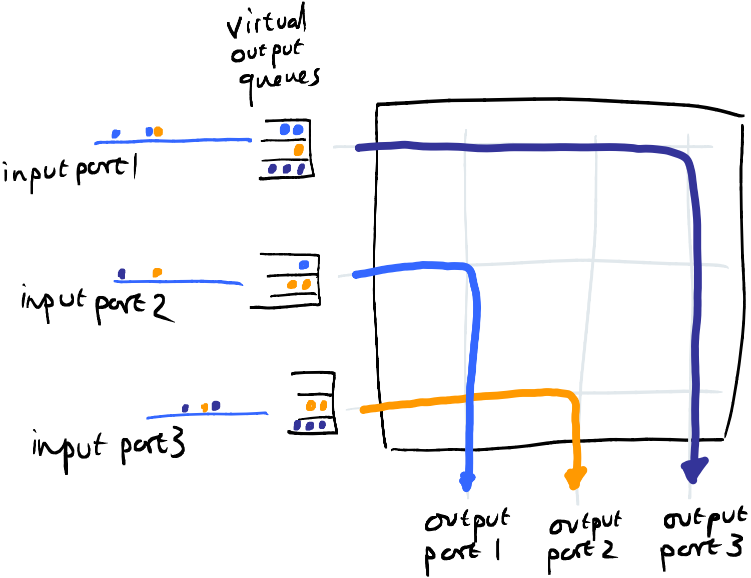
The Teleology of Switched Networks

Damon Wischik, UCL

Devavrat Shah, MIT

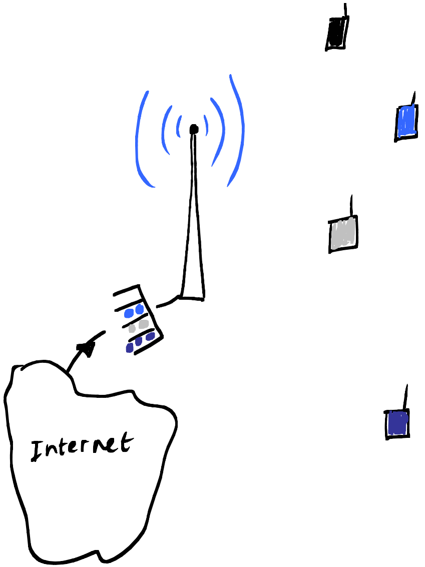
AN INPUT-QUEUED SWITCH



Arriving packets are classified by destination.

Each timestep, the scheduling algorithm chooses a matching from inputs to outputs.

A WIRELESS BASE STATION



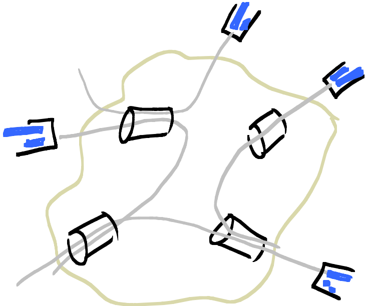
Packets arrive from the Internet, and are queued up at the basestation.

Each timestep, the basestation chooses one queue to serve.

The service rate depends on the state of nature; the basestation knows the current state of nature, and its distribution.

[31] Stolyar, A. L. (2004). MaxWeight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic. *Annals of Applied Probability* 14, 1, 1–53.

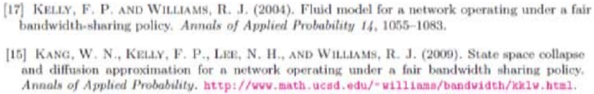
FLOW-LEVEL MODEL OF TCP



Each queue represents the number of active flows on a given route.

TCP chooses transmit rates for each of the flows, a function of the number of active flows on each route.

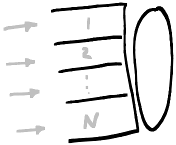
That is, it chooses drain rates for each queue, subject to network capacity constraints.



[17] Kelly, F. P. and Williams, R. J. (2004). Fluid model for a network operating under a fair bandwidth-sharing policy. *Annals of Applied Probability 14*, 1055–1083.

