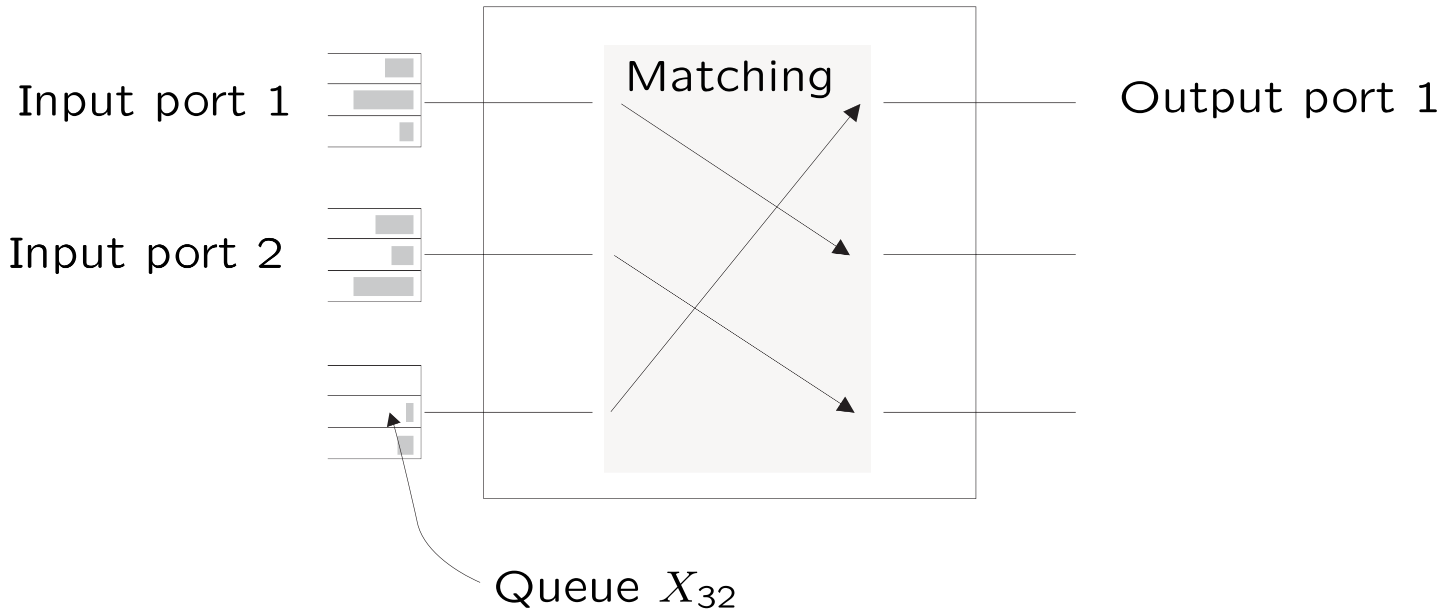
Input-Queued Switches in Heavy Traffic

Damon Wischik
Statistical Laboratory, Cambridge
Electrical Engineering, Stanford
`http://www.wischik.com/damon`

+J.M.Harrison, F.P.Kelly, S.Kumar,
B.Prabhakar, D.Shah, R.Williams.

An $n \times n$ input-queued switch



- Packets arrive at input port i destined for output port j as a Poisson process of rate λ_{ij}. They are stored in queue X_{ij}.
- Every time step, the switch chooses a *matching* of inputs to outputs, and tries to serve one packet from each of the n queues involved in the matching.
- If it offers service to an empty queue, we say it *fires a blank*.

Notation

X_{11}	X_{12}	$\cdot$	$W_{1\cdot}$
X_{21}	$\cdot$	$\cdot$	$\cdot$
$\cdot$	$\cdot$	$\cdot$	$\cdot$
$W_{\cdot 1}$	$\cdot$	$\cdot$	$W_{\cdot\cdot}$

where $W_{1\cdot} = \sum_j X_{1j}$ etc.

A matching corresponds to a permutation matrix (or service matrix): say

$$\pi_{ij} = \begin{cases} 1 & \text{if input } i \text{ matched to output } j \\ 0 & \text{else} \end{cases}$$

Maximum-weight matching algorithm

Let the *weight* of matching π be

$$\pi \cdot X = \sum_{i,j} \pi_{ij} X_{ij}.$$

Let the *maximum-weight* be

$$m = \max_{\pi} \pi \cdot X.$$

The *maximum-weight matching algorithm* MWM chooses, at each time step, some service matrix of weight m.

Theorem. If the matrix of arrival rates $\lambda = (\lambda_{ij})$ is doubly substochastic, then the switch is stable.

(McKeown+Anantharam+Walrand 1996 for independent arrivals, Dai+Prabhakar 2000 for general arrivals).

