} f \iff Ap \leq bu$, $u \in [0, 1]$, and $\forall \bar{p} \in \mathcal{P}$

$$v \geq f(\bar{p}) + c + [s\ ,\ c + f(\bar{p}) - s\bar{p}] \left[\begin{array}{c} p - \bar{p} \\ u - 1 \end{array} \right] \qquad \forall s \in \partial f(\bar{p}) \quad (33)$$

(infinitely many inequalities, at least one for each $\bar{p} \in \mathcal{P}$)

Perspective Cuts (3)

- But every convex function is the supremum of its affine minorants
- $(v, p, u) \in epi\ \overline{co}f \iff Ap \leq bu,\ u \in [0, 1]$, and $\forall \bar{p} \in \mathcal{P}$

$$v \geq f(\bar{p}) + c + [s\ ,\ c + f(\bar{p}) - s\bar{p}] \begin{bmatrix} p - \bar{p} \\ u - 1 \end{bmatrix} \qquad \forall s \in \partial f(\bar{p}) \quad (33)$$

(infinitely many inequalities, at least one for each $\bar{p} \in \mathcal{P}$)

- The quadratic case:

$$v \geq (2a\bar{p} + b)p + (c - a\bar{p}^2)u \qquad (34)$$

Perspective Cuts (3)

- But every convex function is the supremum of its affine minorants
- $(v, p, u) \in epi\ \overline{co}f \iff Ap \leq bu,\ u \in [0,1]$, and $\forall \bar{p} \in \mathcal{P}$

$$v \geq f(\bar{p}) + c + [s\ ,\ c + f(\bar{p}) - s\bar{p}] \begin{bmatrix} p - \bar{p} \\ u - 1 \end{bmatrix} \qquad \forall s \in \partial f(\bar{p}) \quad (33)$$

(infinitely many inequalities, at least one for each $\bar{p} \in \mathcal{P}$)

- The quadratic case:

$$v \geq (2a\bar{p} + b)p + (c - a\bar{p}^2)u \quad (34)$$

- Can implement a Branch & Cut with cuts on the objective function (somewhat tricky)

Alternative MILP (Approximated) formulation

- Replace (1) with

$$\sum_{t \in \mathcal{T}} \sum_{i \in P} v_t^i$$

Alternative MILP (Approximated) formulation

- Replace (1) with

$$\sum_{t \in \mathcal{T}} \sum_{i \in P} v_t^i$$

- Add k cuts (34) for some $\bar{p}_t^{i,h} \in [\bar{p}_{min}^i, \bar{p}_{max}^i]$, $h = 1, \ldots, k$

Alternative MILP (Approximated) formulation

- Replace (1) with

$$\sum_{t \in \mathcal{T}} \sum_{i \in P} v_t^i$$

- Add k cuts (34) for some $\bar{p}_t^{i,h} \in [\bar{p}_{min}^i, \bar{p}_{max}^i]$, $h = 1, \ldots, k$

- $|P||\mathcal{T}|$ more variables and $k|P||\mathcal{T}|$ more constraints, vs $k|P||\mathcal{T}|$ more variables and $(2k+1)|P||\mathcal{T}|$ more constraints

Alternative MILP (Approximated) formulation

- Replace (1) with

$$\sum_{t \in \mathcal{T}} \sum_{i \in P} v_t^i$$

- Add k cuts (34) for some $\bar{p}_t^{i,h} \in [\bar{p}_{min}^i, \bar{p}_{max}^i]$, $h = 1, \ldots, k$

- $|P||\mathcal{T}|$ more variables and $k|P||\mathcal{T}|$ more constraints, vs $k|P||\mathcal{T}|$ more variables and $(2k+1)|P||\mathcal{T}|$ more constraints

- The objective function underestimates $f(p, u) \Rightarrow$ the LB is valid

Alternative MILP (Approximated) formulation

- Replace (1) with

$$\sum_{t \in \mathcal{T}} \sum_{i \in P} v_t^i$$

- Add k cuts (34) for some $\bar{p}_t^{i,h} \in [\bar{p}_{min}^i, \bar{p}_{max}^i]$, $h = 1, \ldots, k$
- $|P||\mathcal{T}|$ more variables and $k|P||\mathcal{T}|$ more constraints, vs $k|P||\mathcal{T}|$ more variables and $(2k+1)|P||\mathcal{T}|$ more constraints
- The objective function underestimates $f(p, u) \Rightarrow$ the LB is valid
- Cuts can be easily be dynamically added, much less true for variables

Alternative MILP (Approximated) formulation

- Replace (1) with

$$\sum_{t \in \mathcal{T}} \sum_{i \in P} v_t^i$$

- Add k cuts (34) for some $\bar{p}_t^{i,h} \in [\bar{p}_{min}^i, \bar{p}_{max}^i]$, $h = 1, \ldots, k$

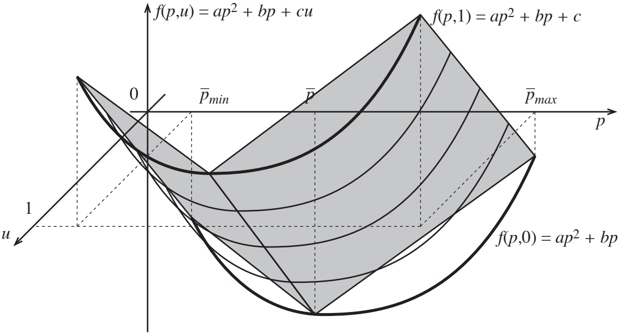
- $|P||\mathcal{T}|$ more variables and $k|P||\mathcal{T}|$ more constraints, vs $k|P||\mathcal{T}|$ more variables and $(2k+1)|P||\mathcal{T}|$ more constraints