[15] Kang, W. N., Kelly, F. P., Lee, N. H., and Williams, R. J. (2009). State space collapse and diffusion approximation for a network operating under a fair bandwidth sharing policy. *Annals of Applied Probability*. http://www.math.ucsd.edu/~williams/bandwidth/kklw.html.

GENERAL QUEUEING MODEL



Consider a collection of N queues, $\underline{Q} = (Q_1, \cdots, Q_N)$ in slotted time.

Each timestep, work arrives, eg Bernoulli arrivals. Each queue has a dedicated arrival process; arrivals are at rate $\underline{\lambda}$

Each timestep, the switch chooses a schedule $\underline{\pi}(t)$ from a finite set $S \subset \mathbb{R}_+^N$.

It picks $\pi(t) = \underset{p \in S}{\arg\max}\ p \cdot Q^{\alpha}(t)$

i.e. $\sum_{1 \le n \le N} p_n Q_n^{\alpha}(t)$

for some fixed $\alpha > 0$.

PRIMAL(λ)

minimize $\sum_{\pi \in S} \alpha_\pi$

over $\alpha \geq 0$

suchthat $\lambda \leq \sum_{\pi \in S} \alpha_\pi \pi$

This is what an omniscient scheduler might solve: find a mixture of schedules that can serve the arriving traffic.

If PRIMAL(λ) ≤ 1, this is possible.

DUAL(λ)

maximize $\xi \cdot \lambda$

over $\xi \in \mathbb{R}_+^N$

such that $\xi \cdot \pi \leq 1$ for all $\pi \in S$.

$\xi \cdot Q(t)$ represents a virtual queue; it gets ξ_n tokens whenever a packet arrives at queue n. The maximum possible drain rate of tokens is 1/timeslot. On average, $\xi \cdot \lambda$ arrive per timeslot.

If DUAL(λ) > 1, the switch must be unstable.

THE FLUID SCALE

From the discrete quantities

$Q(t)$ — queue sizes at time t

$A(t)$ — arrivals over time interval $\{1,2,\dots,t\}$

$S_\pi(t)$ — number of timesteps in which π was chosen, in that interval

$Z(t)$ — amount of idling at each queue in that interval

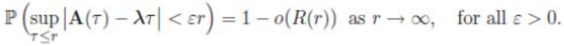
we define fluid-scaled quantities

$$q^r(t) = \frac{Q(rt)}{r} \qquad a^r(t) = \frac{A(rt)}{r} \qquad s^r_\pi(t) = \frac{S_\pi(rt)}{r} \qquad z(t) = \frac{Z(rt)}{r}$$

When r is large, these scaled-quantities lie close to continuous functions $q(\cdot)$, $a(\cdot)$, $s_\pi(\cdot)$, $z(\cdot)$ that satisfy differential equations.

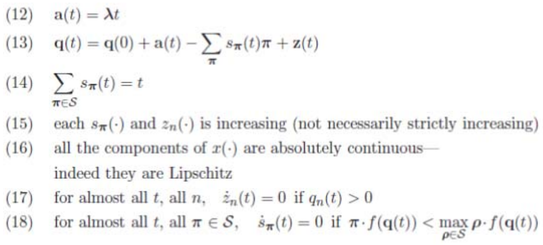
The results in this talk are proved by analysing these diff. eq.

Assume the arrival process satisfies



$$\mathbb{P}\left(\sup_{\tau \le r} |\mathbf{A}(\tau) - \boldsymbol{\lambda}\tau| < \varepsilon r\right) = 1 - o(R(r)) \text{ as } r \to \infty, \quad \text{for all } \varepsilon > 0.$$

Specify the differential equations:



(12) $\mathbf{a}(t) = \boldsymbol{\lambda} t$

(13) $\mathbf{q}(t) = \mathbf{q}(0) + \mathbf{a}(t) - \sum_{\pi} s_{\pi}(t)\pi + \mathbf{z}(t)$

