Real-Time Embedded Optimization for the Smart Grid

Stephen Boyd
Matt Kraning, Yang Wang, Jacob Mattingley

LANL Workshop on Optimization & Control Theory for Smart Grids, 8/10/10

Smart grid

- embed intelligence in energy systems to
  - do more with less
  - reduce CO_2 emissions
  - handle uncertainties in generation (wind, solar, . . .)
  - exploit new demand response capabilities
  - handle shift towards EVs
  - extend life of current infrastructure

- cf. current system
  - load is what it is; generation scheduled to match it
  - systems built with large margins for max load

Smart grid critical technologies: The big picture

- physical layer
  - photovoltaics, switches, storage, fuel cells, . . .
- infrastructure/plumbing
  - smart enabled stuff, communication protocols, security, . . .
- **algorithms** (our focus)
  - real-time decision making
- economics layer
  - markets, investment, regulation, . . .

Optimization

- algorithm chooses optimal (or just good) values of some (decision) variables, given mathematical model, objectives, and constraints
- a.k.a. operations research, synthesis, automatic control, planning, . . .
- modern age dates to 1948; huge advances (mostly, Moore's law) since
- widely used in hundreds of disciplines and industries
  - economics, finance, supply-chain, operations, advertising
  - statistics, machine learning, signal processing
  - aerospace, engineering design
  - and yes, energy systems

Optimization

- optimization can be organized/implemented several ways
  - centralized
  - distributed (tightly or loosely coupled)
  - ad hoc, self-organized, peer-to-peer
  - market, auction

  or any combination . . .

- our ability to solve optimization problems varies widely, depending on
  - mathematical form of problem (convexity)
  - problem scale
  - required solution time, reliability

Real-time embedded optimization for the smart grid

- embed optimization technology in devices & systems for energy generation, delivery, storage, and use
- embedded optimization can be used for (real-time)
  - allocation (and re-allocation) of resources
  - routing of power, work, other commodities over a network
  - scheduling delivery, generation, usage
  - clearing markets, coordination, planning

Real-time embedded optimization for the smart grid

- embedded optimization can handle
  - dynamic (time) effects: storage, deferrable loads, dynamic constraints
  - spatial effects: networks, generator/load locations, transmission line losses/capacities
  - uncertainty in demand, generation (wind/solar), prices
  - losses, failures, gross system changes ($e.g.$, communication loss)

- **embedded optimization is what will make the smart grid 'smart'**

Real-time embedded optimization

- not a radical concept: already used for
  - generator dispatch
  - process control
  - flight management, control
  - finance
  - airline scheduling
  - supply chain optimization, revenue management
- often associated with 'big iron' systems
  - big computers
  - hours of computation time
  - staff of PhDs to babysit/oversee

What's new

- **optimization can be embedded in small systems**

- new methods allow
  - optimization in micro/milliseconds
    ($1000\times$ faster than generic fast solvers)
  - reliable code, small footprint
  - distributed architectures

- can embed in individual HVAC systems, refrigerators, PHEVs, data centers, distributed generation/storage, . . .

Dynamic optimization with recourse

- actions (choices)
  - are taken (made) repeatedly
  - affect future (expend resources, do work, . . .)
  - must be made with current information
- has many names
  - sequential decision making
  - automatic control, stochastic control
- extensive theory
  - can solve some special cases (linear dynamics, quadratic objective)
  - general case intractable
  - many suboptimal methods that work well

Receding horizon control

- a (powerful) heuristic for stochastic control
- based on solving an optimization problem in each step
- relies on model of system evolution, including effects
  - within our control (‘actions’ or ‘inputs’)
  - outside our control (‘disturbances’, ‘exogenous inputs’)
- RHC algorithm: at each time step
  - predict future disturbances using current information
  - plan (optimize) actions 30 steps into the future, assuming predictions are correct
  - execute first step in the plan

Receding horizon control

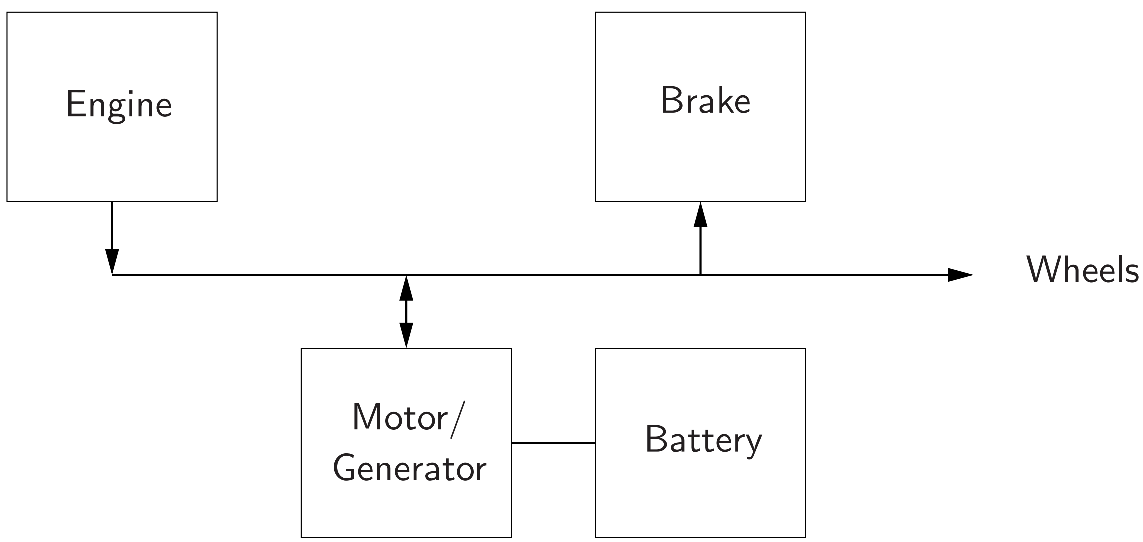
- predictions can come from
  - statistical estimates, machine learning
  - analyst forecasts, futures markets
- works extremely well, even with bad predictions
- handles constraints (transmission line capacities, generator limits)
- used in many application areas, $e.g.$, finance, aerospace, chemical process control, supply chain, revenue management, unit commitment
- known by many other names: model predictive control, dynamic linear programming, rolling horizon planning

Examples

- some very simple examples
  - hybrid vehicle energy management
  - HVAC control
  - processor speed scheduling
  - energy storage control
  - multi-carrier energy system
  - load balancing
- even for these examples, optimization beats heuristics
- optimization can just as well handle more complex, large-scale models