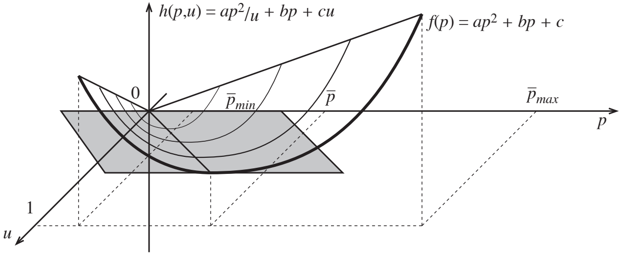
- The objective function underestimates $f(p, u) \Rightarrow$ the LB is valid

- Cuts can be easily be dynamically added, much less true for variables

- All the rest equal (static version very easy to implement)

Graphical comparison





Test setup

- SPWF: MILP formulation with (29), $k = 4$ equidistant points

Test setup

- SPWF: MILP formulation with (29), $k = 4$ equidistant points
- PCF: MILP formulation with (34), $k = 4$ equidistant points

Test setup

- SPWF: MILP formulation with (29), $k = 4$ equidistant points
- PCF: MILP formulation with (34), $k = 4$ equidistant points
- PCFD$_k$: initially, only two cuts (34) ($\bar{p} = \bar{p}_{min}$, and $\bar{p} = \bar{p}_{max}$); then, dynamic generation up to a maximum of $k|P||\mathcal{T}|$

Test setup

- SPWF: MILP formulation with (29), $k = 4$ equidistant points
- PCF: MILP formulation with (34), $k = 4$ equidistant points
- PCFD$_k$: initially, only two cuts (34) ($\bar{p} = \bar{p}_{min}$, and $\bar{p} = \bar{p}_{max}$); then, dynamic generation up to a maximum of $k|P||\mathcal{T}|$
- `Cplex 9.1`, Opteron 246 (2 GHz), 2 Gb RAM

Test setup

- SPWF: MILP formulation with (29), $k = 4$ equidistant points
- PCF: MILP formulation with (34), $k = 4$ equidistant points
- PCFD$_k$: initially, only two cuts (34) ($\bar{p} = \bar{p}_{min}$, and $\bar{p} = \bar{p}_{max}$); then, dynamic generation up to a maximum of $k|P||\mathcal{T}|$
- `Cplex 9.1`, Opteron 246 (2 GHz), 2 Gb RAM
- Two stopping tolerances: low (0.5%) and high (0.01%)

Test setup

- SPWF: MILP formulation with (29), $k = 4$ equidistant points
- PCF: MILP formulation with (34), $k = 4$ equidistant points
- PCFD$_k$: initially, only two cuts (34) ($\bar{p} = \bar{p}_{min}$, and $\bar{p} = \bar{p}_{max}$); then, dynamic generation up to a maximum of $k|P||\mathcal{T}|$
- `Cplex 9.1`, Opteron 246 (2 GHz), 2 Gb RAM
- Two stopping tolerances: low (0.5%) and high (0.01%)
- Gap w.r.t. tight Lagrangian LB, inherent one visibly worse

Test setup

- SPWF: MILP formulation with (29), $k = 4$ equidistant points
- PCF: MILP formulation with (34), $k = 4$ equidistant points
- PCFD$_k$: initially, only two cuts (34) ($\bar{p} = \bar{p}_{min}$, and $\bar{p} = \bar{p}_{max}$); then, dynamic generation up to a maximum of $k|P||\mathcal{T}|$
- `Cplex 9.1`, Opteron 246 (2 GHz), 2 Gb RAM
- Two stopping tolerances: low (0.5%) and high (0.01%)
- Gap w.r.t. tight Lagrangian LB, inherent one visibly worse
- Randomly-generated, realistic, hydro-thermal instances

  `http://www.di.unipi.it/optimize/Data`

Comparing static formulations at lower accuracy

		SPWF				PCF			
p	h	gap	nd	time	rgap	gap	nd	time	rgap
10	0	0.31	0	0.95	1.61	0.28	0	0.76	1.50
20	0	0.34	0	3.72	1.34	0.36	8	3.56	1.25
50	0	0.21	0	21.93	1.38	0.21	0	12.09	1.26
75	0	0.20	10	56.31	1.43	0.18	14	45.88	1.30
100	0	0.17	16	94.09	1.39	0.15	0	43.55	1.27
150	0	0.12	16	218.69	1.32	0.11	2	146.80	1.20
200	0	0.09	6	267.78	1.37	0.08	0	234.97	1.25
20	10	0.21	140	93.53	0.82	0.20	0	3.71	0.69
50	20	0.06	0	17.98	0.70	0.10	0	18.93	0.63
75	35	0.11	170	96.86	0.57	0.07	70	64.52	0.52
100	50	0.06	180	130.86	0.58	0.07	35	81.41	0.53
150	75	0.06	300	467.62	0.58	0.05	90	293.50	0.52
200	100	0.05	205	427.71	0.56	0.03	35	314.00	0.51