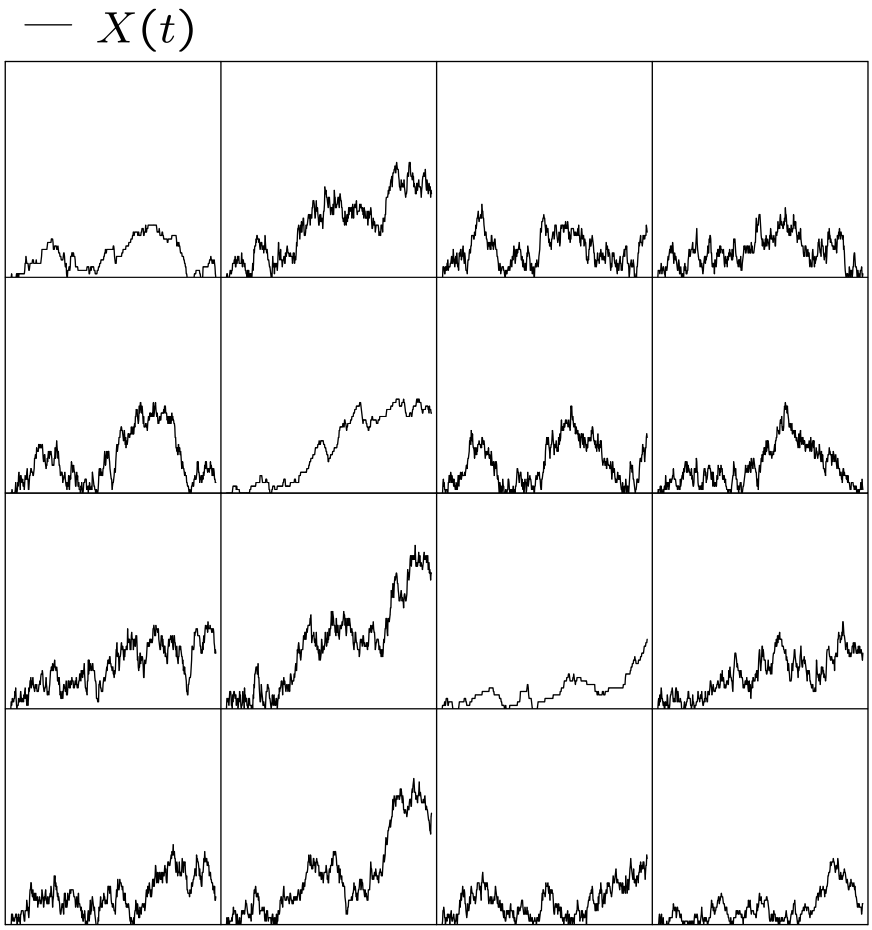
The fact of state space collapse

1. Simulate a switch running MWM. Record the queue process $X(t)$.
2. Calculate the row and column workloads $W(t) = (W_{1\cdot}(t), \dots, W_{\cdot 1}, \dots)$.
3. Define $\tilde{X}(t) = \Delta W(t)$, for a function Δ (the *lifting map*) defined below.
4. Observe: $\tilde{X}(t) \approx X(t)$.

Definition. The lifting map $\Delta(w)$ gives a solution x to the linear program:

$$
\begin{aligned}
&\min \max_{\pi} \pi \cdot x \\
&\text{subject to} \quad \begin{cases} \sum_j x_{ij} = w_{i\cdot} \\ \sum_i x_{ij} = w_{\cdot j} \\ *\text{if } \lambda_{ij} = 0 \text{ then } x_{ij} = 0 \end{cases} \\
&\text{over} \quad x \geq 0.
\end{aligned}
$$