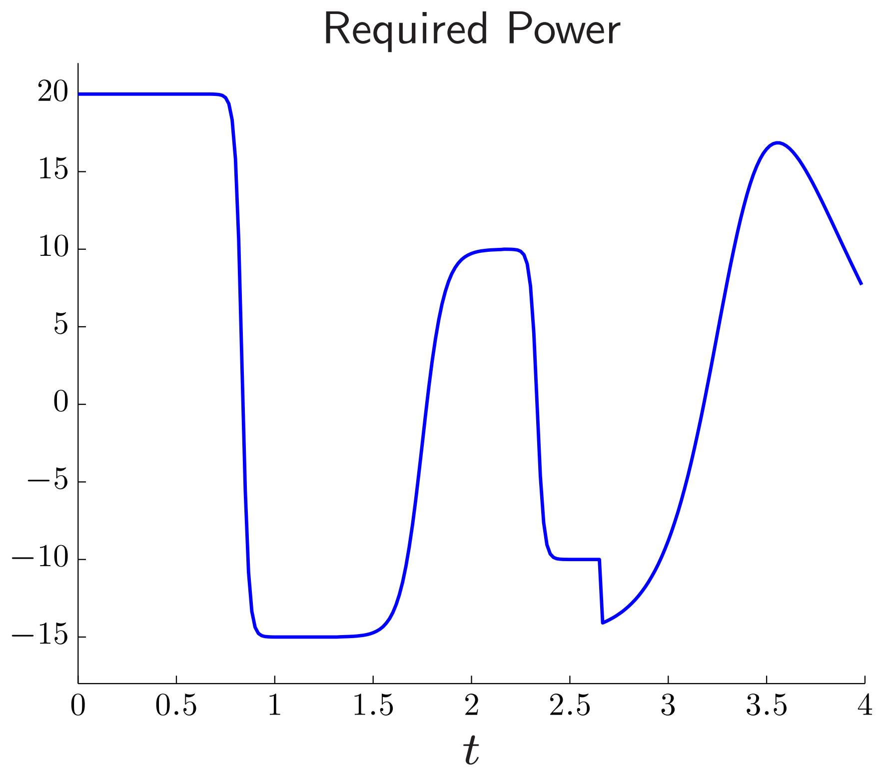
Hybrid vehicle power scheduling

- simplified model of parallel hybrid vehicle
- time varying required power at wheels
- objective: minimize fuel consumption subject to limits on engine/motor power, battery capacity

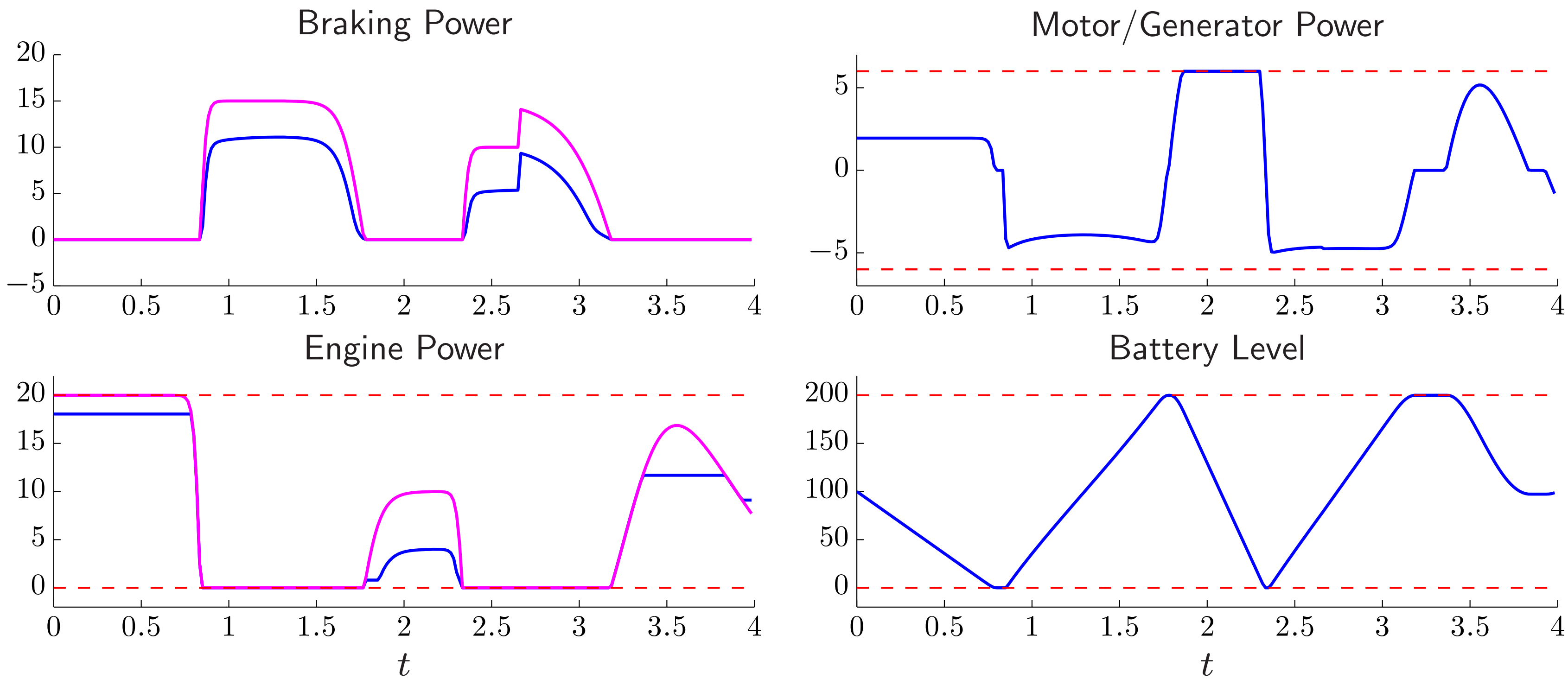


Example

- required power (computed from speed, road slope, and losses)



