Dynamic Network Utility Maximization

Stephen Boyd   Argyrios Zymnis   Dan O'Neill   Andrea Goldsmith
Electrical Engineering Department, Stanford University

Network Utility Maximization

$$\begin{array}{ll} \text{maximize} & U(f) \\ \text{subject to} & Rf \leq c, \quad f \geq 0 \end{array}$$

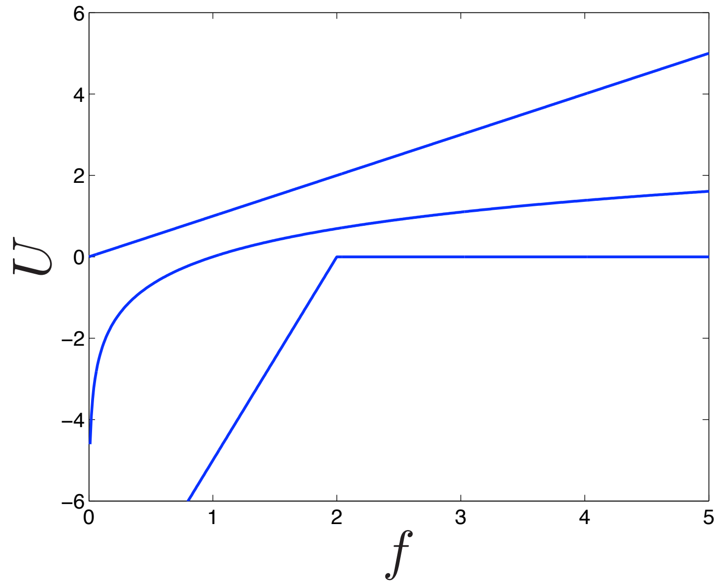
with variable f

- $f = (f_1, \ldots, f_n)$ is vector of flow rates
- $U(f) = \sum_{i=1}^n U_i(f_i)$ is (separable) utility function
- $R \in \mathbf{R}^{m \times n}$ is routing matrix
- $c \in \mathbf{R}^m$ is link capacity vector

Network Utility Maximization

- a resource allocation problem
- convex problem if U_i are concave
- can solve via distributed iterative methods (dual decomposition)
- utility function U_i models utility derived from flow f_i
- **single period; no concept of time**
- if c (or U_i) 'change', iterative methods will 'adjust' f

Typical Utility Functions



- *best effort* (linear): $U(f) = wf$ ($w > 0$ is weight)
- *diminishing returns* (logarithmic): $U(f) = \log f$
- *contract with penalty* (piecewise linear): $U(f) = u_c - p(f_c - f)_+$
  u_c is contract utility; $(f_c - f)_+$ is shortfall; $p > 0$ is penalty

Dynamic Network Utility Maximization

now we're going to explicitly add the concept of time

$$\begin{array}{ll} \text{maximize} & U(f(1),\ldots,f(T)) \\ \text{subject to} & R(t)f(t) \leq c(t), \quad f(t) \geq 0, \quad t = 1,\ldots,T \end{array}$$

- $f(t) \in \mathbf{R}^n_+$ is vector of flow rates at time t
- $R(t)$, $c(t)$ are routing matrix, capacity vector at time t
  - capacity limits must hold at each time (no buffering)
  - captures time-varying network topology, link state, . . .
- we assume $U = \sum_i U_i(f_i(1),\ldots,f_i(T))$ is separable across flows *but not time*

Dynamic Network Utility Maximization

- a multi-period resource allocation problem
- convex problem if U_i are concave
- can solve by distributed iterative methods (dual decomposition)
  these are not obvious
- utility function U_i models utility derived from flow *sequence* $f_i(1), \ldots, f_i(T)$
- if U_i are also separable in time, can solve DNUM as T separate NUMs, once for each t