Static vs. dynamic formulations at lower accuracy

		PCF			$PCFD_4$			$PCFD_\infty$		
p	h	gap	nd	time	gap	nd	time	gap	nd	time
10	0	0.28	0	0.76	0.30	0	0.86	0.28	0	0.80
20	0	0.36	8	3.56	0.36	0	2.51	0.33	0	3.00
50	0	0.21	0	12.09	0.19	0	14.17	0.18	0	13.08
75	0	0.18	14	45.88	0.19	2	36.62	0.22	0	22.58
100	0	0.15	0	43.55	0.17	0	34.31	0.20	0	36.51
150	0	0.11	2	146.80	0.11	4	104.68	0.12	10	169.68
200	0	0.08	0	234.97	0.10	0	183.01	0.14	12	235.60
20	10	0.20	0	3.71	0.30	5	4.18	0.15	0	2.51
50	20	0.10	0	18.93	0.10	10	19.06	0.13	0	10.93
75	35	0.07	70	64.52	0.05	115	70.55	0.03	95	64.80
100	50	0.07	35	81.41	0.05	15	47.62	0.04	40	60.78
150	75	0.05	90	293.50	0.05	115	194.10	0.05	115	216.33
200	100	0.03	35	314.00	0.02	0	155.36	0.03	135	342.69

Results with higher accuracy (0.01%)

		SPWF		PCF		$PCFD_4$		$PCFD_\infty$	
p	h	gap	time	gap	time	gap	time	gap	time
10	0	0.01	22	0.01	15	0.01	12	0.01	16
20	0	0.01	3480	0.02	2969	0.02	3614	0.01	3481
50	0	0.09	10000	0.09	10000	0.08	10000	0.09	10000
75	0	0.09	10000	0.09	10000	0.08	10000	0.08	10000
100	0	0.07	10000	0.06	10000	0.06	10000	0.06	10000
150	0	0.07	10000	0.05	10000	0.05	10000	0.05	10000
200	0	0.07	10000	0.06	10000	0.05	10000	0.05	10000
20	10	0.01	288	0.01	383	0.01	238	0.01	317
50	20	0.01	9613	0.00	6855	0.00	7772	0.01	8326
75	35	0.01	10000	0.01	10000	0.01	10000	0.01	8326
100	50	0.01	10000	0.01	10000	0.01	10000	0.01	10000
150	75	0.01	10000	0.01	10000	0.01	10000	0.01	10000
200	100	0.01	10000	0.01	10000	0.01	10000	0.01	10000

Nonseparable problems

- What happens if the problem if nonseparable?

$$\begin{array}{ll} \min & x^T Q x + qx + cy \\ & Ax + By \geq b \\ & l_i y_i \leq x_i \leq u_i y_i \quad , \quad y_i \in \{0,1\} \quad i = 1, \ldots, n \end{array} \tag{35}$$

Nonseparable problems

- What happens if the problem if nonseparable?

$$\begin{array}{ll} \min & x^T Qx + qx + cy \\ & Ax + By \geq b \\ & l_i y_i \leq x_i \leq u_i y_i \ \ , \ \ y_i \in \{0,1\} \quad i = 1, \ldots, n \end{array} \tag{35}$$

- Even assuming $Q \succeq 0 \Rightarrow$ o.f. convex, (33) cannot be used

Nonseparable problems

- What happens if the problem if nonseparable?

$$
\begin{array}{ll}
\min & x^T Qx + qx + cy \\
 & Ax + By \geq b \\
 & l_i y_i \leq x_i \leq u_i y_i \ \ , \ \ y_i \in \{0,1\} \quad i = 1, \ldots, n
\end{array}
\tag{35}
$$

- Even assuming $Q \succeq 0 \Rightarrow$ o.f. convex, (33) cannot be used
- However, a dirty trick was proposed in our [13]

$$
\begin{array}{ll}
\min & x^T Dx + z^T(Q - D)z + qx + cy \\
 & Ax + By \geq b \ \ , \ \ z = x \\
 & l_i y_i \leq x_i \leq u_i y_i \ \ , \ \ y_i \in \{0,1\} \qquad i = 1, \ldots, n
\end{array}
\tag{36}
$$

for non-negative diagonal $D \in \mathbb{R}^{n \times n}$ such that $Q - D \succeq 0$

Nonseparable problems

- What happens if the problem if nonseparable?

$$\begin{array}{ll} \min & x^T Q x + qx + cy \\ & Ax + By \geq b \\ & l_i y_i \leq x_i \leq u_i y_i \;\;,\;\; y_i \in \{0,1\} \quad i = 1, \dots, n \end{array} \tag{35}$$

- Even assuming $Q \succeq 0 \Rightarrow$ o.f. convex, (33) cannot be used
- However, a dirty trick was proposed in our [13]

$$\begin{array}{ll} \min & x^T D x + z^T (Q - D) z + qx + cy \\ & Ax + By \geq b \;\;,\;\; z = x \\ & l_i y_i \leq x_i \leq u_i y_i \;\;,\;\; y_i \in \{0,1\} \qquad i = 1, \dots, n \end{array} \tag{36}$$

for non-negative diagonal $D \in \mathbb{R}^{n \times n}$ such that $Q - D \succeq 0$

- Move nonseparability to new variables z, let D "as large as possible"

Nonseparable problems

- What happens if the problem if nonseparable?

$$\begin{array}{ll} \min & x^T Qx + qx + cy \\ & Ax + By \geq b \\ & l_i y_i \leq x_i \leq u_i y_i \;\;,\;\; y_i \in \{0,1\} \quad i = 1,\ldots,n \end{array} \tag{35}$$

- Even assuming $Q \succeq 0 \Rightarrow$ o.f. convex, (33) cannot be used
- However, a dirty trick was proposed in our [13]

$$\begin{array}{ll} \min & x^T Dx + z^T (Q - D) z + qx + cy \\ & Ax + By \geq b \;\;,\;\; z = x \\ & l_i y_i \leq x_i \leq u_i y_i \;\;,\;\; y_i \in \{0,1\} \qquad i = 1,\ldots,n \end{array} \tag{36}$$

  for non-negative diagonal $D \in \mathbb{R}^{n \times n}$ such that $Q - D \succeq 0$
- Move nonseparability to new variables z, let D "as large as possible"
- D can be chosen e.g. as $\lambda_{min}(Q)I$