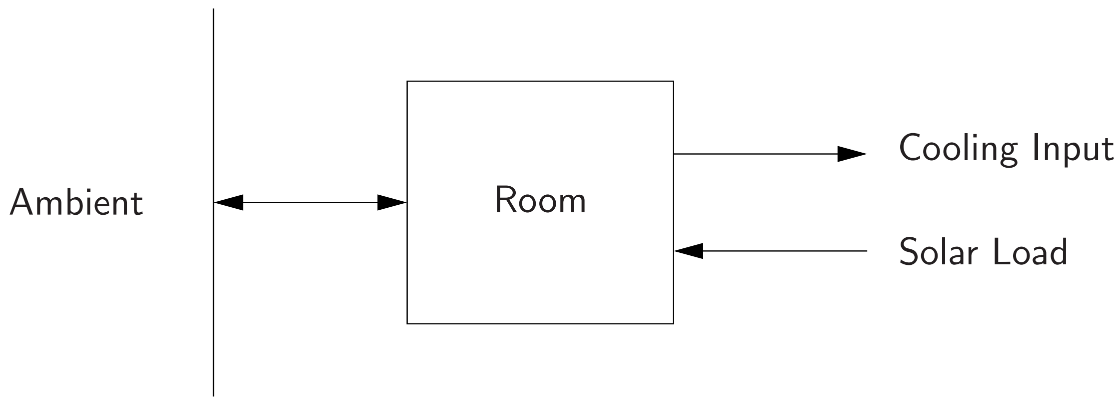
Example



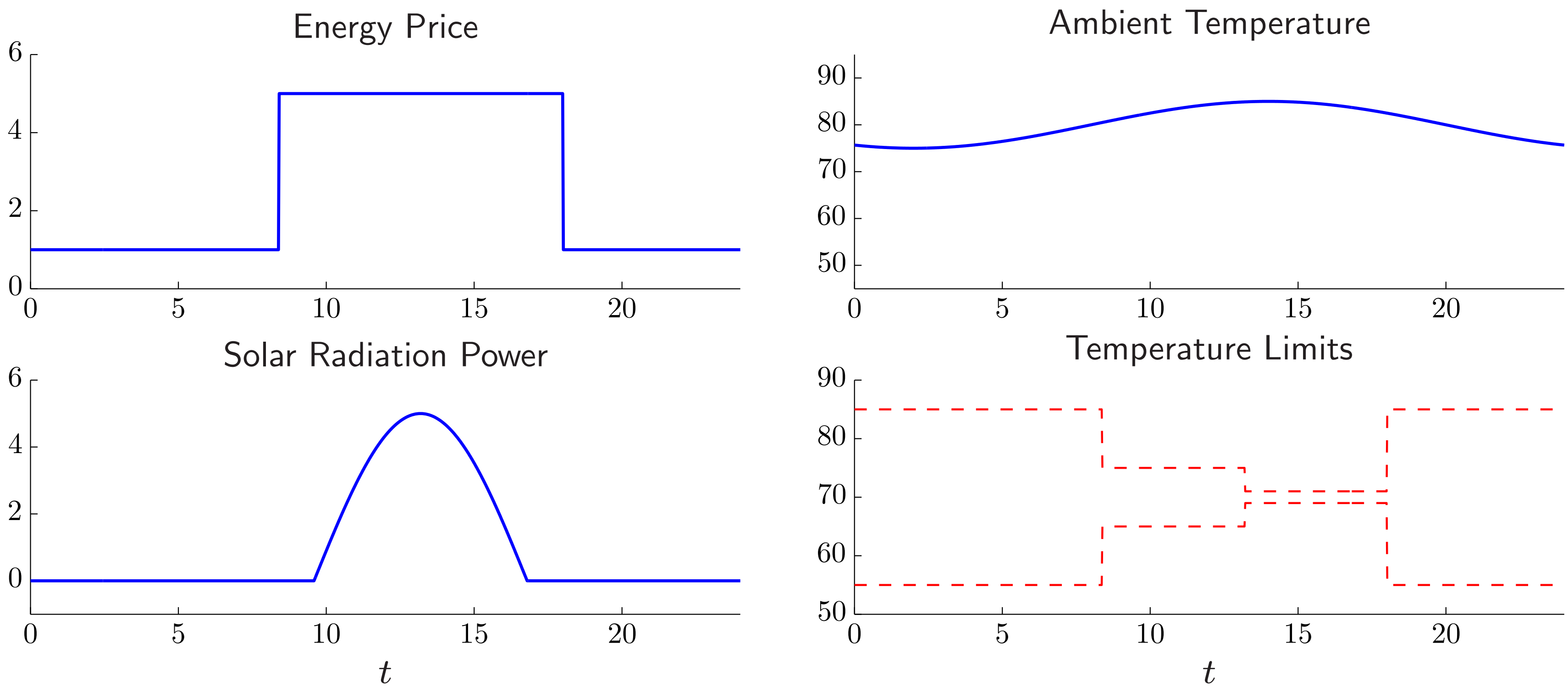
- blue: hybrid vehicle; magenta: without battery
- energy savings: 25%

HVAC Control

- single room with temperature sensor, conduction to outside, solar load
- time-varying solar load, outside temperature, temperature limits, electricity price
- find cooling schedule that minimizes energy cost, while keeping temperature within limits

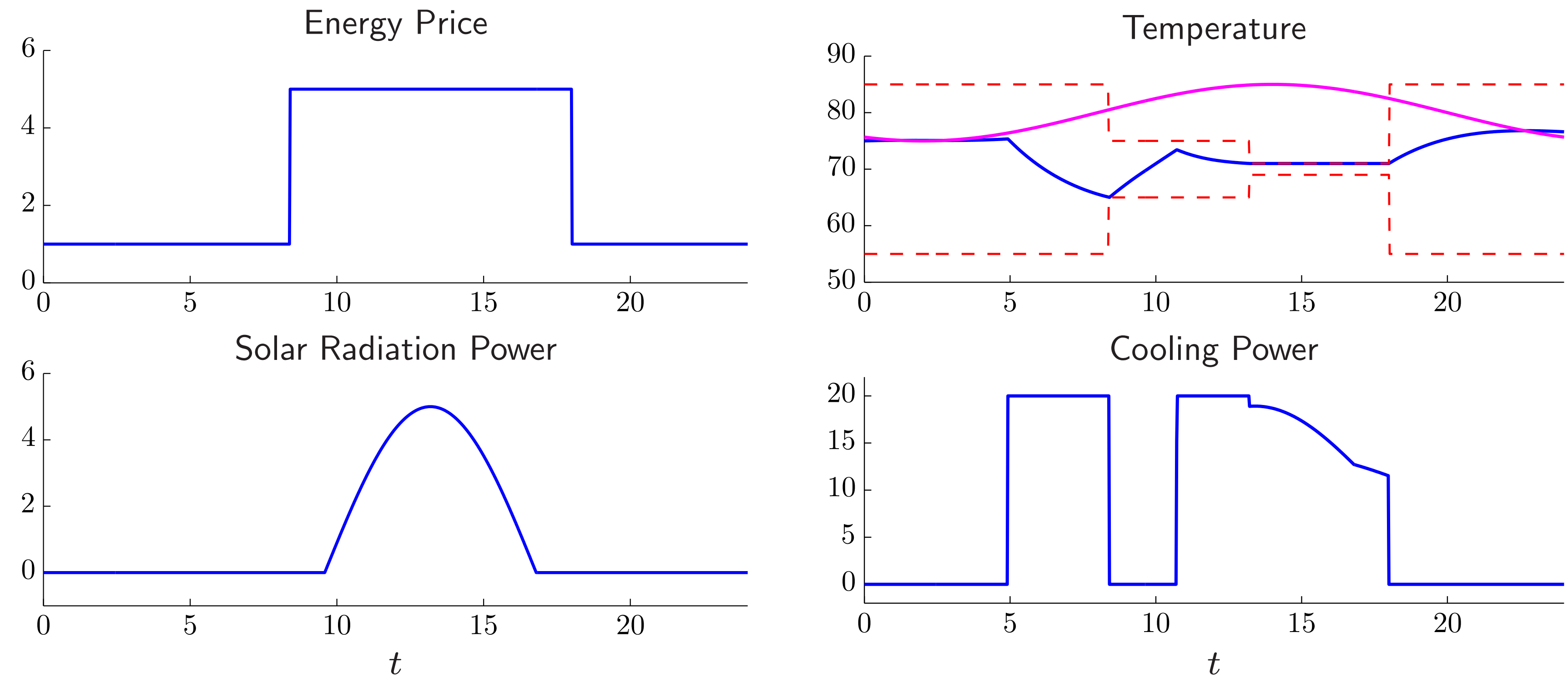


Example



Energy Price
Ambient Temperature
Solar Radiation Power
Temperature Limits
0
2
4
6
50
60
70
80
90
5
10
15
20
t
t