(14) $\sum_{\pi \in \mathcal{S}} s_{\pi}(t) = t$

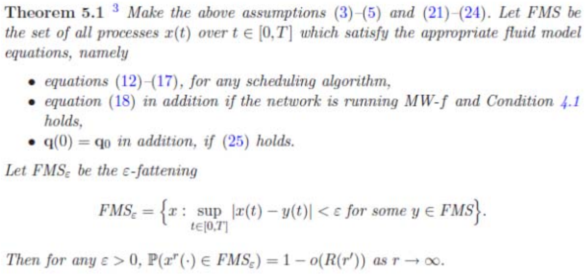
(15) each $s_{\pi}(\cdot)$ and $z_n(\cdot)$ is increasing (not necessarily strictly increasing)

(16) all the components of $x(\cdot)$ are absolutely continuous—
indeed they are Lipschitz

(17) for almost all t, all n, $\dot{z}_n(t) = 0$ if $q_n(t) > 0$

(18) for almost all t, all $\pi \in \mathcal{S}$, $\dot{s}_{\pi}(t) = 0$ if $\pi \cdot f(\mathbf{q}(t)) < \max_{\rho \in \mathcal{S}} \rho \cdot f(\mathbf{q}(t))$

Then, $x^r = (q^r, a^r, s_\pi^r, z^r)$ is close to a solution of those equations:



Theorem 5.1 [3] *Make the above assumptions (3)–(5) and (21)–(24). Let FMS be the set of all processes $x(t)$ over $t \in [0,T]$ which satisfy the appropriate fluid model equations, namely*

- *equations (12)–(17), for any scheduling algorithm,*
- *equation (18) in addition if the network is running MW-f and Condition 4.1 holds,*
- $\mathbf{q}(0) = \mathbf{q}_0$ *in addition, if (25) holds.*

Let FMS_ε be the ε-fattening

$$FMS_\varepsilon = \left\{x : \sup_{t \in [0,T]} |x(t) - y(t)| < \varepsilon \text{ for some } y \in FMS\right\}.$$

Then for any $\varepsilon > 0$, $\mathbb{P}(x^r(\cdot) \in FMS_\varepsilon) = 1 - o(R(r'))$ as $r \to \infty$.

STABILITY

Suppose $\mathrm{PRIMAL}(\lambda) < 1$ hence $\lambda \leq \sum \alpha_\pi \pi$ where $\sum \alpha_\pi < 1$

Define the Lyapunov function $L(q) = \left(1 \cdot q^{1+\alpha}\right)^{1/1+\alpha}$

THEOREM. $\frac{d}{dt} L(q_t) \leq -\eta < 0$ if $q_t \neq 0$.

COROLLARY. If arrivals are IID, so that the queue size process is a Markov chain, then it is +ve recurrent.

ALGP(q)

minimize $L(r)$

over $r \in \mathbb{R}_+^N$

such that $r \geq q + t(\lambda - \sigma)$

for some $\sigma \in \langle \mathcal{S} \rangle$

convex hull of $\mathcal{S}$

ALGD(q)

minimize $L(r)$

over $r \in \mathbb{R}_+^N$

such that $\xi \cdot r \geq \xi \cdot q$

for all $\xi \in \mathcal{S}^*(\lambda)$

extreme optimal solutions to DUAL(λ)

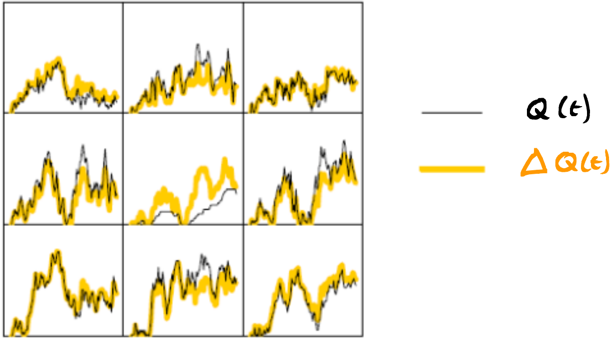
These problems both ask: what is the "cheapest" way to arrange the work in the queues, given that we started at q?

The problems are identical, and have a unique solution, $\Delta(q)$.