Nonseparable applications

- Mean-Variance problem with min and max buy-in thresholds

$$\min \left\{ x^T Q x \;\middle|\; \begin{array}{l} ex = 1 \;\;,\;\; \mu x \geq \rho \;, \\ l_i y_i \leq x_i \leq u_i y_i \;,\; y_i \in \{0,1\} \;\; i = 1, \ldots, n \end{array} \right\}$$

μ = expected return, Q = covariance matrix, ρ = desired return
l, u = min, max buy-in thresholds

- Real-world requirement, almost no structure = good for testing

Nonseparable applications

- Mean-Variance problem with min and max buy-in thresholds

$$\min \left\{ x^T Q x \;\middle|\; \begin{array}{l} ex = 1 \;\;,\;\; \mu x \geq \rho \;, \\ l_i y_i \leq x_i \leq u_i y_i \;,\;\; y_i \in \{0,1\} \;\; i = 1, \ldots, n \end{array} \right\}$$

  $\mu =$ expected return, $Q =$ covariance matrix, $\rho =$ desired return
  $l, u =$ min, max buy-in thresholds

- Real-world requirement, almost no structure = good for testing

- Dirty trick + (33) improve lower bounds a lot

  ...but not enough for routinely solving large instances

  ...although a lot better than `Cplex`

Nonseparable applications

- Mean-Variance problem with min and max buy-in thresholds

$$\min \left\{ x^T Q x \;\middle|\; \begin{array}{l} ex = 1 \;\;,\;\; \mu x \geq \rho \;, \\ l_i y_i \leq x_i \leq u_i y_i \;,\; y_i \in \{0,1\} \;\; i = 1, \ldots, n \end{array} \right\}$$

  μ = expected return, Q = covariance matrix, ρ = desired return
  l, u = min, max buy-in thresholds

- Real-world requirement, almost no structure = good for testing

- Dirty trick + (33) improve lower bounds a lot

  ...but not enough for routinely solving large instances

  ...although a lot better than `Cplex`

- Maybe due to a wrong choice of D?

Choosing D via SDP

- Assuming $tr(D)$ the relevant metric, the "largest" D solves[14]

$$\begin{array}{c} \max \left\{ \sum_{i=1}^{n} d_i \ : \ Q - \sum_{i=1}^{n} d_i(e_i e_i^T) \succeq 0 \ , \ d \geq 0 \right\} \\ \min \left\{ tr(QX) \ : \ diag(X) \geq e \ , \ X \succeq 0 \right\} \end{array} \tag{37}$$

dual pair of SemiDefinite (= convex = easy) Problems

[14] F., C. Gentile "SDP Diagonalizations and Perspective Cuts for a Class of Nonseparable MIQP", *Operations Research Letters*, 2007

Choosing D via SDP

- Assuming $tr(D)$ the relevant metric, the "largest" D solves[14]

$$\begin{array}{c} \max \left\{ \sum_{i=1}^{n} d_i \ : \ Q - \sum_{i=1}^{n} d_i(e_i e_i^T) \succeq 0 \ , \ d \geq 0 \right\} \\ \min \left\{ tr(QX) \ : \ diag(X) \geq e \ , \ X \succeq 0 \right\} \end{array} \tag{37}$$

  dual pair of SemiDefinite (= convex = easy) Problems

- Several, efficient, open-source SDP codes

[14] F., C. Gentile "SDP Diagonalizations and Perspective Cuts for a Class of Nonseparable MIQP", *Operations Research Letters*, 2007

Choosing D via SDP

- Assuming $tr(D)$ the relevant metric, the "largest" D solves[14]

$$\max \left\{ \sum_{i=1}^n d_i \ :\ Q - \sum_{i=1}^n d_i(e_i e_i^T) \succeq 0 \ ,\ d \geq 0 \right\}$$
$$\min \left\{ tr(QX) \ :\ diag(X) \geq e \ ,\ X \succeq 0 \right\} \tag{37}$$

  dual pair of SemiDefinite (= convex = easy) Problems

- Several, efficient, open-source SDP codes

- Interesting relaxation: removing $d \geq 0$ in the primal gives

$$\min \left\{ tr(QX) \ :\ diag(X) = e \ ,\ X \succeq 0 \right\} \tag{38}$$

[14] F., C. Gentile "SDP Diagonalizations and Perspective Cuts for a Class of Nonseparable MIQP", *Operations Research Letters*, 2007

Choosing D via SDP

- Assuming $tr(D)$ the relevant metric, the "largest" D solves[14]

$$\begin{array}{c} \max \left\{ \sum_{i=1}^{n} d_i \;:\; Q - \sum_{i=1}^{n} d_i(e_i e_i^T) \succeq 0 \,,\; d \geq 0 \right\} \\ \min \left\{ tr(QX) \;:\; diag(X) \geq e \,,\; X \succeq 0 \right\} \end{array} \tag{37}$$

  dual pair of SemiDefinite (= convex = easy) Problems

- Several, efficient, open-source SDP codes

- Interesting relaxation: removing $d \geq 0$ in the primal gives

$$\min \left\{ tr(QX) \;:\; diag(X) = e \,,\; X \succeq 0 \right\} \tag{38}$$

  - $d^* > 0$ anyway in all our tests

[14] F., C. Gentile "SDP Diagonalizations and Perspective Cuts for a Class of Nonseparable MIQP", *Operations Research Letters*, 2007