Results



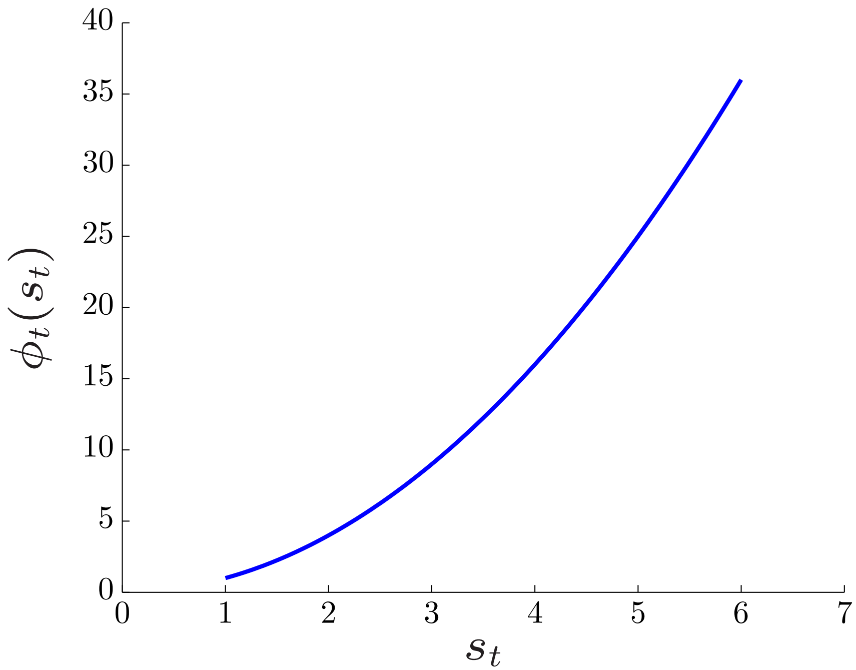
- magenta: ambient temperature; blue: room temperature
- optimal action: pre-cool the room when energy price is low

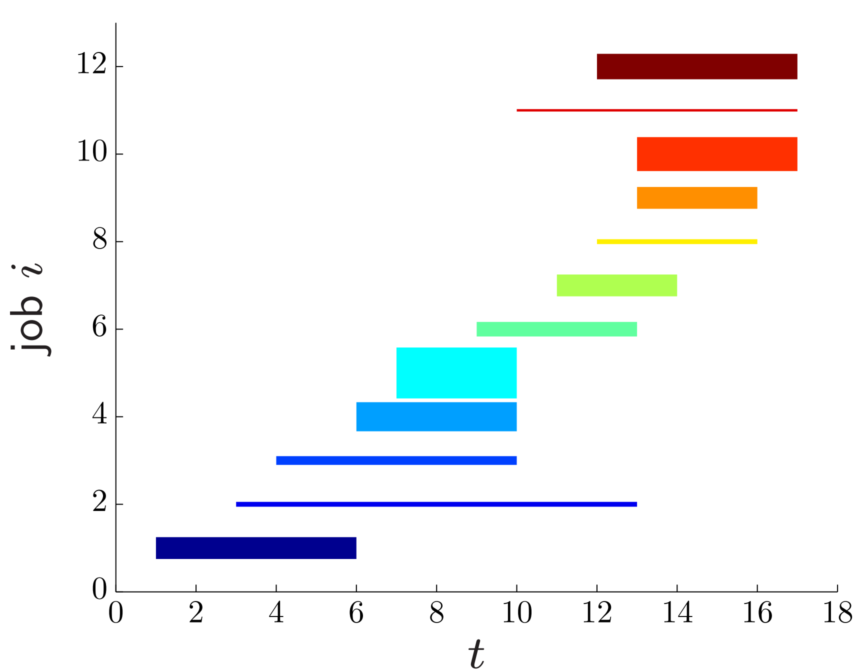
Multi-period processor speed scheduling

- processor adjusts its speed $s_t \in [s^{\min}, s^{\max}]$ over T time periods
- must execute n jobs with known arrival times and deadlines
- energy consumed in period t is $\phi(s_t)$; total energy is $E = \sum_{t=1}^{T} \phi(s_t)$
- objective: minimize total energy consumed subject to completion of jobs, processor speed limits

Example

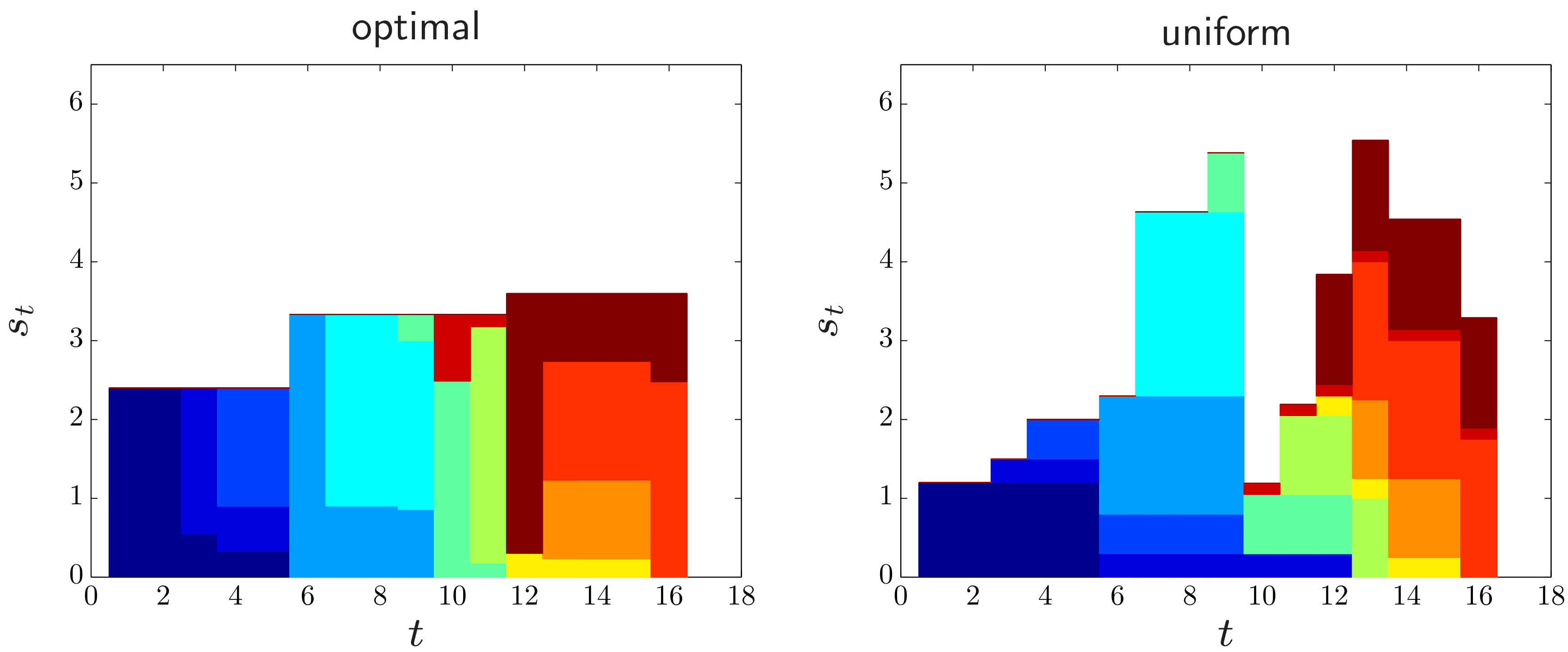
- $T = 16$ periods, $n = 12$ jobs
- $s^{\min} = 1$, $s^{\max} = 6$, $\phi(s_t) = s_t^2$
- jobs shown as bars over $[A_i, D_i]$ with area proportional to workload





