Decidability issues in the theory of queueing networks

David Gamarnik
MIT

33rd CONFERENCE ON THE MATHEMATICS OF OPERATIONS RESEARCH

January 2008

Joint work with
Dmitriy Katz, MIT

Talk Outline

- Stochastic queueing networks and stability

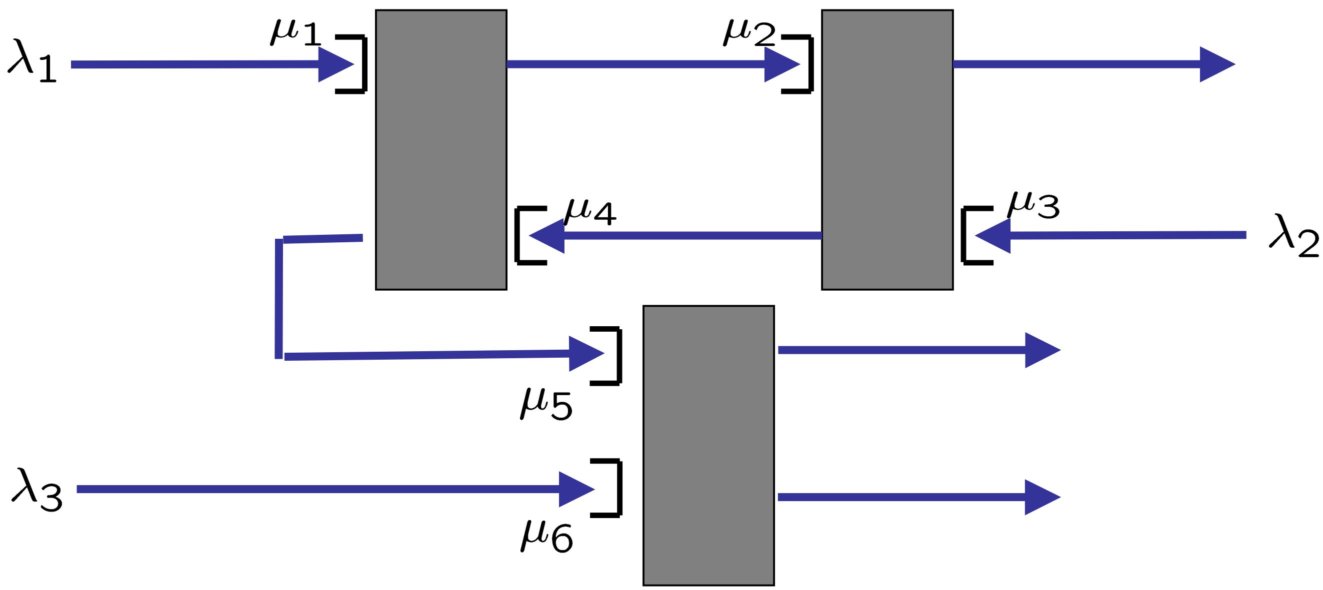
- Our model: *Deterministic* multiclass queueing networks.

- Detour into undecidability. Counter machines and halting problem.

- **Main result:**
  Stability of multiclass queueing networks with fininte and infinite buffers under static buffer priority policies is undecidable.

- Further work.