In a critically-loaded switch, the state is nearly always in

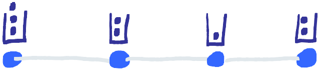
$$\mathcal{I} = \{ q : q = \Delta(q) \}.$$

Example: a 3x3 input-queued switch



$$\lambda = \begin{pmatrix} 0.1413 & 0.4626 & 0.3910 \\ 0.4626 & 0.0055 & 0.5268 \\ 0.3910 & 0.5268 & 0.0771 \end{pmatrix}$$

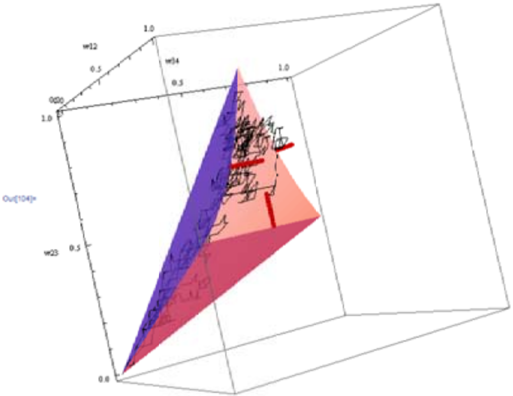
Example: a 4-hop wireless link



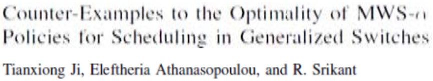
let $\lambda = (.3, \; .7, \; .3, \; .7)$

$$S = \{ (1,0,1,0), \; (0,1,0,1), \; (1,0,0,0) \}$$

$$S^* = \{ (1,1,0,0), \; (0,0,1,1), \; (0,1,1,0) \}$$



In all the examples I've looked at, the bigger Σ the better the performance.



Counter-Examples to the Optimality of MWS-α Policies for Scheduling in Generalized Switches

Tianxiong Ji, Eleftheria Athanasopoulou, and R. Srikant

ALGD$^+$

minimize $L(q)$

over $q \in \mathbb{R}_+^N$

such that $\xi \cdot q \geq \xi \cdot \lambda - 1$

for all dual-extreme ξ

such that $\xi \cdot \lambda > 1$

RATE$^+$

maximize $\lambda \cdot q^\alpha - \max_\pi \pi \cdot q^\alpha$

over $q \in \mathbb{R}_+^N$

such that $L(q) = 1$

let q^+ be the unique solution to ALGD$^+$.

THEOREM If the switch is overloaded then $\frac{q(t)}{t} \to q^+$ for any fluid model solution $q(t)$.

The ALGD$^+$ problem is: find the growth rate for $q(t)$ that incurs cost (measured by $L(q(t))$) as slowly as possible.

[8] Egorova, R., Borst, S., and Zwart, B. (2007). Bandwidth-sharing networks in overload. *Performance Evaluation 64*, 978–993.

Example: 2x2 input-queued switch with $\alpha = 1$

$$\lambda^{\text{critical}} = \begin{pmatrix} 0.3 & 0.7 \\ 0.7 & 0.3 \end{pmatrix}$$

$$\text{service rate} = \begin{pmatrix} 0.3 & 0.7 \\ 0.7 & 0.3 \end{pmatrix}$$

$$q(t) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\text{departure rate} = 2$$

$$\lambda^{\text{overload}} = \begin{pmatrix} 0.3 & 1.0 \\ 0.7 & 0.3 \end{pmatrix}$$

$$\text{service rate} = \begin{pmatrix} 0.2 & 0.8 \\ 0.8 & 0.2 \end{pmatrix}$$

$$q(t) = \begin{pmatrix} 0.1t & 0.2t \\ 0 & 0.1t \end{pmatrix}$$

$$\text{departure rate} = 1.9$$

This switch has the paradoxical behaviour of serving <u>less</u> traffic the greater the arrival rate.