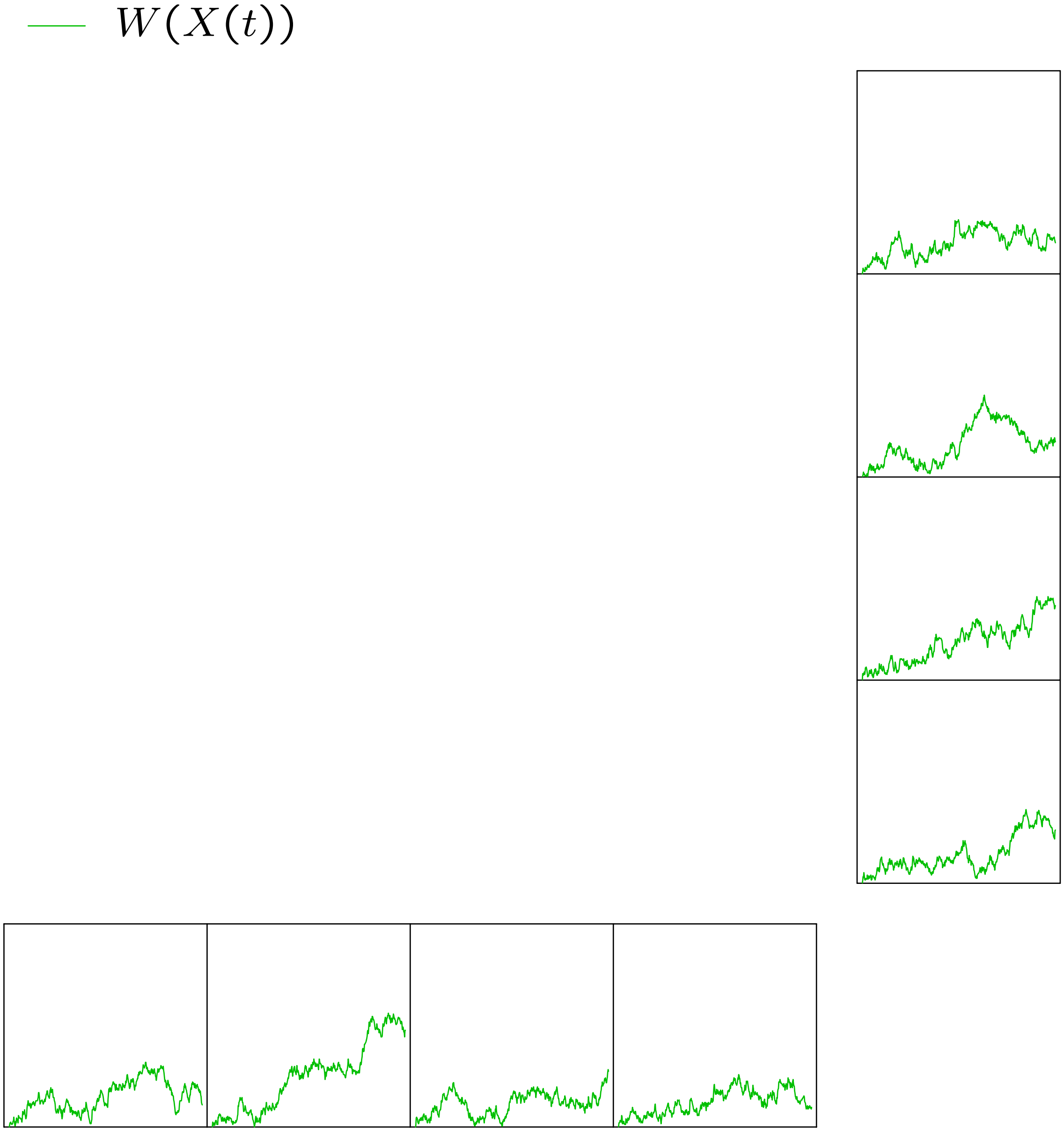
Traces



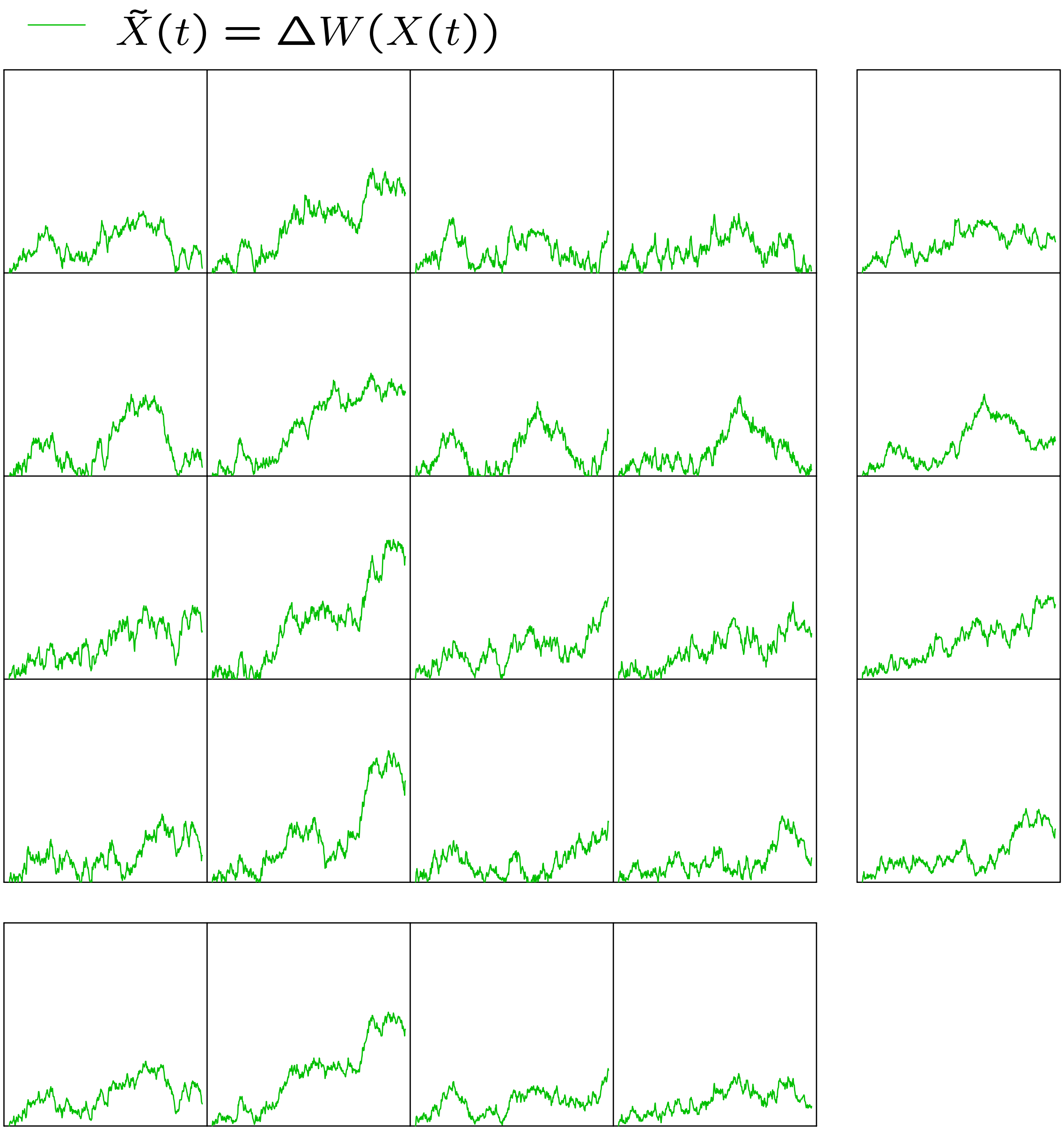
$\lambda =$

.022	.310	.309	.354	*.995*
.310	.021	.386	.278	*.995*
.309	.386	.012	.288	*.995*
.354	.278	.288	.076	*.995*
.995	*.995*	*.995*	*.995*	