Optimal and uniform schedules

- uniform schedule gives $E^{\mathrm{unif}} = 194.2$
- optimal schedule gives $E^\star = 160.3$

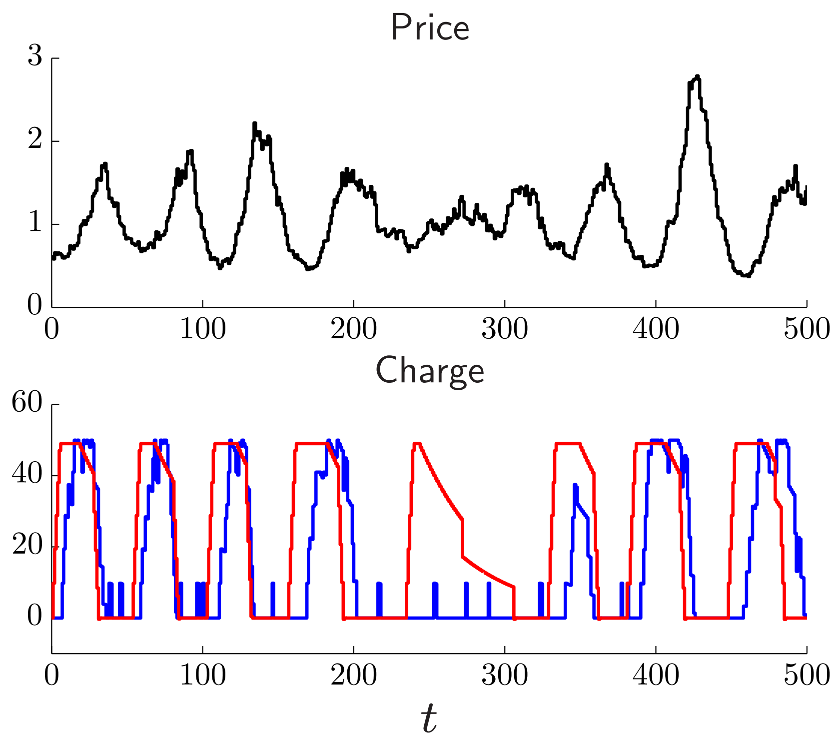


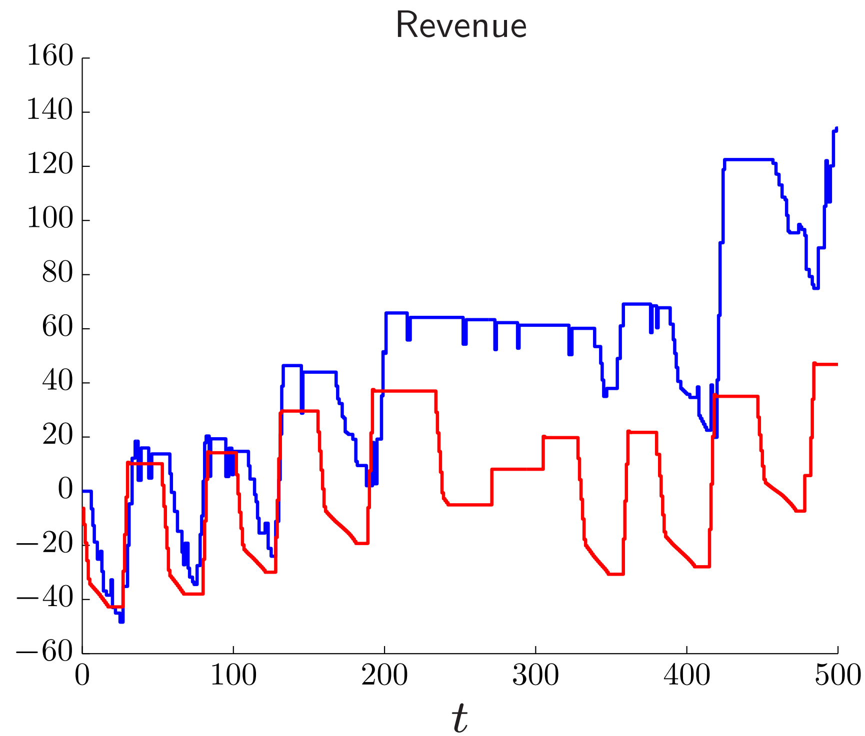
Energy storage control

- charge/discharge battery with varying, uncertain electricity price
- we pay to charge the battery; we are paid for discharging
- charging/discharging incurs a transaction cost
- profit is revenue minus transaction cost
- maximize profit subject to constraints on battery capacity, charge/discharge rates, . . .