Choosing D via SDP

- Assuming $tr(D)$ the relevant metric, the "largest" D solves[14]

$$\begin{array}{c} \max \left\{ \sum_{i=1}^{n} d_i \;:\; Q - \sum_{i=1}^{n} d_i(e_i e_i^T) \succeq 0 \;,\; d \geq 0 \right\} \\ \min \left\{ tr(QX) \;:\; diag(X) \geq e \;,\; X \succeq 0 \right\} \end{array} \tag{37}$$

  dual pair of SemiDefinite (= convex = easy) Problems

- Several, efficient, open-source SDP codes

- Interesting relaxation: removing $d \geq 0$ in the primal gives

$$\min \left\{ tr(QX) \;:\; diag(X) = e \;,\; X \succeq 0 \right\} \tag{38}$$

  - $d^* > 0$ anyway in all our tests
  - most often faster to solve in practice by all codes

[14] F., C. Gentile "SDP Diagonalizations and Perspective Cuts for a Class of Nonseparable MIQP", *Operations Research Letters*, 2007

Choosing D via SDP

- Assuming $tr(D)$ the relevant metric, the "largest" D solves[14]

$$\max \left\{ \textstyle\sum_{i=1}^{n} d_i \;:\; Q - \sum_{i=1}^{n} d_i(e_i e_i^T) \succeq 0 \;,\; d \geq 0 \right\}$$
$$\min \left\{ tr(QX) \;:\; diag(X) \geq e \;,\; X \succeq 0 \right\} \tag{37}$$

  dual pair of SemiDefinite ($=$ convex $=$ easy) Problems

- Several, efficient, open-source SDP codes

- Interesting relaxation: removing $d \geq 0$ in the primal gives

$$\min \left\{ tr(QX) \;:\; diag(X) = e \;,\; X \succeq 0 \right\} \tag{38}$$

  - $d^* > 0$ anyway in all our tests
  - most often faster to solve in practice by all codes
  - *constant trace* $=$ max eigenvalue problem, specialized approaches (`SBundle`)

[14] F., C. Gentile "SDP Diagonalizations and Perspective Cuts for a Class of Nonseparable MIQP", *Operations Research Letters*, 2007

Choosing D via SDP (cont.d)

- Simple idea: compare min-eigenvalue with (37)/(38)
- Trade-off: improvement in “size” of D versus running time
- Different SDP solvers, different instances
- Improving $tr(D) \Rightarrow$ better cuts $\Rightarrow$ better bounds?
- Notes:
  - new approach works even if $\lambda_{min}(Q) = 0$
  - could use weighted objective function wd
    (but how to choose weights w?)
  - funny coincidence: (38) is the SDP relaxation of Max-Cut
    (maximizing, i.e., with “$-Q$”)

The instances

- 30 randomly-generated instances for each $n \in \{200, 300, 400\}$
- $\mu_i \in [0.002, 0.01]$, $l_i \in [0.075, 0.125]$, $u_i \in [0.375, 0.425]$ (uniformly)

[15] P.M. Pardalos, G.P. Rodgers "Computing Aspects of a Branch and Bound Algorithm for Quadratic Zero-One Programming" *Computing*, 1990

The instances

- 30 randomly-generated instances for each $n \in \{200, 300, 400\}$
- $\mu_i \in [0.002, 0.01]$, $l_i \in [0.075, 0.125]$, $u_i \in [0.375, 0.425]$ (uniformly)
- Q = well-known random generator[15]

[15] P.M. Pardalos, G.P. Rodgers "Computing Aspects of a Branch and Bound Algorithm for Quadratic Zero-One Programming" *Computing*, 1990

The instances

- 30 randomly-generated instances for each $n \in \{200, 300, 400\}$
- $\mu_i \in [0.002, 0.01]$, $l_i \in [0.075, 0.125]$, $u_i \in [0.375, 0.425]$ (uniformly)
- $Q =$ well-known random generator[15]
- Parameters of the generator heavily impact dominance index

$$S = \text{average} \left\{ \frac{Q_{ii} - \sum_{j \neq i} |Q_{ij}|}{Q_{ii}} \ : \ i = 1, \ldots, n \right\}$$

which in turn heavily impacts effectiveness of perspective cuts

[15] P.M. Pardalos, G.P. Rodgers "Computing Aspects of a Branch and Bound Algorithm for Quadratic Zero-One Programming" *Computing*, 1990

The instances

- 30 randomly-generated instances for each $n \in \{200, 300, 400\}$
- $\mu_i \in [0.002, 0.01]$, $l_i \in [0.075, 0.125]$, $u_i \in [0.375, 0.425]$ (uniformly)
- Q = well-known random generator[15]
- Parameters of the generator heavily impact dominance index

$$S = \text{average} \left\{ \frac{Q_{ii} - \sum_{j \neq i} |Q_{ij}|}{Q_{ii}} \;:\; i = 1, \ldots, n \right\}$$

  which in turn heavily impacts effectiveness of perspective cuts
- For each n, three classes of instances (10 each):
  - "+" instances, $S \approx 0.6$ (diagonally dominant)
  - "0" instances, $S \approx 0$ (diagonally quasi-dominant)
  - "−" instances, $S \approx -0.5$ (not diagonally dominant)

[15] P.M. Pardalos, G.P. Rodgers "Computing Aspects of a Branch and Bound Algorithm for Quadratic Zero-One Programming" *Computing*, 1990

SDP codes

- Five open-source standalone (no Matlab) SDP codes:
  `CSDP 4.9`, `DSDP 5.6`, `SBmethod 1.1.3`, `SDPA 6.0`, `SDPLR 1.02`
  `http://www-user.tu-chemnitz.de/ helmberg/semidef.html`
  (C. Helmberg's SDP page)