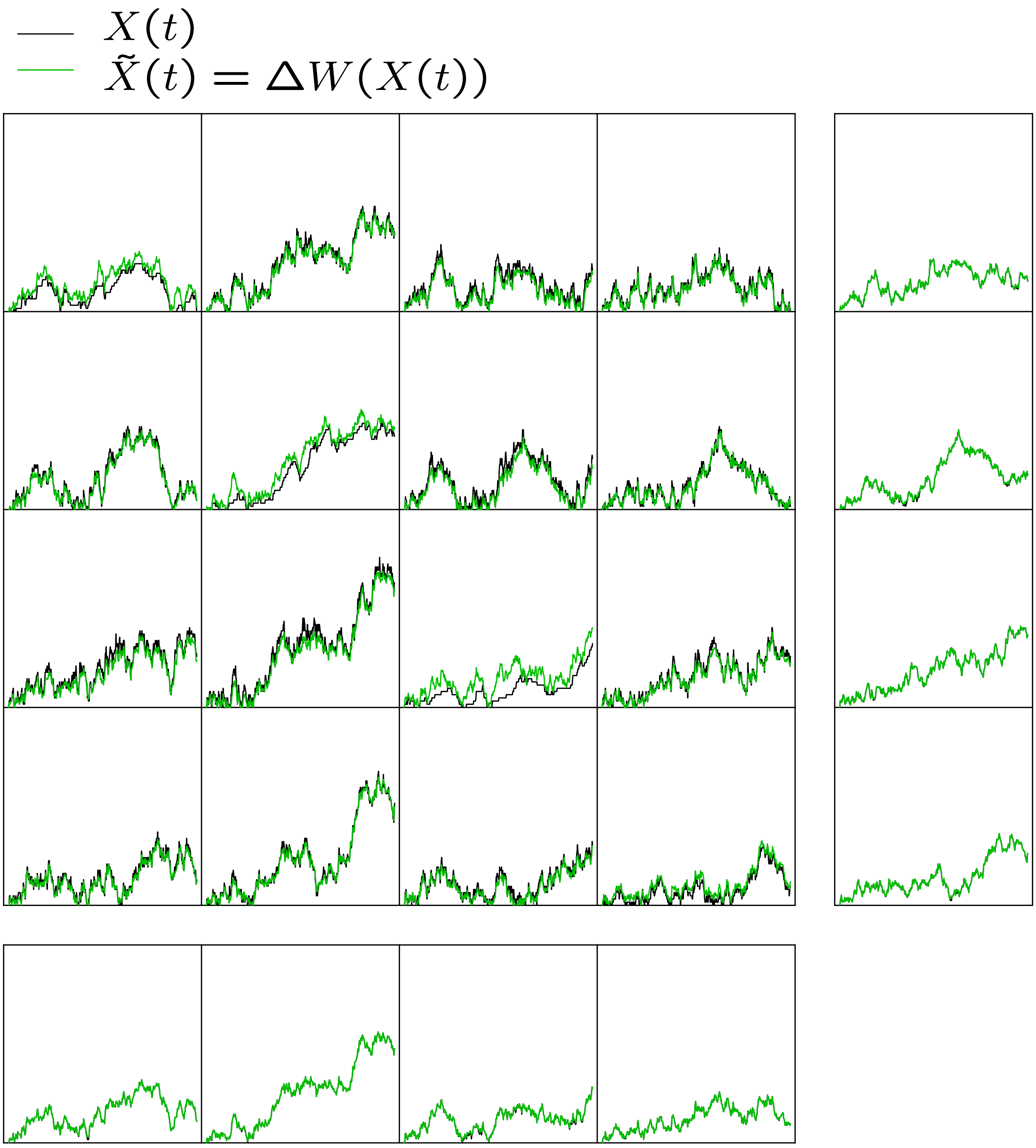
Traces



Traces

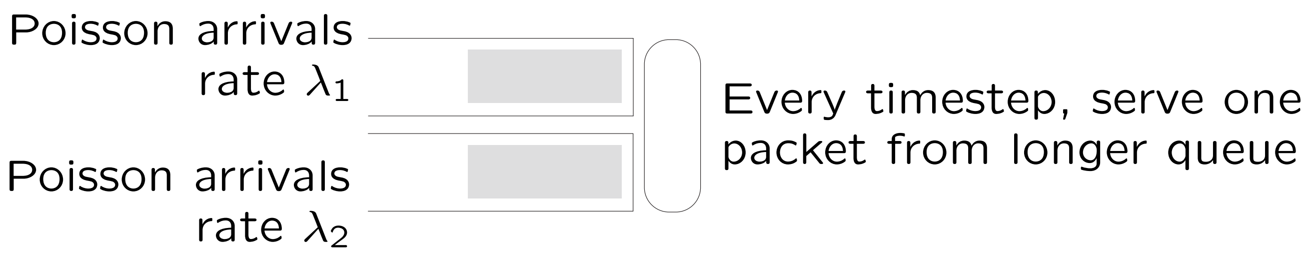


Traces



Why does SSC happen?

Consider a simpler model:



Let $\lambda = \lambda_1 + \lambda_2$. Let X_1 and X_2 be the two queue sizes, and $W = X_1 + X_2$.

Timescale separation

How do W and X_i evolve, over timescales $L\delta$ and $L^2\delta$? (L large, δ small.)

Suppose the system is in heavy traffic: $\lambda = 1 - \frac{1}{L}C$.

Over timescale $L^2\delta$:

- Arrivals $\sim$ Poisson$(\lambda L^2\delta) \approx \lambda L^2\delta + L\,\mathsf{N}(0, \sigma^2\delta)$.
- Service $L^2\delta$.
- Net change in W is $L^2\delta(\lambda - 1) + L\,\mathsf{N}(0, \sigma^2\delta)$, i.e. $L\Big(-\delta C + \mathsf{N}(0, \sigma^2\delta)\Big)$.
- W/L behaves like reflected Brownian Motion, drift $-C$.

The relevant timescales and spacescales are:

- How much does W/L change by over time $L^2\delta$? — By $-\delta C + \mathsf{N}(0, \sigma^2\delta)$.
- How much does W/L change by over time $L\delta$? — By $O(1/\sqrt{L})$.
- How much does X_1/L change by over time $L\delta$? — By $\delta(\lambda_1 - C_1) + O(1/\sqrt{L})$, where C_i is the fraction of service effort devoted to server i: $C_1 + C_2 = 1$.

Summary of SSC

Suppose $\lambda = 1 - \frac{1}{L}C$. Then:

- over timescale L^2, the scaled aggregate workload W/L evolves like a reflected Brownian motion;
- over timescale L, the balanced fluid model tells us about the disposition of workload over the two queues.