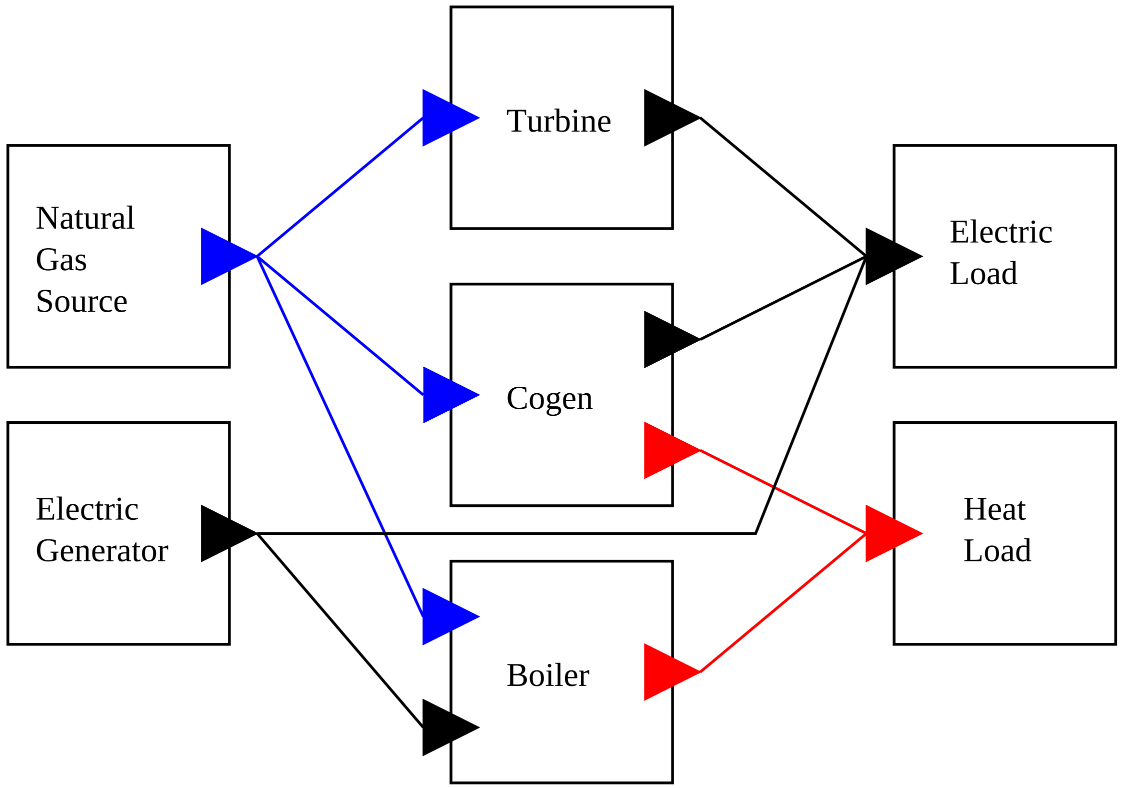
Example

- blue: receding horizon policy; red: thresholding policy





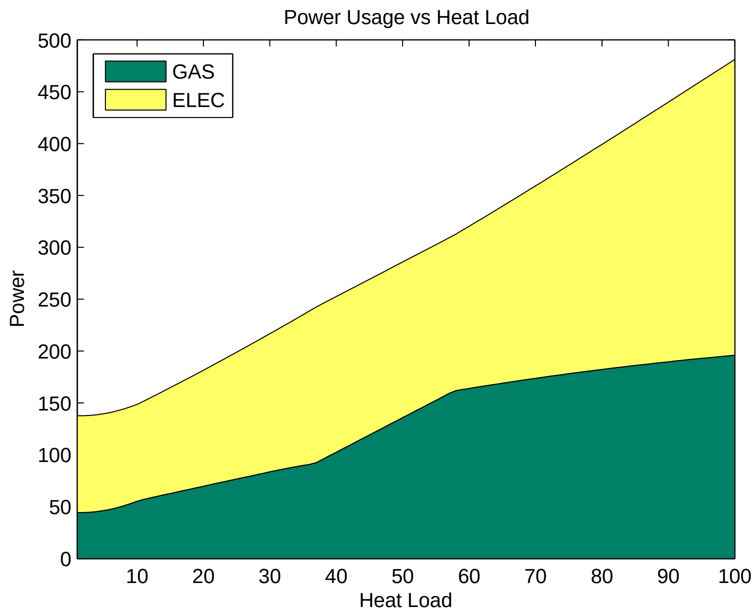
Multi-carrier energy system

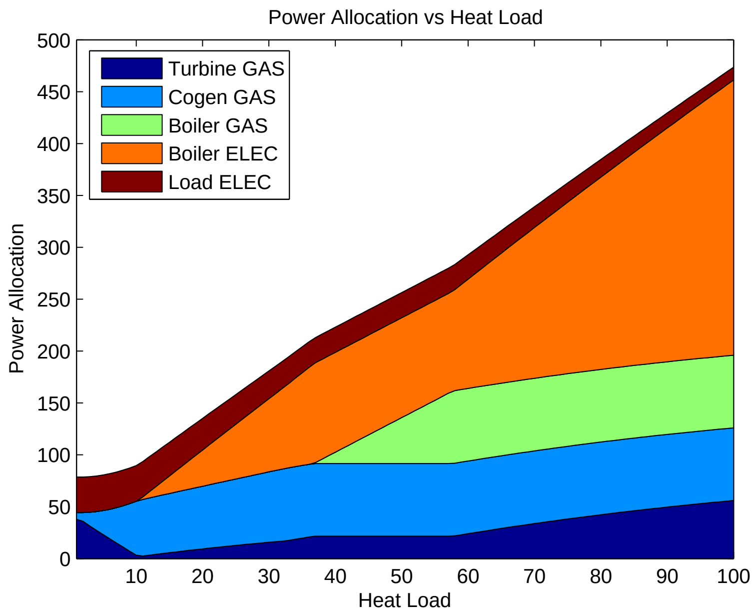


Multi-carrier energy system

- electric load and heat load must be met by combination of turbine, cogen, generator, and boiler
- all have (nonlinearly varying) efficiencies, capacities
- fixed gas price
- goal: minimize operating cost

Optimal operation





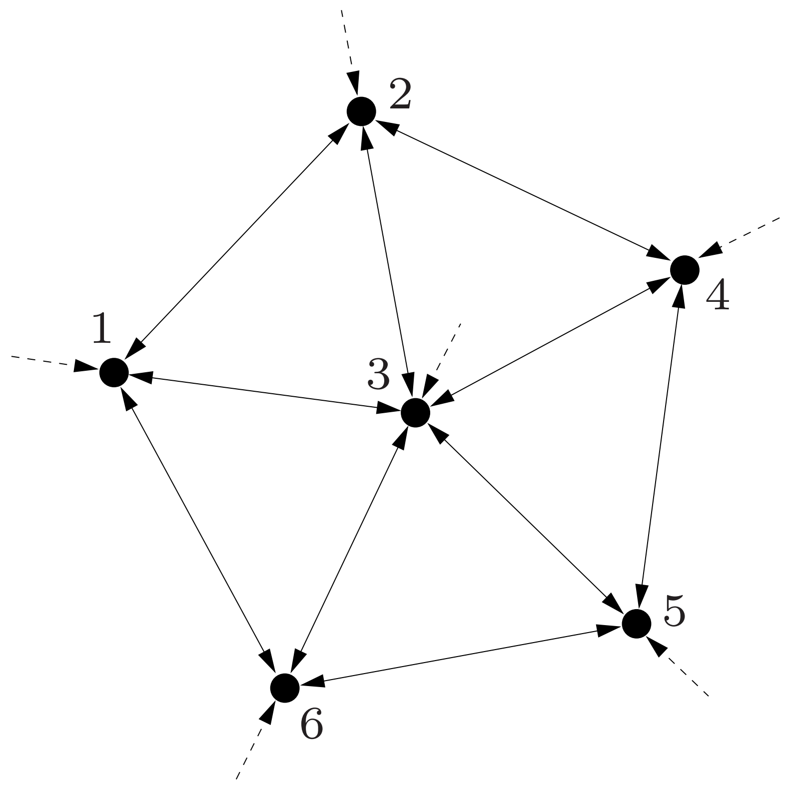
- optimal operation with fixed electric load, varying heat load
- results plausible, but not obvious