SDP codes

- Five open-source standalone (no Matlab) SDP codes:
  `CSDP 4.9`, `DSDP 5.6`, `SBmethod 1.1.3`, `SDPA 6.0`, `SDPLR 1.02`
  `http://www-user.tu-chemnitz.de/ helmberg/semidef.html`
  (C. Helmberg's SDP page)
- Min-Eigenvalue solved by `eig()` of `octave 2.1` (+ sorting)

SDP codes

- Five open-source standalone (no Matlab) SDP codes:
  `CSDP 4.9`, `DSDP 5.6`, `SBmethod 1.1.3`, `SDPA 6.0`, `SDPLR 1.02`
  `http://www-user.tu-chemnitz.de/ helmberg/semidef.html`
  (C. Helmberg's SDP page)

- Min-Eigenvalue solved by `eig()` of `octave 2.1` (+ sorting)

- d_{max}, d_{min}, d_{avg} = ratio w.r.t. $\lambda_{min}(Q)$

SDP codes

- Five open-source standalone (no Matlab) SDP codes:
  `CSDP 4.9`, `DSDP 5.6`, `SBmethod 1.1.3`, `SDPA 6.0`, `SDPLR 1.02`
  `http://www-user.tu-chemnitz.de/ helmberg/semidef.html`
  (C. Helmberg's SDP page)

- Min-Eigenvalue solved by `eig()` of `octave 2.1` (+ sorting)

- d_{max}, d_{min}, d_{avg} = ratio w.r.t. $\lambda_{min}(Q)$

- "$\geq$" full version (37), "$=$" relaxation (38)
  (except `SBundle` which can only solve "$=$")

SDP codes

- Five open-source standalone (no Matlab) SDP codes:
  `CSDP 4.9`, `DSDP 5.6`, `SBmethod 1.1.3`, `SDPA 6.0`, `SDPLR 1.02`
  `http://www-user.tu-chemnitz.de/ helmberg/semidef.html`
  (C. Helmberg's SDP page)

- Min-Eigenvalue solved by `eig()` of `octave 2.1` (+ sorting)

- d_{max}, d_{min}, d_{avg} = ratio w.r.t. $\lambda_{min}(Q)$

- "$\geq$" full version (37), "$=$" relaxation (38)
  (except `SBundle` which can only solve "$=$")

- Running times on a bi-Opteron 246 processor, 2Gb RAM, Linux, gcc.

Comparison of SDP codes (only three)

				ME	CSDP		DSDP		SB
	d_{max}	d_{min}	d_{avg}		$\geq$	$=$	$\geq$	$=$	$=$
200^+	1.96	0.97	1.47	0.13	3.12	2.98	1.86	0.10	23.77
200^0	1.93	0.90	1.41	0.13	3.03	2.99	1.87	0.10	16.39
200^-	1.86	0.87	1.37	0.13	3.00	2.95	1.86	0.10	16.58
300^+	1.97	0.97	1.47	0.23	10.54	9.84	4.92	0.26	69.13
300^0	1.93	0.91	1.42	0.23	10.91	9.55	4.99	0.26	46.01
300^-	1.69	0.89	1.29	0.23	10.91	9.62	5.10	0.26	41.82
400^+	1.98	0.97	1.47	0.39	31.03	29.28	10.56	0.52	146.07
400^0	1.93	0.93	1.43	0.39	37.24	31.27	10.86	0.52	94.62
400^-	1.87	0.89	1.38	0.39	36.77	31.61	10.75	0.52	90.07

Comparison of SDP codes (only three)

	d_{max}	d_{min}	d_{avg}	ME	CSDP $\geq$	CSDP $=$	DSDP $\geq$	DSDP $=$	SB $=$
200^+	1.96	0.97	1.47	0.13	3.12	2.98	1.86	0.10	23.77
200^0	1.93	0.90	1.41	0.13	3.03	2.99	1.87	0.10	16.39
200^-	1.86	0.87	1.37	0.13	3.00	2.95	1.86	0.10	16.58
300^+	1.97	0.97	1.47	0.23	10.54	9.84	4.92	0.26	69.13
300^0	1.93	0.91	1.42	0.23	10.91	9.55	4.99	0.26	46.01
300^-	1.69	0.89	1.29	0.23	10.91	9.62	5.10	0.26	41.82
400^+	1.98	0.97	1.47	0.39	31.03	29.28	10.56	0.52	146.07
400^0	1.93	0.93	1.43	0.39	37.24	31.27	10.86	0.52	94.62
400^-	1.87	0.89	1.38	0.39	36.77	31.61	10.75	0.52	90.07

- On average 50% better than λ_{min}, worst case $\approx$ few % worse
- Results getting worse as Q less diagonally dominant
- Times not much worse using right code and model
- Is it worth?

Impact on the B&C approach

	SDP			ME			Cplex		
	time	d.gap	r.gap	time	d.gap	r.gap	p.gap	d.gap	r.gap
200^+	164		1.14	904		6.48	0.14	45.33	85.63
200^0	161		2.14	320		6.10	0.38	51.27	84.47
200^-	1902		3.65	3306	0.02	6.69	0.24	42.09	78.88
300^+	818		4.54	2061		5.62	0.41	64.68	92.01
300^0	856		1.97	1715		6.28	0.43	59.91	87.87
300^-	1709		2.68	2797	0.05	7.04	0.53	45.11	78.77
400^+	2264		4.79	4756	0.10	6.15	1.03	61.47	89.06
400^0	4378	0.10	2.29	7421	0.16	6.53	1.18	68.68	90.03
400^-	6311	0.23	3.06	6901	0.36	6.49	1.60	65.88	88.47