Stochastic Multiclass Queueing Network



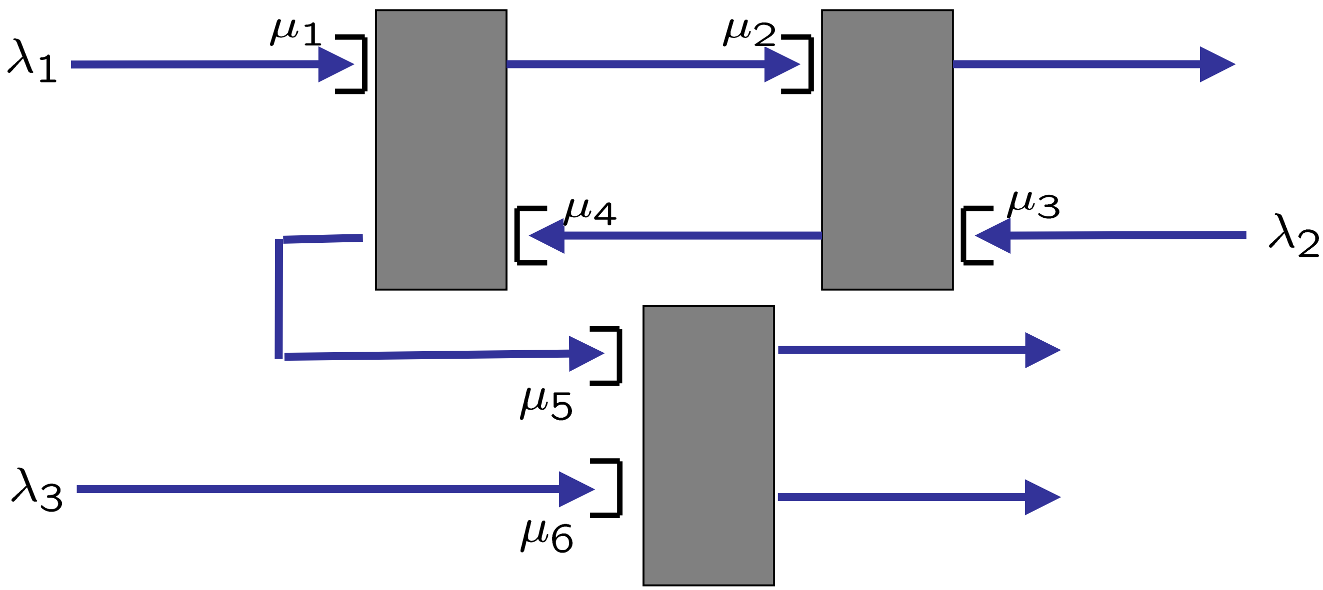
i.i.d. interarrival time with rates λ_i

i.i.d. service times with rates μ_j

Scheduling policy: FIFO, buffer priority, etc.

Stability: positive Harris recurrence. Informally, $\sup_{t \geq 0} \mathbb{E}[\|Q(t)\|] < \infty$

Stochastic Multiclass Queueing Network



Necessary condition for stability: for every server σ:

$$\rho_\sigma \triangleq \sum_{i \in \sigma} \frac{\lambda_i}{\mu_i} < 1$$

Stochastic Multiclass Queueing Network

- Kumar & Seidman [1990], Lu & Kumar, Rybko & Stolyar [1991] Introduced MQNET. First instability results.

- Bramson, Seidman [1994]. FIFO can be unstable.

- Dai, Stolyar [1995] Stability of a fluid model implies stability of MQNET.

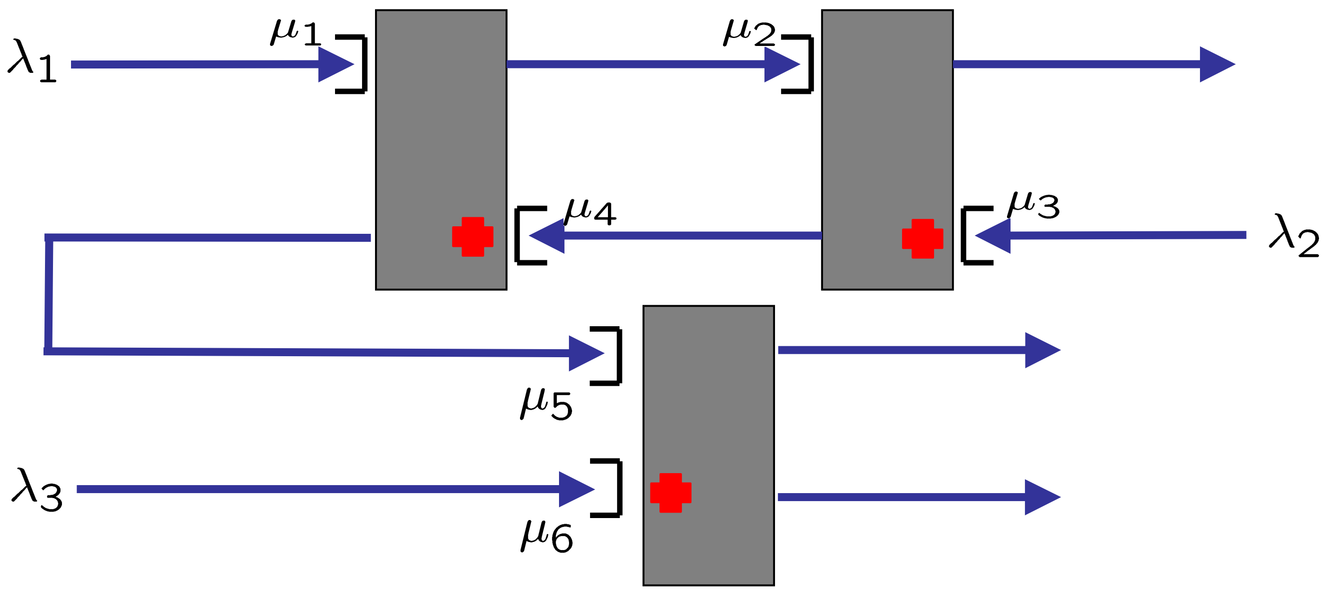
- Bertsimas, G. & Tsitsiklis [1996], Classification of globally stable fluid 2-server queueing networks.