Typical (Dynamic) Utility Functions

- *best effort*: $U(f(1), \ldots, f(T)) = \sum_t w(t) f(t)$
  ($w(t)$ are possibly time-varying weights)
- *file transfer*: need total flow S over period $[t_i, t_f]$

  $$U(f(1), \ldots, f(T)) = -p\left(S - (f(t_i) + \cdots + f(t_f))\right)_+$$

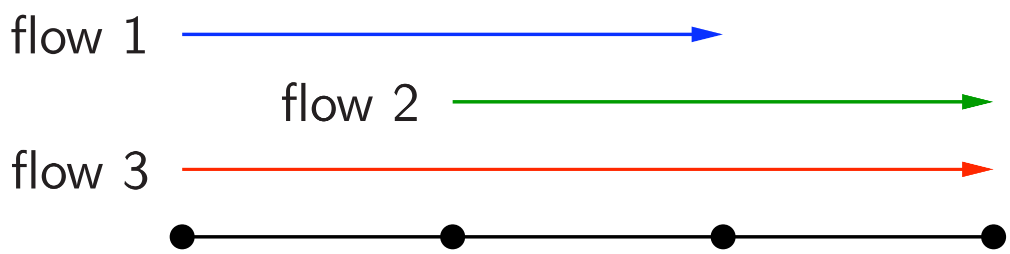
  assesses (linear) penalty for shortfall
- *streaming*: need total flow S for successive k-long periods

  $$\begin{aligned} U(f(1), \ldots, f(T)) &= -p\left(S - (f(1) + \cdots + f(k))\right)_+ \\ &\quad -p\left(S - (f(k+1) + \cdots + f(2k))\right)_+ \\ &\quad \vdots \\ &\quad -p\left(S - (f(T-k+1) + \cdots + f(T))\right)_+ \end{aligned}$$

Typical (Dynamic) Utility Functions

- these utility functions *cannot* be represented in time-separable form
- they capture what the applications need *much better* than time-separable utilities

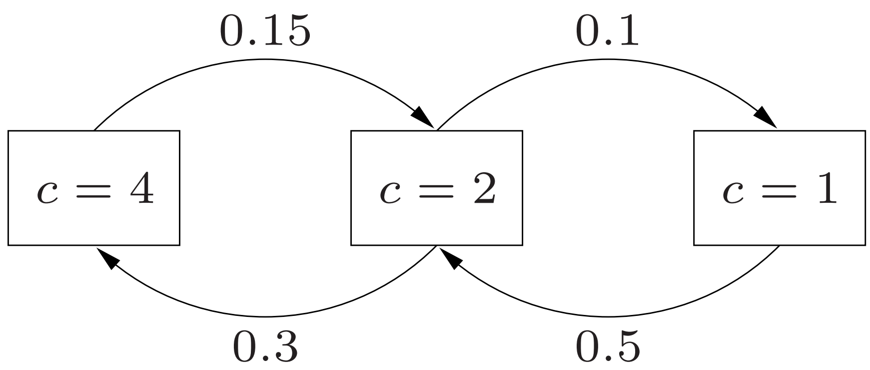
Example



- $T = 50$ horizon
- $c(t)$ is Markov
- 3 file transfers, with (linear) shortfall penalty