HEURISTIC FOR LDP FOR UNDERLOADED SWITCH

$$\mathbb{P}\left(L(Q(rT)) \approx r \right) = \mathbb{P}\left(L(q^r(T)) \approx 1 \right)$$

since L linear

$$\approx \mathbb{P}\left(a^r(\cdot) \in \left\{ a(\cdot) : \begin{array}{l} \text{queue fed by} \\ a(\cdot) \text{ reaches} \\ L(q(T)) = 1 \end{array} \right\} \right)$$

rearranging

$$\approx \sup_{a(\cdot):\, L(q(T))=1} \mathbb{P}\left(a^r(\cdot) \approx a \right)$$

Principle of the Largest Term

$$\frac{1}{r} \log \mathbb{P}\left(L(Q(rT)) \approx r \right) \approx \sup_{a(\cdot):\, L(q(T))=1} \frac{1}{r} \log \mathbb{P}\left(a^r(\cdot) \approx a \right)$$

rearranging

$$\approx - \inf_{a(\cdot):\, L(q(T))=1} \int_{t=0}^{T} \ell(\dot{a}(t))\, dt$$

LDP for arrival process, assuming IID increments.

[10] Vijay G. Subramanian, Tara Javidi, and Somsak Kittipiyakul. Many-sources large deviations for max-weight scheduling. arXiv:0902.4569v1, 2009.

[11] V. J. Venkataramanan and Xiaojun Lin. Structural properties of LDP for queue-length based wireless scheduling algorithms. In *Proceedings of Allerton*, 2007.

$$\frac{1}{r}\log\mathbb{P}\big(L(Q(rT))\approx r\big) \approx -\inf_{a(\cdot):L(q(T))=1} \int_{t=0}^{T} \ell(\dot{a}(t))\,dt$$

The most likely path is:
send at mean rate λ for time $T-\mathcal{U}$;
then send at some overload rate λ^+ for time $\mathcal{U}$.
Choose $\mathcal{U} = \frac{1}{ALGP^+(\lambda^+)}$ ie enough time to reach $L(q(T)) = 1$.

The cost of the path is

$$= -\inf_{\lambda^+\in\mathbb{R}_+^N} \frac{\ell(\lambda^+)}{ALGP^+(\lambda^+)}.$$

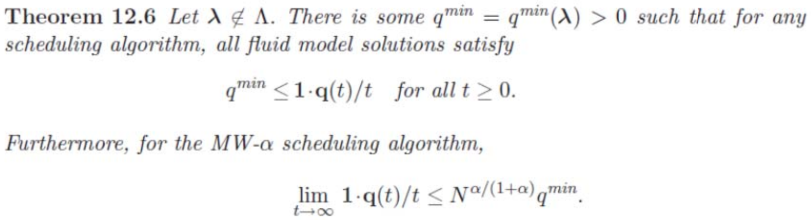
The proof uses $ALGP^+$ and its dual $RATE^+$.

DESIGNING OPTIMAL ALGORITHMS

Suppose we want to keep the total queue size $Q\cdot 1$ small.
We only know how about $L(Q)$; but happily

$$\frac{1}{N^{\alpha/1+\alpha}}\, Q\cdot 1 \le L(Q) \le Q\cdot 1$$

This bound translates into performance bounds.
For example, in overload,



Theorem 12.6 *Let $\lambda \notin \Lambda$. There is some $q^{min} = q^{min}(\lambda) > 0$ such that for any scheduling algorithm, all fluid model solutions satisfy*

$$q^{min} \le \mathbf{1}\cdot\mathbf{q}(t)/t \quad \textit{for all } t \ge 0.$$

Furthermore, for the MW-α scheduling algorithm,

$$\lim_{t\to\infty} \mathbf{1}\cdot\mathbf{q}(t)/t \le N^{\alpha/(1+\alpha)} q^{min}.$$

Hence the smaller α is, the closer we are to an optimal algorithm.

FUTURE WORK

- Multiscale phenomena

  For some scheduling algorithms, e.g. choose $\pi(t)$ to be

  $$\underset{p \in \mathcal{S}}{\operatorname{arg\,max}} \sum p_n \log(1+Q_n)$$

  it is hard to identify the fluid model. The fluid model seems to depend on higher-order statistics of the arrival process.

- Distributed scheduling algorithms

  What distributed algorithms can we analyse/design, e.g. for wireless ad-hoc networks?