- Dai & Vande Vate [2000], G. & Hasenbein. Classification of globally stable stochastic 2-server queueing networks.

- Many more results …

No constructive method for determining stability of a given MQNET/scheduling policy is known

Our model: *deterministic* Multiclass Queueing Network



Deterministic interarrival time with rates λ_i

Deterministic service times with rates μ_j

Scheduling policy: buffer priority

$\boxed{\text{Additional feature: infinite/finite buffers}}$

Stability: $\sup_{t \geq 0} \|Q(t)\| < \infty$

Main Result

Theorem. The following decision problem

"Given QNET topology/parameters/buffer priority rule $(\lambda, \mu, P, C, \theta)$
determine whether the network is stable"

is **undecidable.**

Proof: reduction from the Counter Machine Halting Problem

Earlier work:

G. [2002]. Stability under *generalized priority* policy is undecidable.

G. [2006]. Computing stationary distribution and large deviations rates is undecidable.

Detour into undecidability

- Alan Turing [1930's]. **Turing Machine** and the **Turing Halting Problem**

- Classical undecidable problems:

  Post Correspondence Problem,
  Probabilistic Automata

- **Godel's Incompleteness Theorem**

Undecidable problems in control theory

- Blondel, Bournez, Papadimitriu, Paterson, Tsitsiklis.

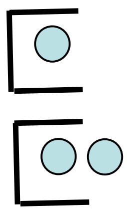
- Such as :
  Joint spectral radius
  Matrix mortality
  Piece-wise affine control (uses **Counter machines***)*

Counter Machine

States 

s_1 s_2 s_3

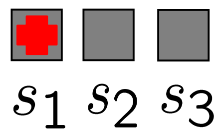
Counter 1.
Counter 2.



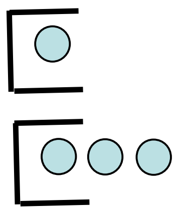
Update Table (example with m=3)

Γ	$(0,0)$	$(0,+)$	$(+,0)$	$(+,+)$
s_1	$(s_3, 0, +1)$	$(s_1, 0, +1)$	$(s_3, -1, 0)$	$(s_1, -1, 0)$
s_2	$(s_3, 0, +1)$	$(s_3, 0, -1)$	$(s_2, 0, +1)$	$(s_1, 0, +1)$
s_3	$(s_2, +1, 0)$	$(s_2, +1, 0)$	$(s_2, -1, 0)$	$(s_2, 0, +1)$

Counter Machine

States 

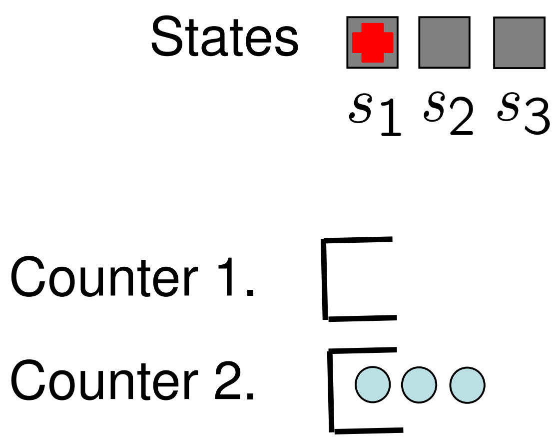
Counter 1.

Counter 2. 