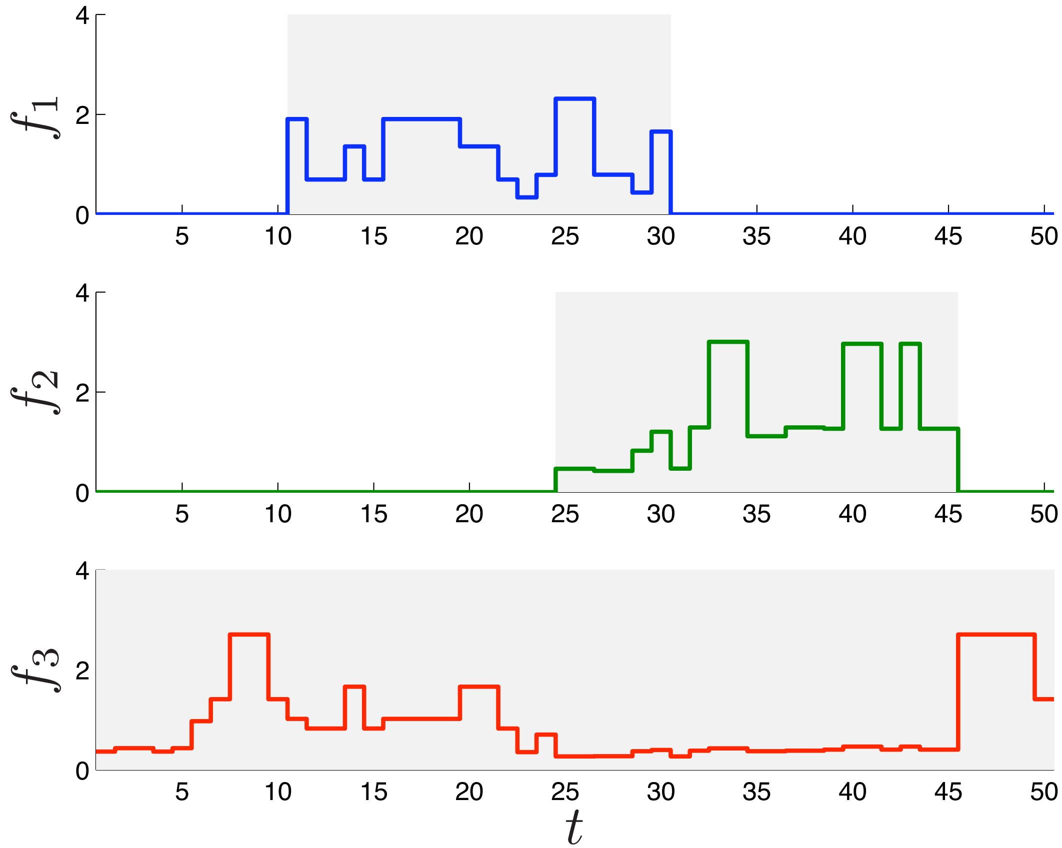
flow	start time t_i	stop time t_f	file size S
1	11	30	25
2	25	45	30
3	1	50	45

Markov Link Capacity Model



- three states: good ($c = 4$), OK ($c = 2$), bad ($c = 1$)
- link capacities evolve independently
- mixing time about 5 periods
- equilibrium distribution is 0.6, 0.3, 0.1; average capacity is $\bar{c} = 3.2$
- all links start in OK state