Dynamic load balancing

- n nodes (buffers/queues)
- m bidirectional links (for shipping between nodes)
- random arrivals of jobs at each node
- linear shipping cost, quadratic processing cost
- linear $+$ quadratic buffering cost
- minimize cost subject to constraints on shipping/processing capacities

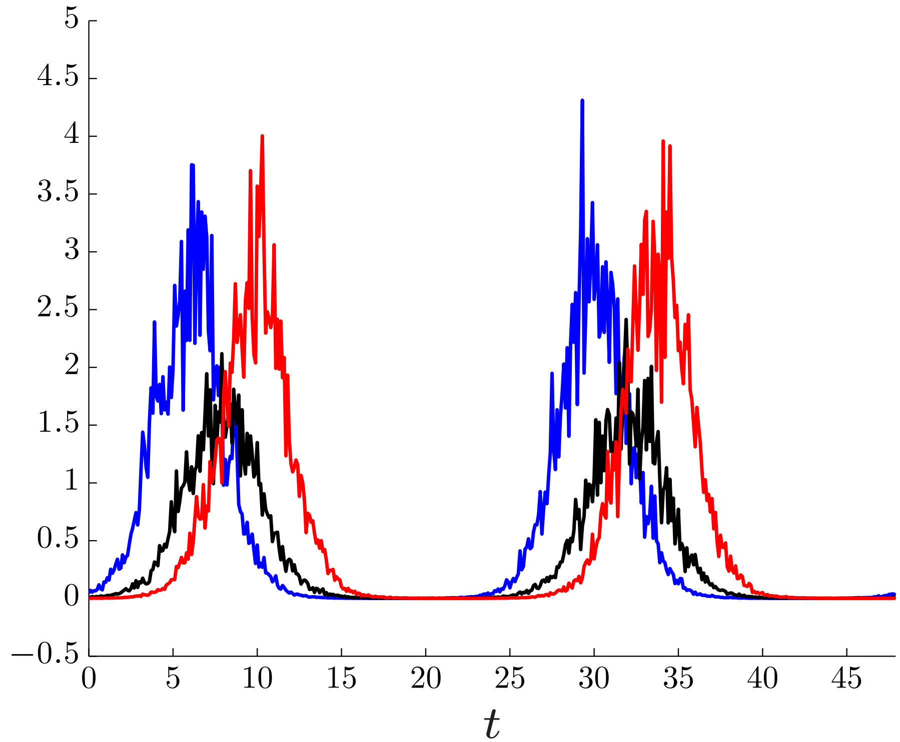
Example

- example with 6 nodes, 10 bidirectional links

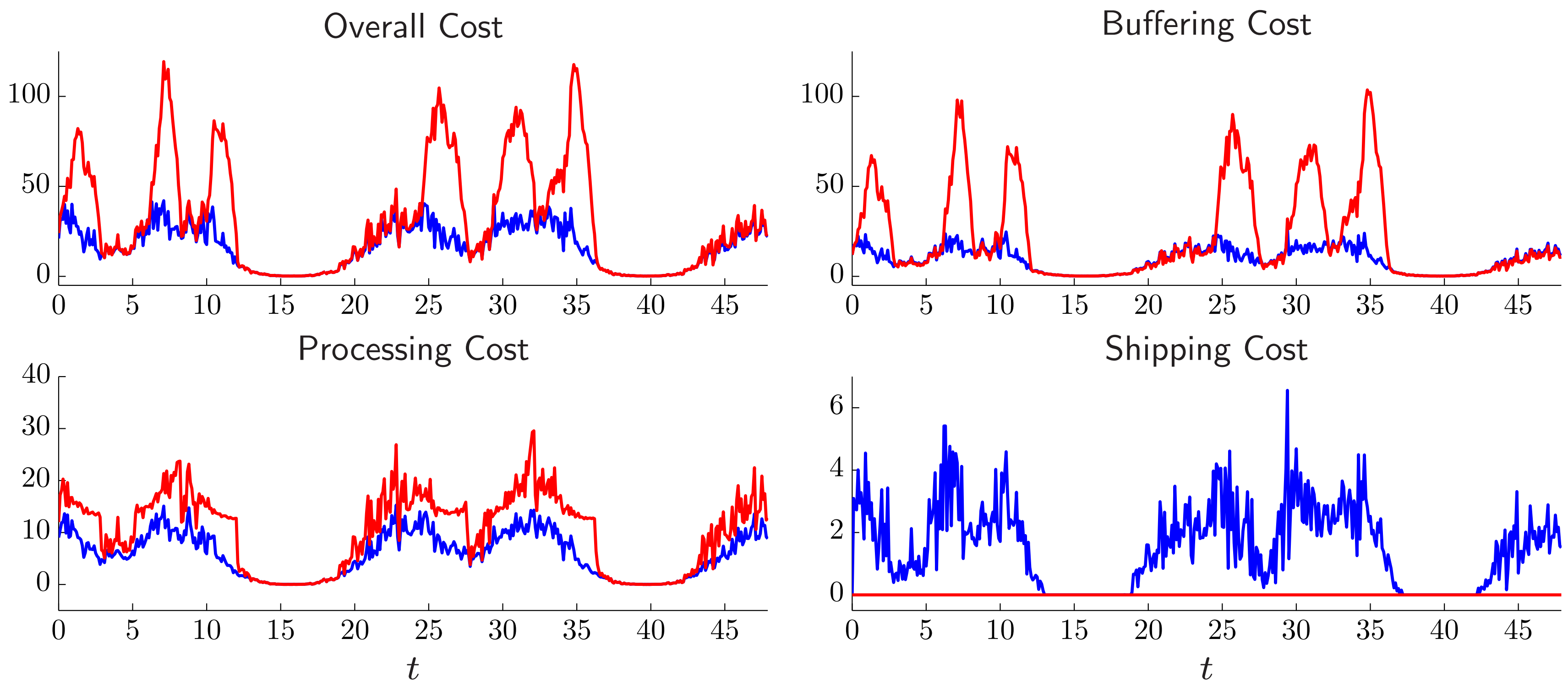


Example

- typical arrivals trajectories; blue: queue 1, black: queue 2, red: queue 3

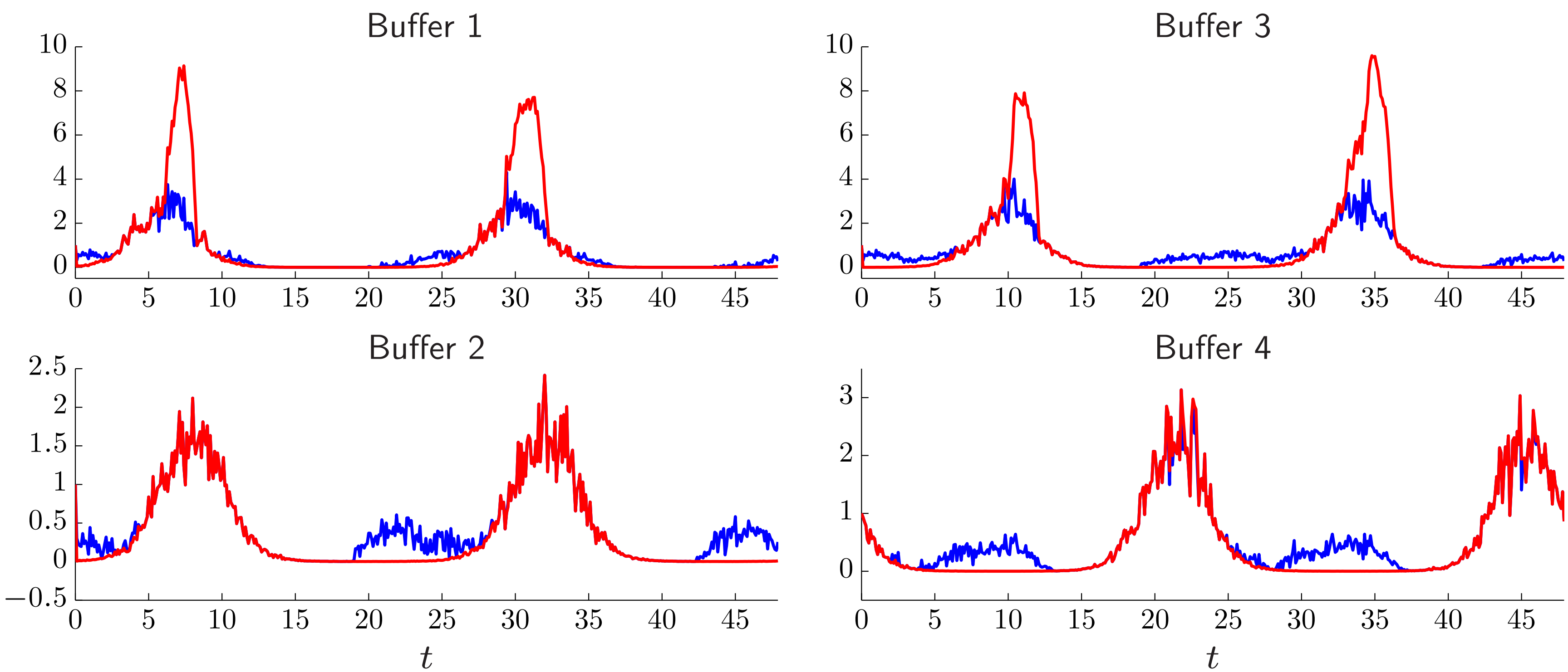


Example