  - here, the balanced fluid model is
    $\dot{x}_i = \lambda_i - c_i$,
    $c_1 + c_2 = 1$, $c_i = 0$ if x_i is not the largest
  - equilibrium states are those where $x_i = x_j$.

So, over timescale L, the system will head to an *invariant state* $X_i = X_j$, while W will hardly change.

- The state space has collapsed from two dimensions (X_1, X_2) to one dimension W.
- The lifting map $\Delta(W) = (\frac{1}{2}W, \frac{1}{2}W)$ maps from the workload to the actual state.

Fluid model of MWM

The fluid model for MWM is (Prabhakar+Dai 2000)

$$\dot{x}_{ij} = \begin{cases} \lambda_{ij} - \sigma_{ij} & \text{if } x_{ij} > 0 \\ (\lambda_{ij} - \sigma_{ij})^{+} & \text{if } x_{ij} = 0 \end{cases}$$

$$\sigma \in \Big\langle \text{maximum-weight matchings} \Big\rangle.$$

Theorem. Let $m(t) = \max_\pi \pi \cdot x(t)$. Then there exists $\varepsilon > 0$ (depending only on λ) such that

- either $\dot{m}(t) < -\varepsilon$,
- or $\dot{x}(t) = 0$ and $x(t)$ is the unique solution to the linear program $x(t) = \Delta w(x(t))$.

The linear program $\Delta(w)$ is:

$$\min \max_\pi \pi \cdot x \quad \text{over } x \geq 0$$

$$\text{subject to} \quad \begin{cases} \text{if } \sum_j \lambda_{ij} = 1 \text{ then } \sum_j x_{ij} = w_{i\cdot} \\ \text{if } \sum_i \lambda_{ij} = 1 \text{ then } \sum_i x_{ij} = w_{\cdot j} \\ \text{if } \lambda_{ij} = 0 \text{ then } x_{ij} = 0 \end{cases}$$

Theorem. A point x is invariant if and only if $x = \Delta w(x)$.