Update Table (example with m=3)

Γ	$(0,0)$	$(0,+)$	$(+,0)$	$(+,+)$
s_1	$(s_3,0,+1)$	$(s_1,0,+1)$	$(s_3,-1,0)$	$(s_1,-1,0)$
s_2	$(s_3,0,+1)$	$(s_3,0,-1)$	$(s_2,0,+1)$	$(s_1,0,+1)$
s_3	$(s_2,+1,0)$	$(s_2,+1,0)$	$(s_2,-1,0)$	$(s_2,0,+1)$

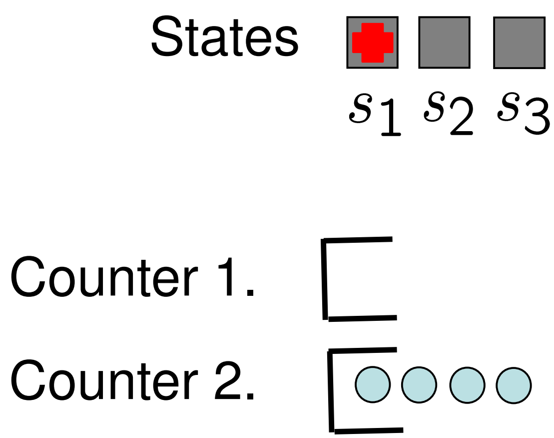
Counter Machine



Update Table (example with 3 states)

Γ	$(0,0)$	$(0,+)$	$(+,0)$	$(+,+)$
s_1	$(s_3, 0, +1)$	$(s_1, 0, +1)$	$(s_3, -1, 0)$	$(s_1, -1, 0)$
s_2	$(s_3, 0, +1)$	$(s_3, 0, -1)$	$(s_2, 0, +1)$	$(s_1, 0, +1)$
s_3	$(s_2, +1, 0)$	$(s_2, +1, 0)$	$(s_2, -1, 0)$	$(s_2, 0, +1)$

Counter Machine



Update Table (example with m=3)

Γ	$(0,0)$	$(0,+)$	$(+,0)$	$(+,+)$
s_1	$(s_3,0,+1)$	$(s_1,0,+1)$	$(s_3,-1,0)$	$(s_1,-1,0)$
s_2	$(s_3,0,+1)$	$(s_3,0,-1)$	$(s_2,0,+1)$	$(s_1,0,+1)$
s_3	$(s_2,+1,0)$	$(s_2,+1,0)$	$(s_2,-1,0)$	$(s_2,0,+1)$

Counter Machine Halting Problem

Theorem. The following decision problem

"Given a starting configuration $(s_1, 0, 0)$ *determine whether it is eventually repeated"*

is **undecidable.**

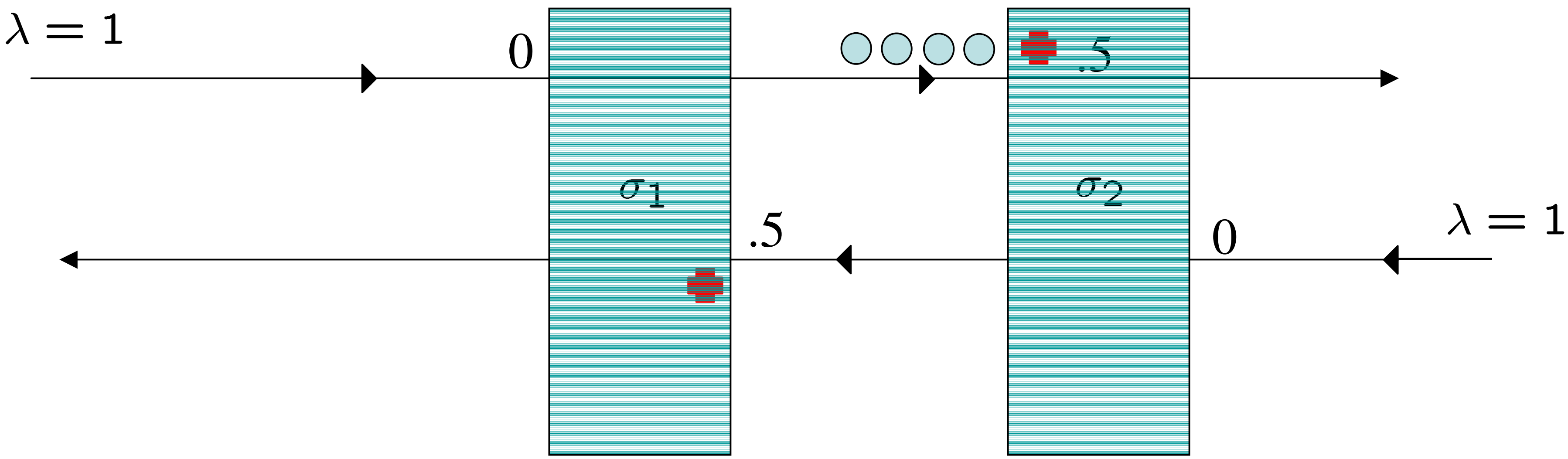
Corollary. "Stability" of a counter machine is undecidable.