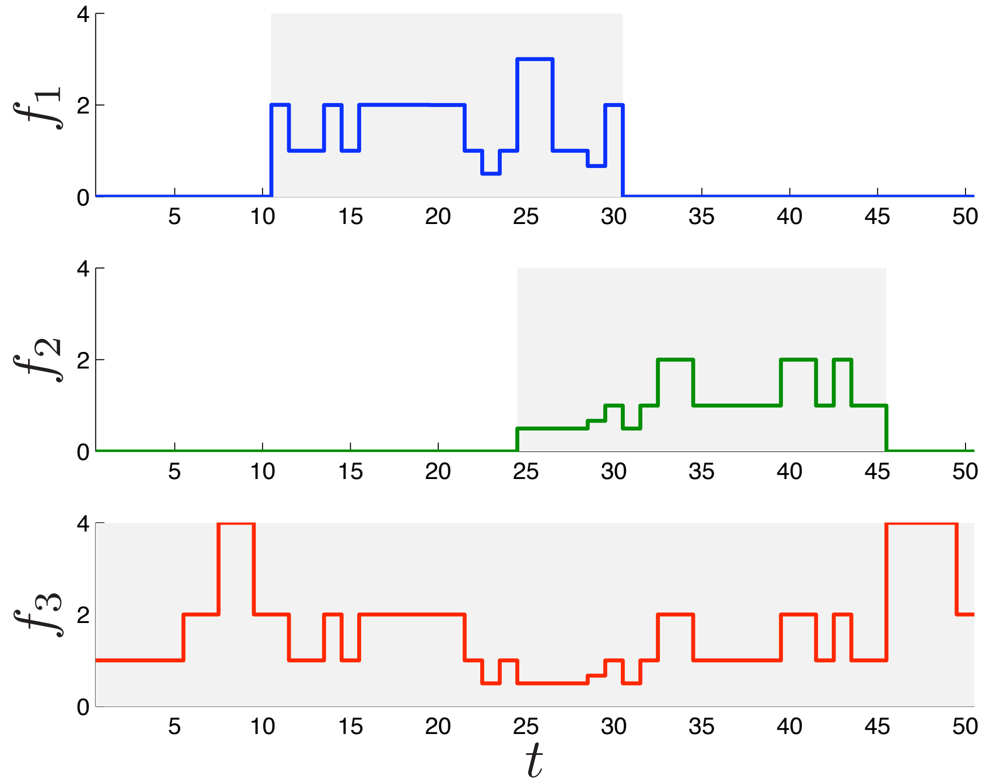
Optimal Flow Rates



shortfalls: 0, 0, 0; total penalty: 0

Flow Rates from (Separable) Log Utility

U is log utility over contract periods



shortfalls: 0, 6.8, 0; total penalty: 6.8

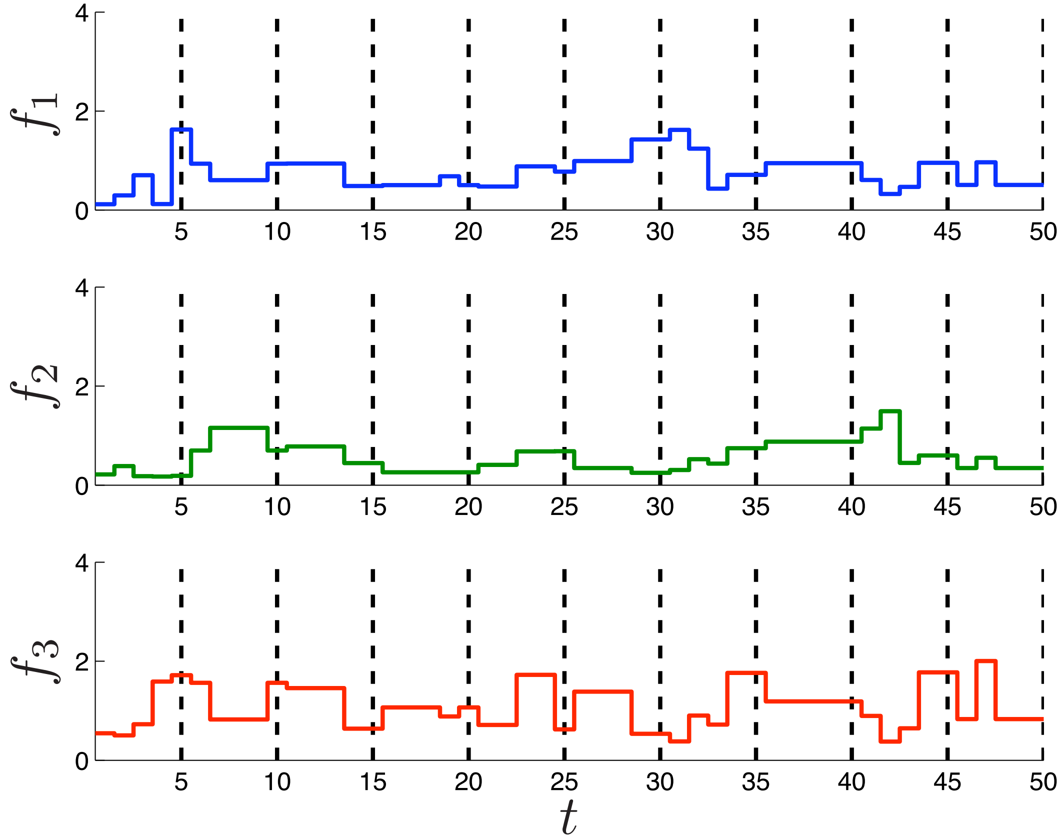
Streaming

- need $S = 1,\ 3,\ 2$ total flow (for $f_1,\ f_2,\ f_3$) in each of 10 successive 5-period long blocks

- we'll compare optimal flows with flows from (separable) log utility

- we'll judge by total penalty, fraction of block contract violations

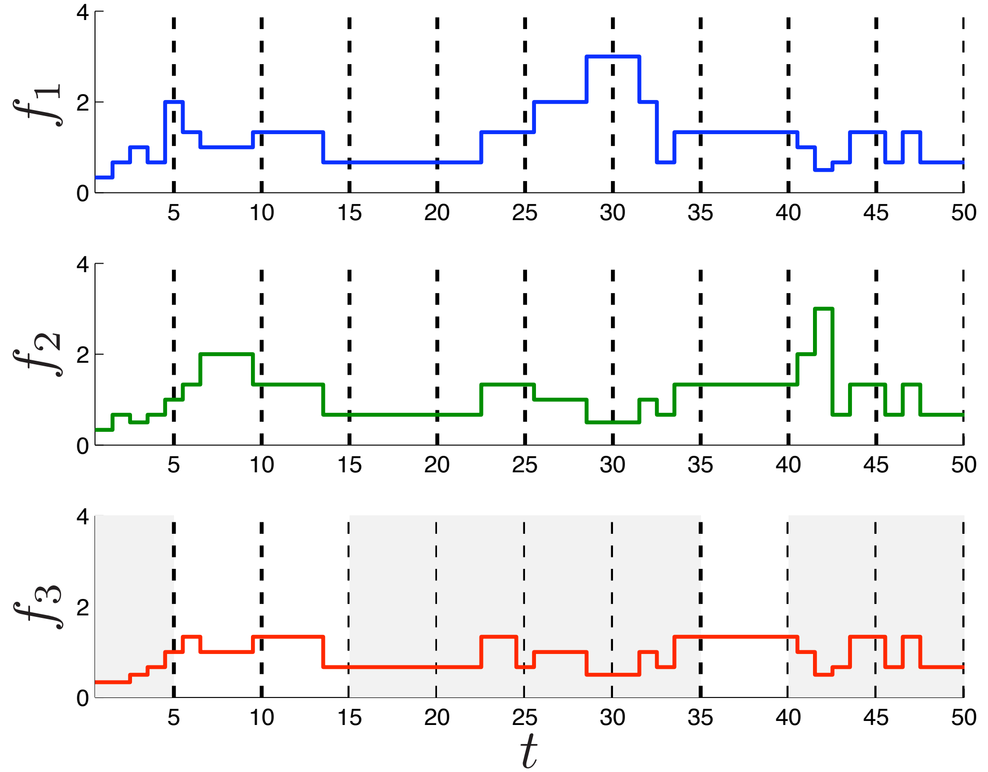
Optimal Flows



0 block shortfalls (out of 30); total penalty: 0

Log Utility Flows

$U = \sum_i \sum_t \log f(t)_i$



7 block shortfalls (out of 30); total penalty: 6.5

Stochastic Dynamic NUM

- so far, we've assumed *future* $c(t)$, $R(t)$, U are *known*
- this is the *prescient* model
- now suppose $c(t)$ not perfectly known ahead of time
- we'll let $\hat{c}(t|\tau)$ be guess of $c(t)$ at time τ; for $\tau \geq t$, $\hat{c}(t|\tau) = c(t)$
- let's impose *causality constraint*: $f(t)$ can only depend on $c(1), \ldots, c(t)$
- DNUM then reduces to (convex) *stochastic control problem*
  (with statistical model of c)