Impact on the B&C approach

	SDP			ME			Cplex		
	time	d.gap	r.gap	time	d.gap	r.gap	p.gap	d.gap	r.gap
200^+	164		1.14	904		6.48	0.14	45.33	85.63
200^0	161		2.14	320		6.10	0.38	51.27	84.47
200^-	1902		3.65	3306	0.02	6.69	0.24	42.09	78.88
300^+	818		4.54	2061		5.62	0.41	64.68	92.01
300^0	856		1.97	1715		6.28	0.43	59.91	87.87
300^-	1709		2.68	2797	0.05	7.04	0.53	45.11	78.77
400^+	2264		4.79	4756	0.10	6.15	1.03	61.47	89.06
400^0	4378	0.10	2.29	7421	0.16	6.53	1.18	68.68	90.03
400^-	6311	0.23	3.06	6901	0.36	6.49	1.60	65.88	88.47

- root node gap halved+ w.r.t. ME, $\approx$ 1% w.r.t. $\approx$ 80% for `Cplex`
- All instances up to $n = 300$ solved to optimality within 10000s, `Cplex` solves none, ME does not solve some
- Effectiveness worsens as Q less dominant, could not solve a few 400^- instances

A Hybrid Approach

- Lagrangian approach provides very good lower bounds, quickly

A Hybrid Approach

- Lagrangian approach provides very good lower bounds, quickly

  but heuristic not so effective and efficient (many (ED), costly)

A Hybrid Approach

- Lagrangian approach provides very good lower bounds, quickly

  but heuristic not so effective and efficient (many (ED), costly)

- MILP approach provides very good feasible solutions, quickly

A Hybrid Approach

- Lagrangian approach provides very good lower bounds, quickly

  but heuristic not so effective and efficient (many (ED), costly)

- MILP approach provides very good feasible solutions, quickly

  but lower bounds loose, no termination for tight tolerance

A Hybrid Approach

- Lagrangian approach provides very good lower bounds, quickly

  but heuristic not so effective and efficient (many (ED), costly)

- MILP approach provides very good feasible solutions, quickly

  but lower bounds loose, no termination for tight tolerance

$$\Downarrow$$

Why not using both?

A Hybrid Approach

- Lagrangian approach provides very good lower bounds, quickly

  but heuristic not so effective and efficient (many (ED), costly)

- MILP approach provides very good feasible solutions, quickly

  but lower bounds loose, no termination for tight tolerance

$$\Downarrow$$

Why not using both?

- Lagrangian bound computed at root node (no heuristic = quick)
- Used to stop search as soon as good enough feasible solution found

  (admittedly very coarse)

Results – 0.5%

		$PCFD_4$				$PCFD_\infty$			
		NoLB		LB		NoLB		LB	
p	h	gap	time	gap	time	gap	time	gap	time
10	0	0.28	0.80	0.34	0.97	0.30	0.86	0.37	1.06
20	0	0.33	3.00	0.32	3.60	0.36	2.51	0.36	3.16
50	0	0.18	13.08	0.19	27.46	0.19	14.17	0.20	16.39
75	0	0.22	22.58	0.25	28.82	0.19	36.62	0.22	28.05
100	0	0.20	36.51	0.15	41.44	0.17	34.31	0.16	60.16
150	0	0.12	169.68	0.10	148.88	0.11	104.68	0.11	136.18
200	0	0.14	235.60	0.08	323.36	0.10	183.01	0.08	258.57
20	10	0.15	2.51	0.17	4.21	0.30	4.18	0.24	6.34
50	20	0.13	10.93	0.10	26.96	0.10	19.06	0.10	12.51
75	35	0.03	64.80	0.06	59.47	0.05	70.55	0.10	75.23
100	50	0.04	60.78	0.04	44.95	0.05	47.62	0.05	66.61
150	75	0.05	216.33	0.02	244.05	0.05	194.10	0.04	228.32
200	100	0.03	342.69	0.03	253.59	0.02	155.36	0.02	217.56

- Sizable relative (although small absolute) increase for small instances
- No clear positive effect

Results – 0.1%

		$PCFD_4$				$PCFD_\infty$			
		NoLB		LB		NoLB		LB	
p	h	gap	time	gap	time	gap	time	gap	time
10	0	0.10	12.45	0.10	12.70	0.10	9.77	0.10	9.97
20	0	0.10	1295.28	0.10	2201.87	0.10	1169.94	0.10	1157.22
50	0	0.09	8279.78	0.11	4084.79	0.10	10000.00	0.11	4014.01
75	0	0.07	10000.00	0.09	3974.94	0.07	10000.00	0.09	2286.03
100	0	0.07	10000.00	0.09	289.01	0.06	10000.00	0.09	94.56
150	0	0.05	10000.00	0.06	193.38	0.05	10000.00	0.08	207.86
200	0	0.05	10000.00	0.07	337.33	0.06	10000.00	0.07	315.88
20	10	0.07	31.38	0.09	14.31	0.07	41.08	0.08	30.01
50	20	0.02	41.86	0.05	27.22	0.02	47.62	0.04	12.92
75	35	0.03	64.45	0.06	57.95	0.04	81.77	0.06	71.03
100	50	0.03	40.61	0.04	41.42	0.04	60.20	0.05	62.85
150	75	0.02	232.99	0.02	235.04	0.04	191.52	0.04	203.18
200	100	0.03	240.38	0.03	231.35	0.02	198.25	0.02	206.66

- Huge positive impact on large thermals, some effect on small hydro
- Gap worsens somewhat (which is expected)