Proof: reduction from the **Turing Halting Problem**.

Note: 1-counter machine is "decidable".

Simplified counter machine: change in counters is a function of the state only. Proof – simple reduction from a general counter machine.

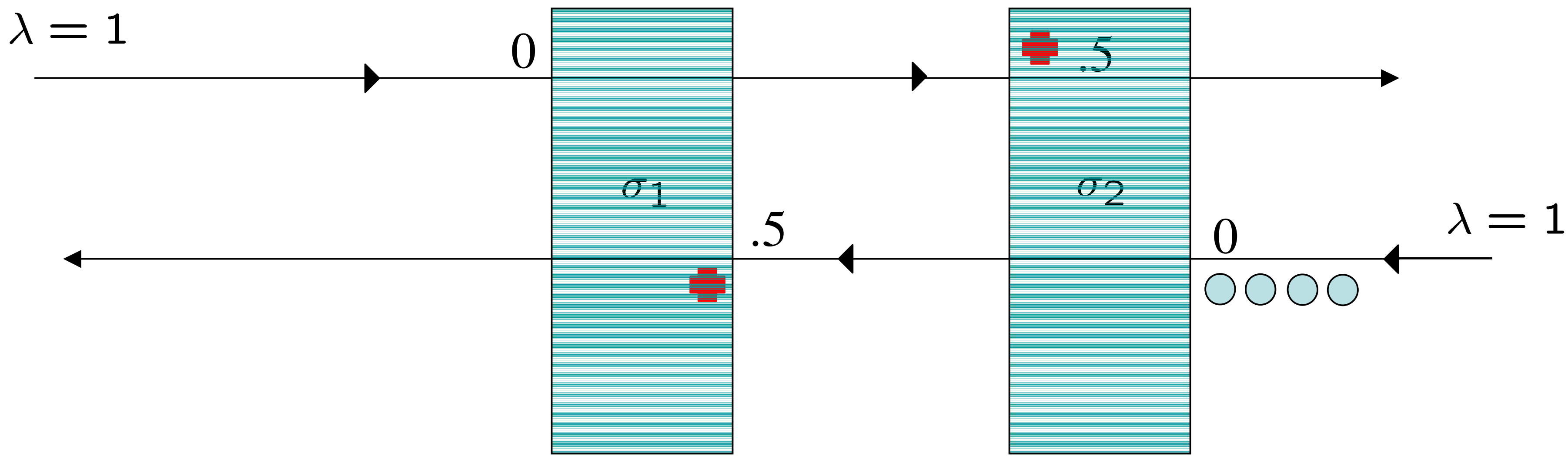
Critical Rybko-Stolyar Network



$$\rho_{\sigma_1} = \rho_{\sigma_2} = 1/2$$

$$Q(0) = (0, n, 0, 0)$$

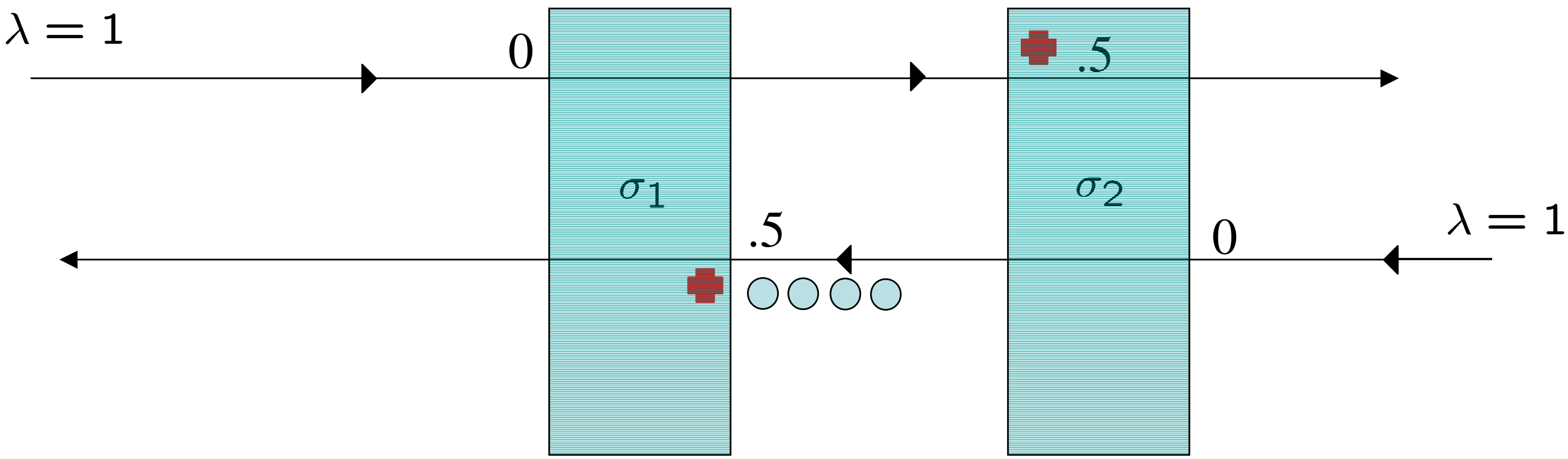
Critical Rybko-Stolyar Network