Consequences of SSC

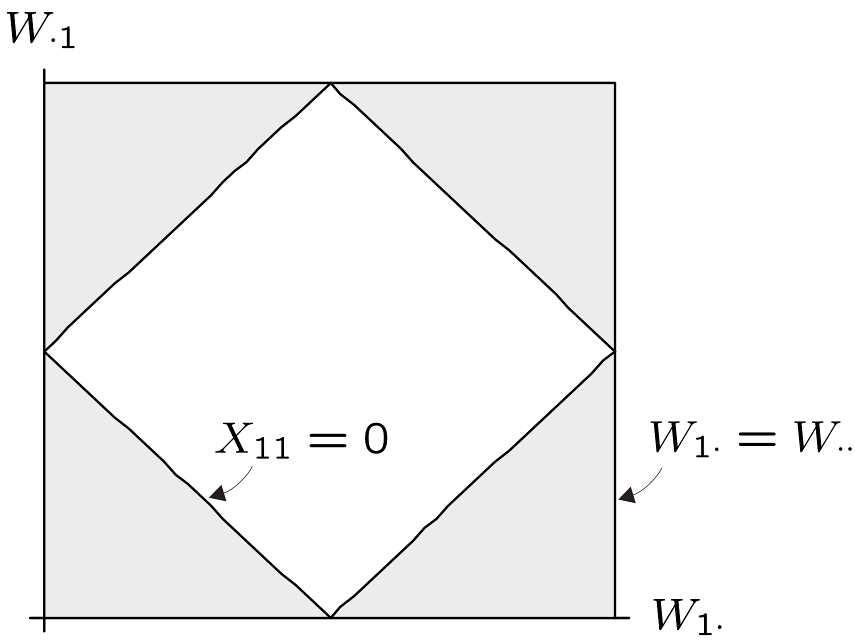
Consider a 2×2 switch running MWM. We only need keep track of the workloads $W = (W_{1\cdot}, W_{\cdot 1}, W_{\cdot\cdot})$: from them we can infer the X_{ij}.

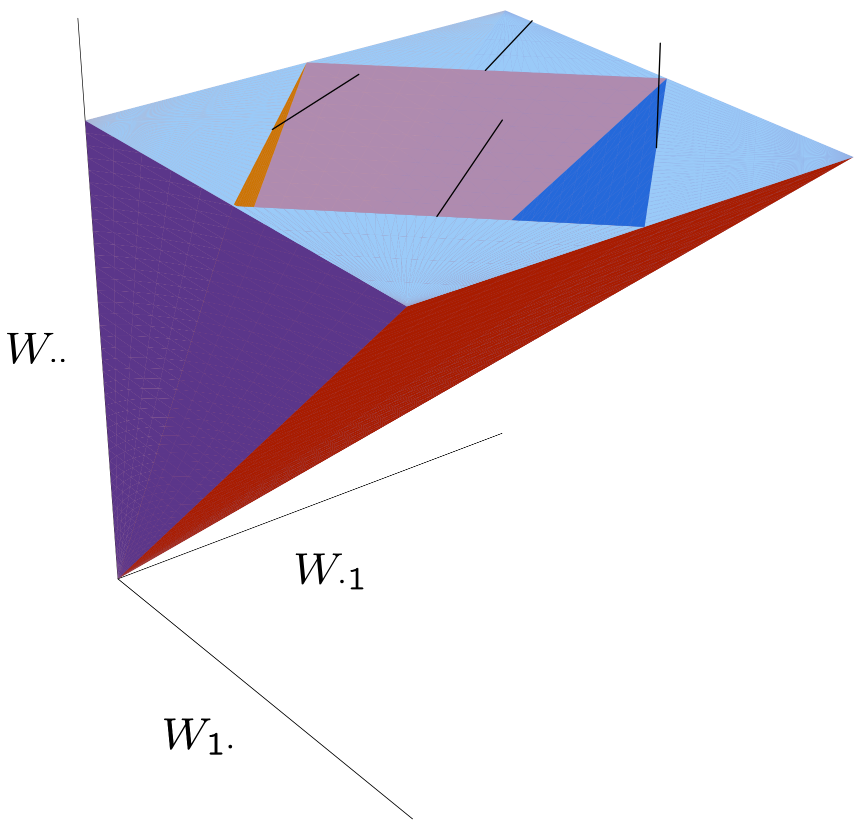
$\frac{1}{2}(W_{1\cdot} + W_{\cdot 1})$ $-\frac{1}{4}W_{\cdot\cdot}$	$\cdot$	$W_{1\cdot}$
$\cdot$	$\cdot$	
$W_{\cdot 1}$		$W_{\cdot\cdot}$

We can calculate the set of invariant states, and the corresponding workloads. The space $\mathcal{W}$ of allowed workloads is bounded by the four planes $X_{ij}(W) = 0$.

The workload process $W(t)$ evolves in $\mathcal{W}$ like a Brownian motion. At the boundaries of $\mathcal{W}$, it may be *reflected* to keep it in the space. A reflection on plane $X_{ij}(W) = 0$ corresponds to firing blanks on queue X_{ij}.