- much known about stochastic control
- prescient solution gives bound on performance of causal scheme
- no analytic solution, but several good heuristics
- model predictive control, a.k.a. rolling horizon control, can work well
- basic idea:
  - solve a DNUM problem at each step, using predictions for unknown future value
  - implement/execute only first value of f

Model Predictive Control

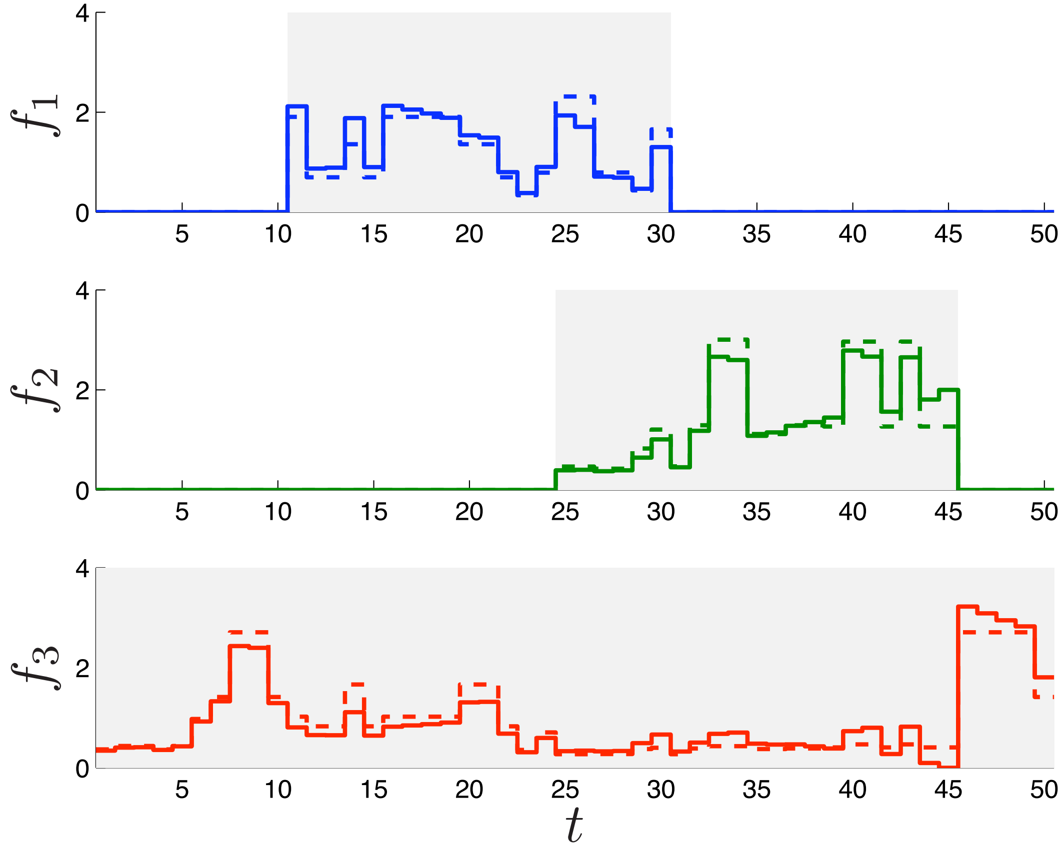
- let $f_{\mathrm{mpc}}(t)$ denote MPC flows
- for $\tau = 1, \ldots, T$ get solution $f^\star$ of

$$
\begin{array}{ll}
\text{maximize} & U(f(1), \ldots, f(T)) \\
\text{subject to} & R(t)f(t) \leq \hat{c}(t|\tau), \quad f(t) \geq 0, \quad t = 1, \ldots, T \\
 & f(t) = f_{\mathrm{mpc}}(t), \quad t = 1, \ldots, \tau - 1
\end{array}
$$