$$\rho_{\sigma_1} = \rho_{\sigma_2} = 1/2$$

$$Q(n-) = (0, 0, n, 0)$$

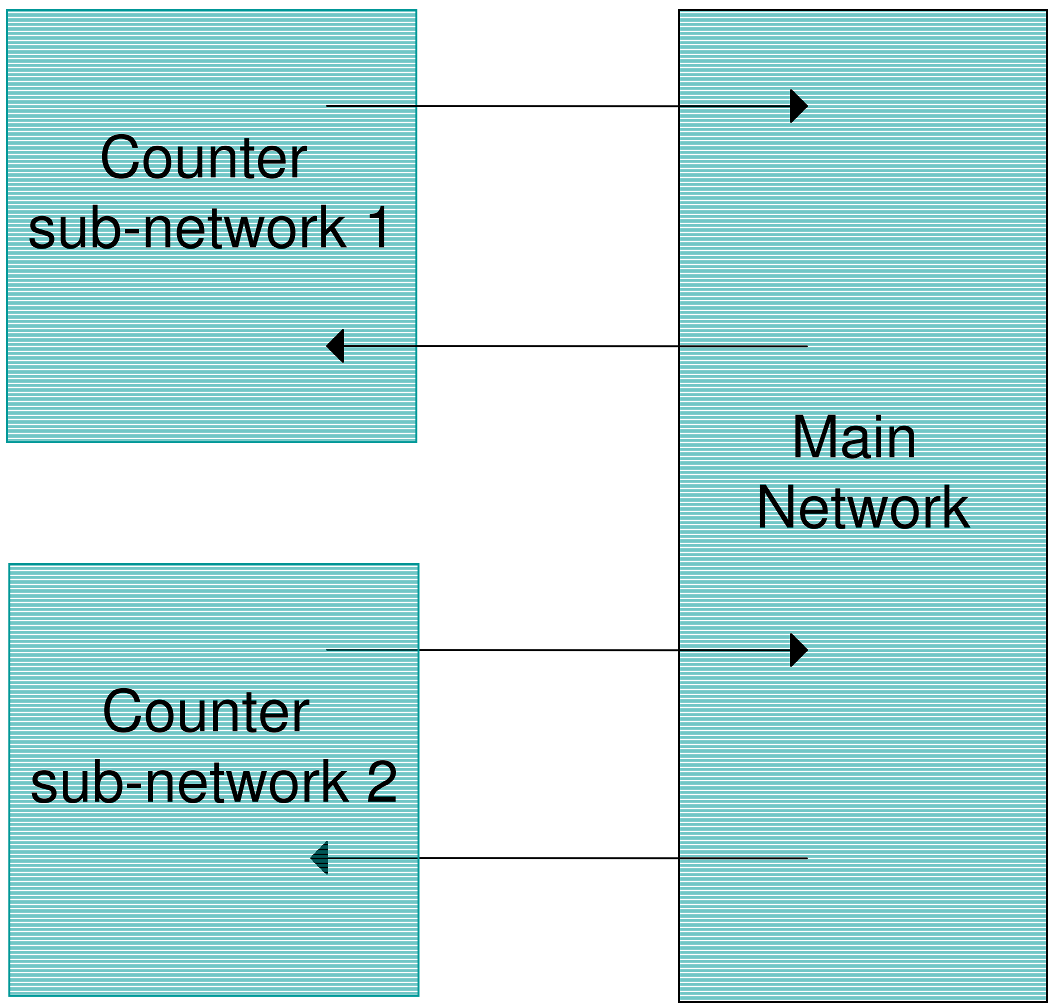
Critical Rybko-Stolyar Network



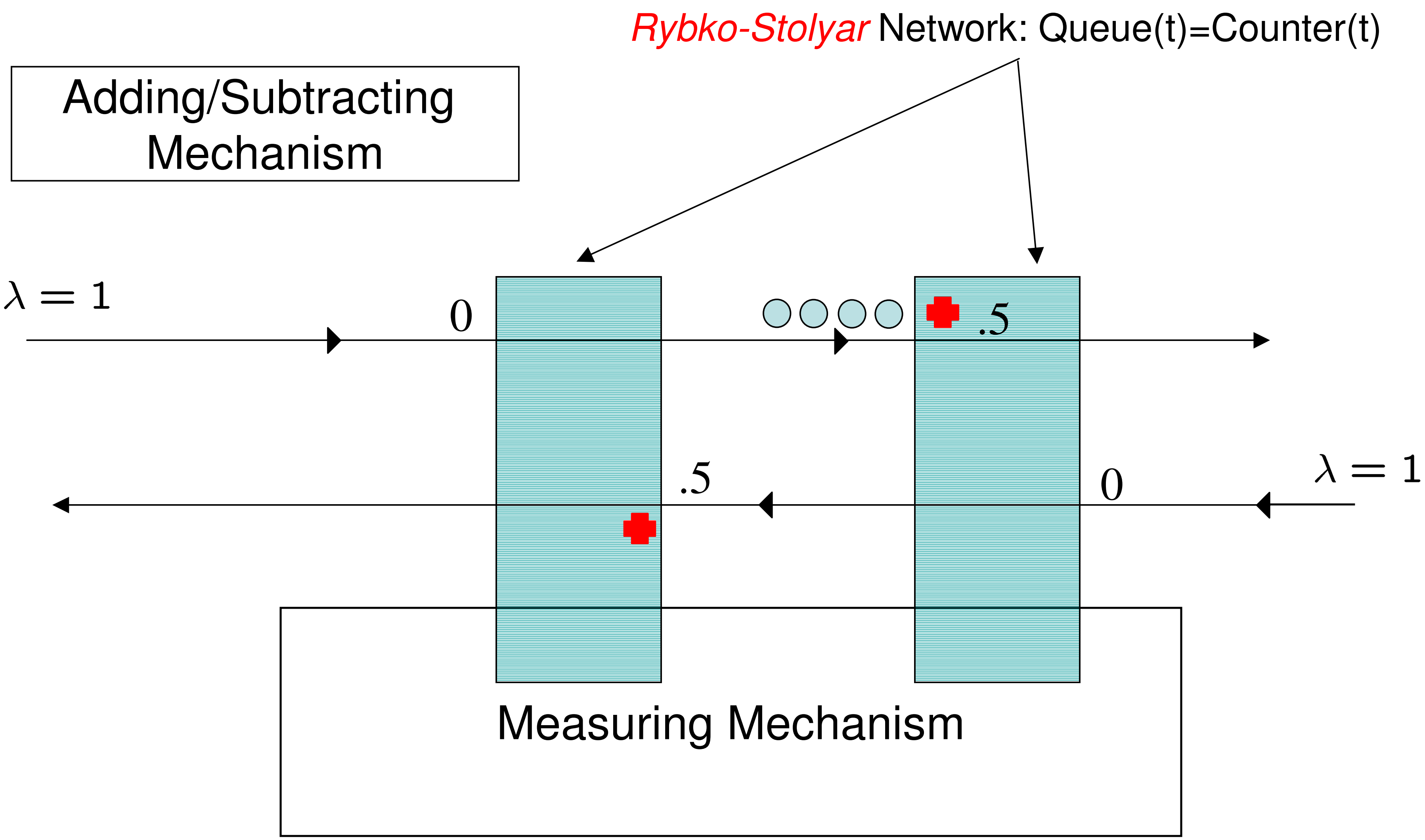
$$\rho_{\sigma_1} = \rho_{\sigma_2} = 1/2$$