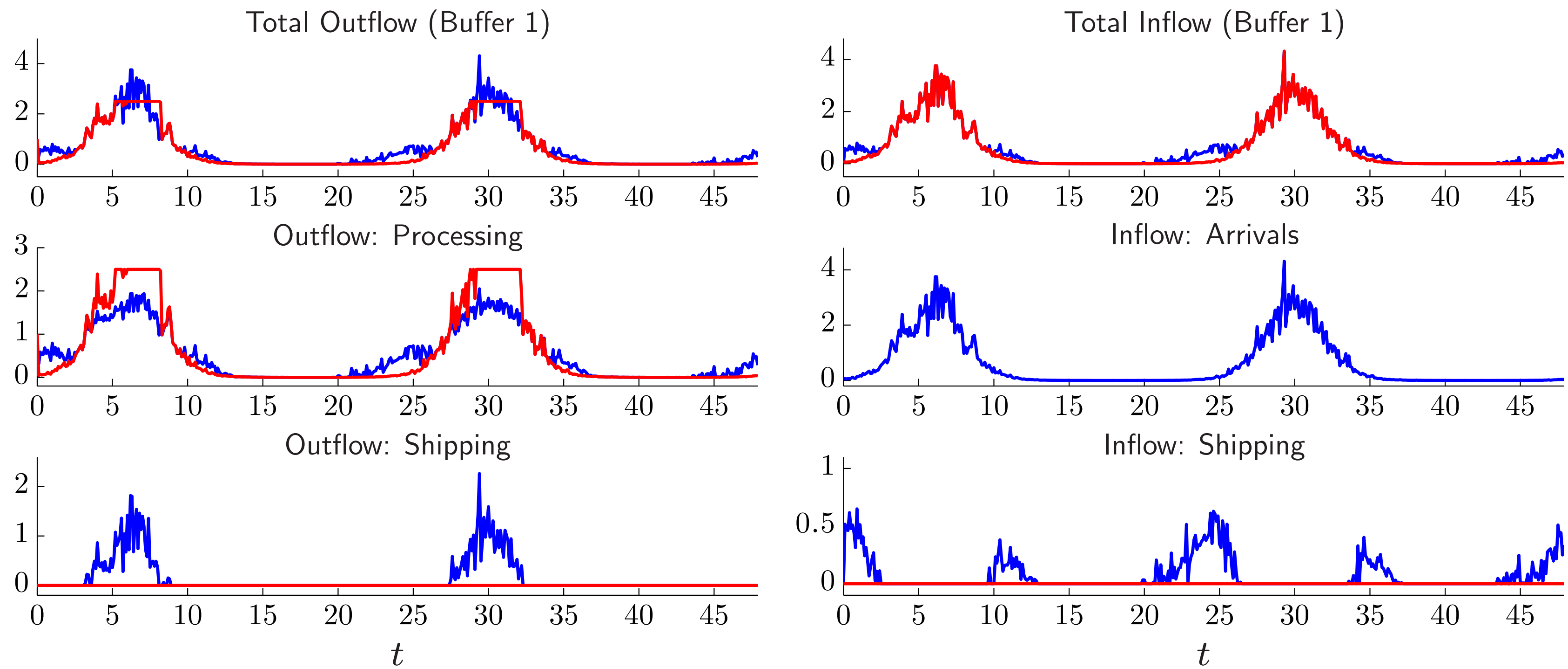
- blue: RHC; red: proportional policy (without shipping)

Example



- blue: RHC; red: proportional policy (without shipping)

Example



- blue: RHC; red: proportional policy (without shipping)

Conclusions

optimization (and control)

- comes up in many smart grid contexts
- has been used in large complex applications with
  - slow dynamics
  - big, expensive computers (with staff)

  ($e.g.$, dispatch, refining)
- can be used in smaller applications, with fast dynamics
- should be a core technology in providing **automated, smart operation**