- then set $f_{\mathrm{mpc}}(\tau) = f^\star(\tau)$
- $f_{\mathrm{mpc}}(t)$ depends only on $c(1), \ldots, c(t)$, $i.e.$, it is *causal*

Results: Rates for Contracts

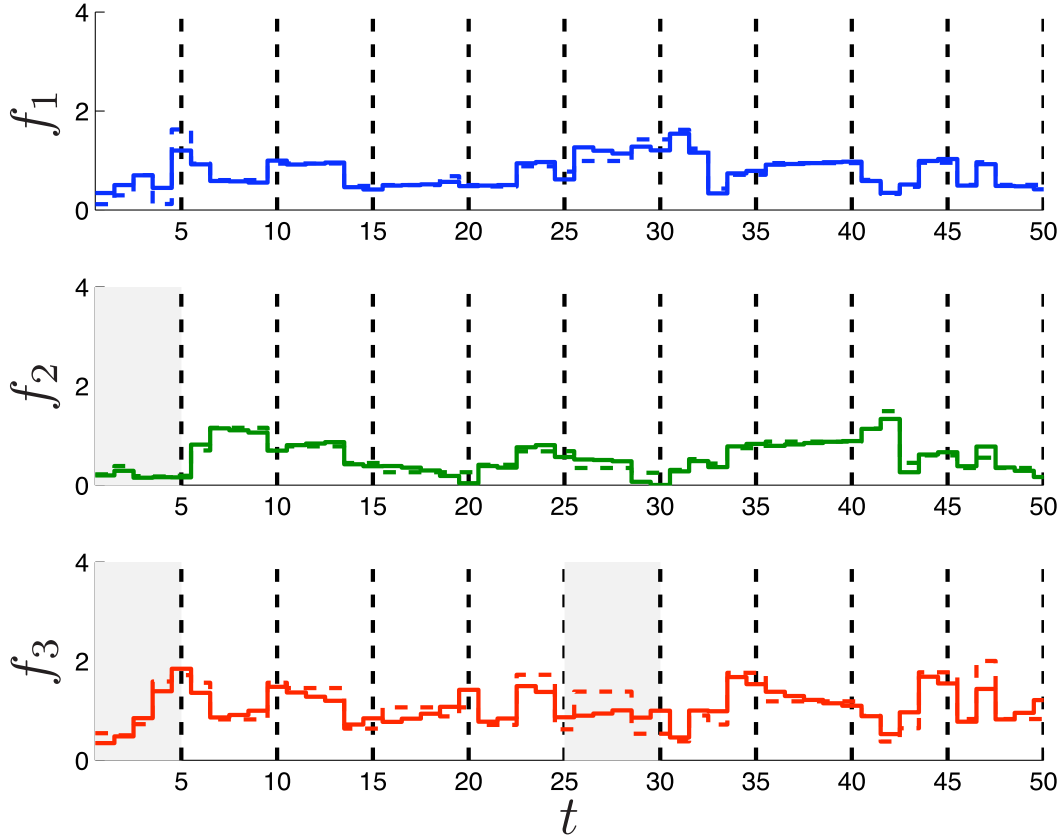
dashed prescient; *solid* MPC



shortfalls: 0, 0.1, 0; total penalty: 0.1

Results: Rates for Streaming

dashed prescient; *solid* MPC



3 block shortfalls (out of 30); total penalty: 0.4

Conclusions

- we think that the explicit idea of time (dynamics) needs to be introduced in the NUM framework
- this allows us to describe different requirements on traffic, urgency, and scheduling in a sensible way
- many static NUM methods extends to DNUM, $e.g.$, dual decomposition
- model predictive control gives causal control law