Feasible workload space





Different weight functions

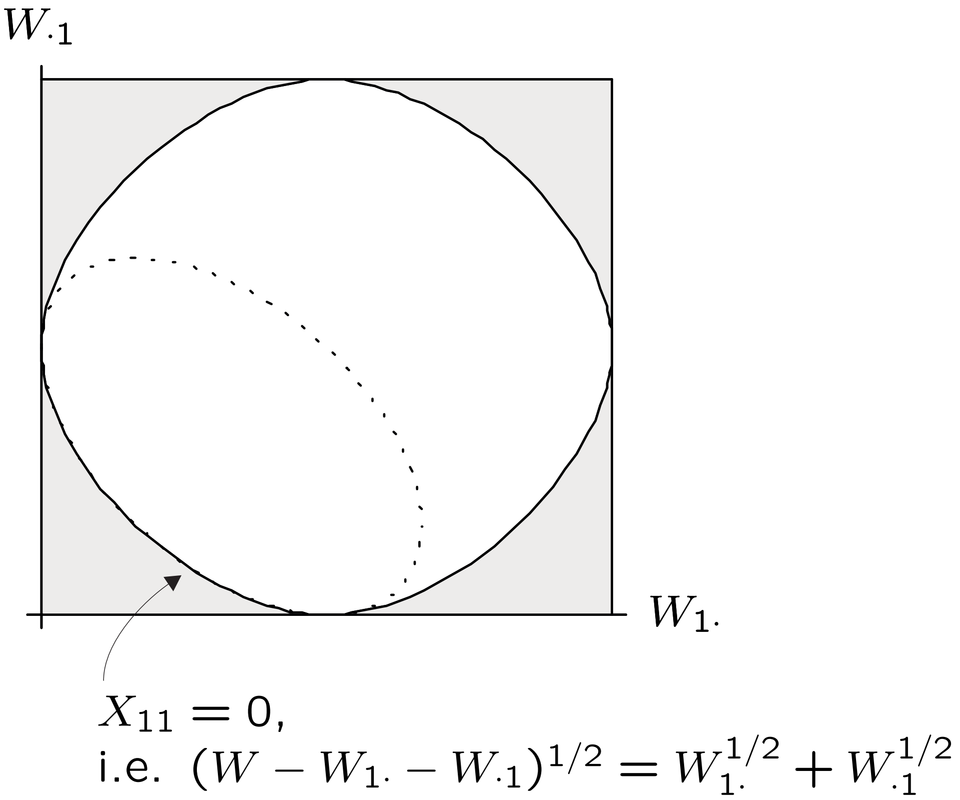
Let the weight of matching π be $\sum_{i,j} \pi_{ij} f(X_{ij})$, with $f(x) = x^\alpha$.

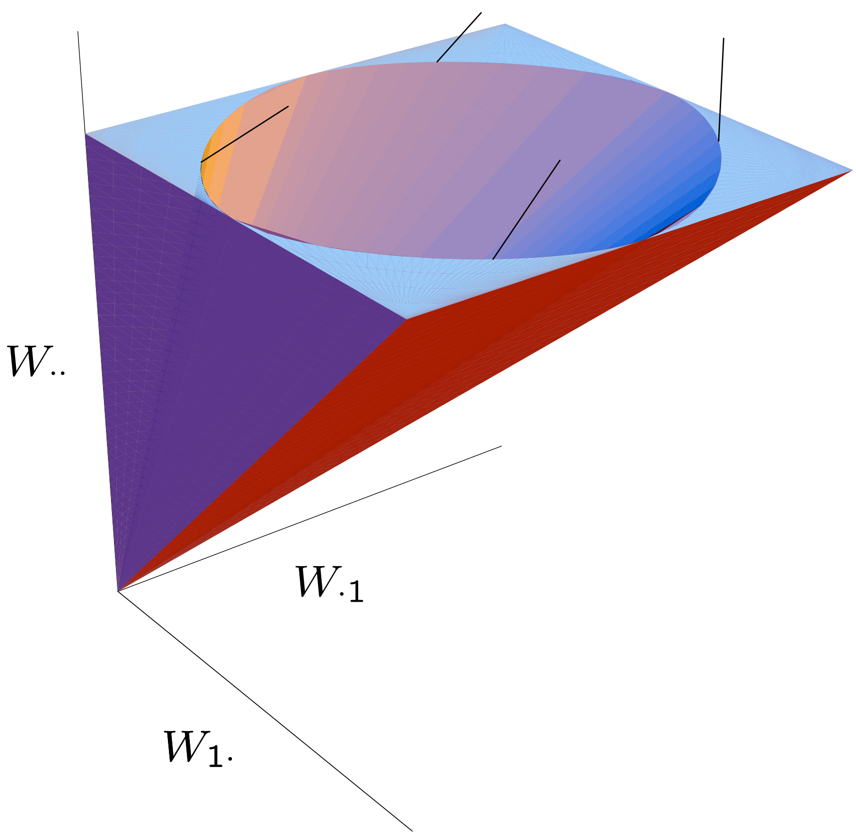
Again, we can find the space of allowed workloads $\mathcal{W}$, and the lifting map $X = \Delta(W)$.