Results – 0.05%

		PCFD$_4$				PCFD$_\infty$			
		NoLB		LB		NoLB		LB	
p	h	gap	time	gap	time	gap	time	gap	time
10	0	0.06	15.42	0.06	15.72	0.06	11.63	0.06	11.85
20	0	0.06	2473.11	0.06	2440.86	0.06	2470.49	0.06	2499.97
50	0	0.09	10000.00	0.09	8113.35	0.09	10000.00	0.10	8489.08
75	0	0.09	10000.00	0.09	10002.22	0.08	8256.79	0.08	8259.00
100	0	0.07	10000.00	0.07	8018.89	0.06	10000.00	0.06	6538.84
150	0	0.05	10000.00	0.06	5151.71	0.05	10000.00	0.06	6151.20
200	0	0.05	10000.00	0.05	6255.99	0.06	10000.00	0.06	6271.77
20	10	0.06	73.26	0.06	73.00	0.06	71.19	0.06	68.40
50	20	0.01	623.95	0.02	34.44	0.01	269.34	0.03	44.53
75	35	0.02	177.50	0.03	59.37	0.02	124.85	0.03	100.47
100	50	0.02	438.39	0.04	39.45	0.02	665.37	0.05	60.00
150	75	0.02	1669.30	0.02	224.67	0.01	1144.10	0.04	201.31
200	100	0.02	1082.41	0.03	238.81	0.01	451.98	0.02	202.94

- Diminishing but still positive on large thermals, especially PCFD$_\infty$
- Huge positive impact on all but the smallest hydro

Results – 0.01%

		PCFD$_4$				PCFD$_\infty$			
		NoLB		LB		NoLB		LB	
p	h	gap	time	gap	time	gap	time	gap	time
10	0	0.02	16.66	0.02	16.84	0.02	12.49	0.02	12.80
20	0	0.02	3547.24	0.02	3699.26	0.02	3914.50	0.02	3946.51
50	0	0.09	10000.00	0.09	10001.25	0.09	10000.00	0.09	10001.25
75	0	0.09	10000.00	0.09	10002.22	0.08	10000.00	0.08	10002.22
100	0	0.07	10000.00	0.07	10003.68	0.06	10000.00	0.06	10003.68
150	0	0.05	10000.00	0.05	10006.14	0.05	10000.00	0.05	10006.14
200	0	0.05	10000.00	0.05	8248.37	0.06	10000.00	0.06	10008.52
20	10	0.02	268.49	0.02	263.40	0.02	248.75	0.02	255.95
50	20	0.00	7285.00	0.01	841.26	0.01	6495.96	0.01	121.86
75	35	0.01	10000.00	0.01	5033.34	0.01	10000.00	0.01	5045.42
100	50	0.01	10000.00	0.01	1198.73	0.01	10000.00	0.01	5789.69
150	75	0.01	10000.00	0.01	3376.87	0.01	10000.00	0.01	1145.61
200	100	0.01	10000.00	0.01	1182.27	0.01	10000.00	0.01	463.46

- No longer any impact (thus, slightly negative) on thermals
- Still huge positive impact on all but the smallest hydro

Conclusions (general)

Conclusions (general)

You get to meet all sorts, in this line of work

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network
  - shortest part (dynamic programming)

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network
  - shortest part (dynamic programming)
  - nonlinear, convex, (two different) special structure(s)

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network
  - shortest part (dynamic programming)
  - nonlinear, convex, (two different) special structure(s)
  - combinatorial, heuristic

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network
  - shortest part (dynamic programming)
  - nonlinear, convex, (two different) special structure(s)
  - combinatorial, heuristic
  - linear, mixed-integer

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network
  - shortest part (dynamic programming)
  - nonlinear, convex, (two different) special structure(s)
  - combinatorial, heuristic
  - linear, mixed-integer
  - semidefinite, convex

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network
  - shortest part (dynamic programming)
  - nonlinear, convex, (two different) special structure(s)
  - combinatorial, heuristic
  - linear, mixed-integer
  - semidefinite, convex

  each one with a definite and useful role

Conclusions (general)

You get to meet all sorts, in this line of work

- In this talk, you have seen algorithms for:
  - quadratic, convex, mixed-integer
  - convex, nondifferentiable
  - linear, network
  - shortest part (dynamic programming)
  - nonlinear, convex, (two different) special structure(s)
  - combinatorial, heuristic
  - linear, mixed-integer
  - semidefinite, convex

  each one with a definite and useful role

- Good methodologies bring good results to interesting problems

  interesting problems motivate the development of good methodologies

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

why not both?

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

  why not both?

- Some problems on the edge of being routinely solvable

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

  why not both?

- Some problems on the edge of being routinely solvable
- Plenty more still extremely difficult:
  - short-long term (weekly)

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

  why not both?

- Some problems on the edge of being routinely solvable
- Plenty more still extremely difficult:
  - short-long term (weekly)
  - market variants (zonal prices)

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

  why not both?

- Some problems on the edge of being routinely solvable
- Plenty more still extremely difficult:
  - short-long term (weekly)
  - market variants (zonal prices)
  - long term (water)

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

  why not both?

- Some problems on the edge of being routinely solvable
- Plenty more still extremely difficult:
  - short-long term (weekly)
  - market variants (zonal prices)
  - long term (water)
  - uncertainty (prices, weather, accidents, competitors, ...)

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

  why not both?

- Some problems on the edge of being routinely solvable
- Plenty more still extremely difficult:
  - short-long term (weekly)
  - market variants (zonal prices)
  - long term (water)
  - uncertainty (prices, weather, accidents, competitors, . . .)
  - . . .

Conclusions (energy problems)

- Lots of challenging problems in (electrical) energy production
- Approximated MILP formulations or Lagrangian techniques?

  why not both?

- Some problems on the edge of being routinely solvable
- Plenty more still extremely difficult:
  - short-long term (weekly)
  - market variants (zonal prices)
  - long term (water)
  - uncertainty (prices, weather, accidents, competitors, . . .)
  - . . .

Bring them on! :-)