It turns out that $\mathcal{W}$ gets smaller as α increases. When W hits the boundary of $\mathcal{W}$, blanks are fired; the smaller $\mathcal{W}$, the more blanks. Thus the performance of MWM is better for small α.

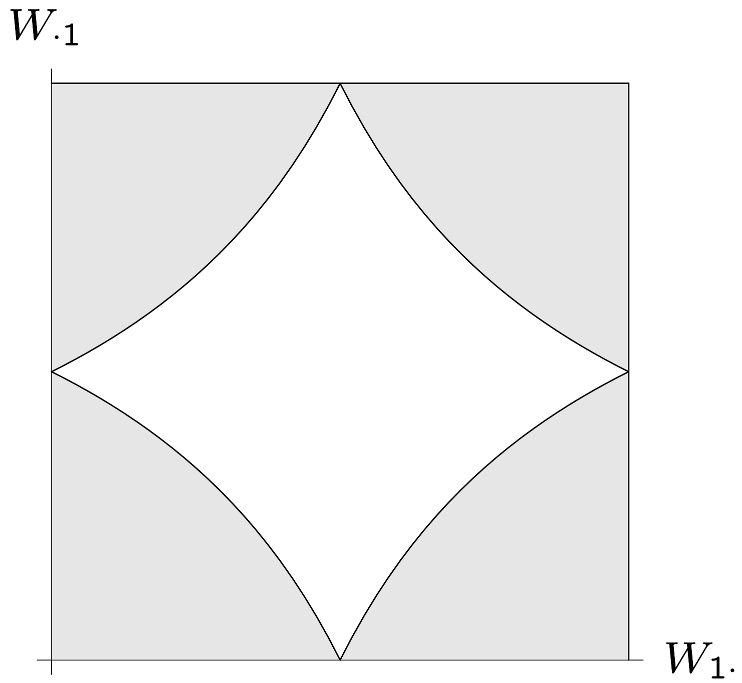
Conjecture. An optimal matching algorithm is MWM with $\alpha \to 0$. That is, look at all maximum-size matchings, and choose the one with the largest weight, using weight function $f(x) = \log x$.

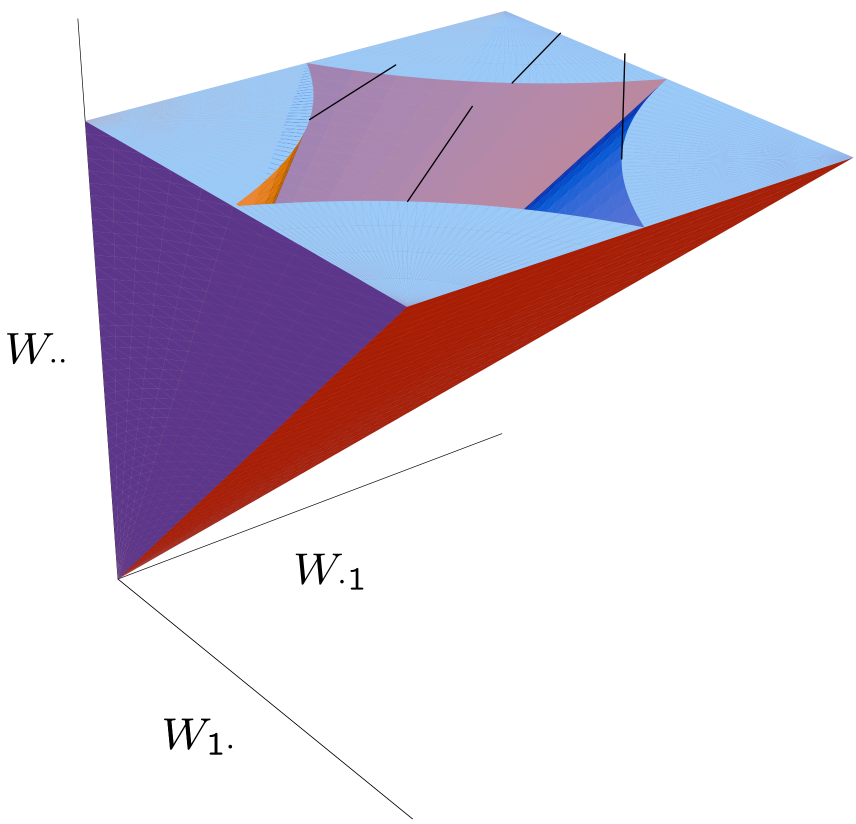
Feasible workload space, $f(x) = x^{1/2}$





Feasible workload space, $f(x) = x^2$





Calculating the probability of overflow

Suppose the line card for input port 1 has buffer B, i.e. loss will occur if packets arrive on input port 1 when $W_{1\cdot} = B$.

We have seen that the workload process W evolves like a reflected Brownian motion. We know the drift, the state space, and the angles of reflection.

We would like to calculate $\mathbb{P}(W_{1\cdot} \geq B)$.

- Perhaps amenable to numerical estimation, if B small.
- Perhaps amenable to calculation using large deviations techniques, if B large.