$$Q(n+) = (0, 0, 0, n)$$

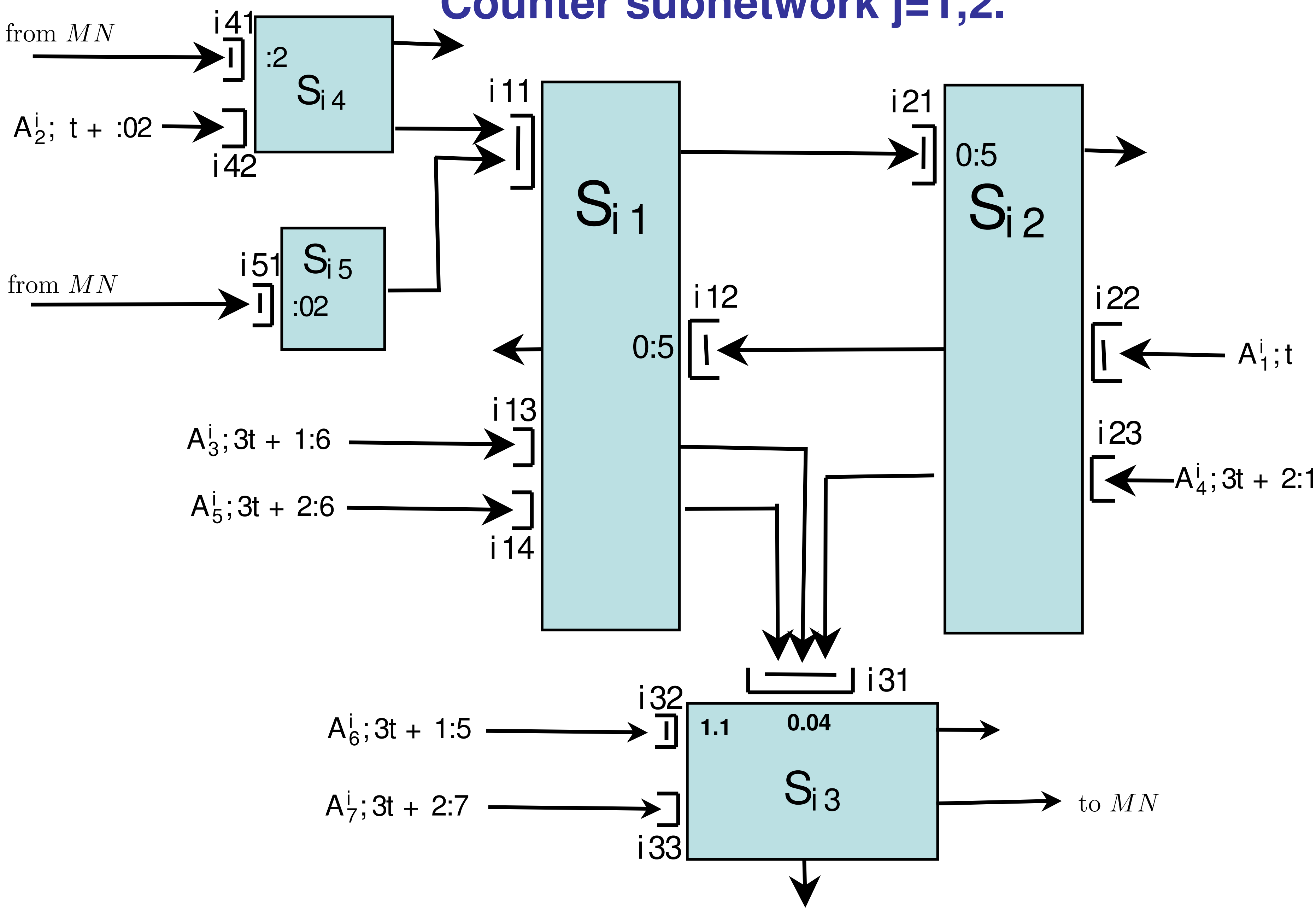
Counter Machine to Queueing Network. Reduction



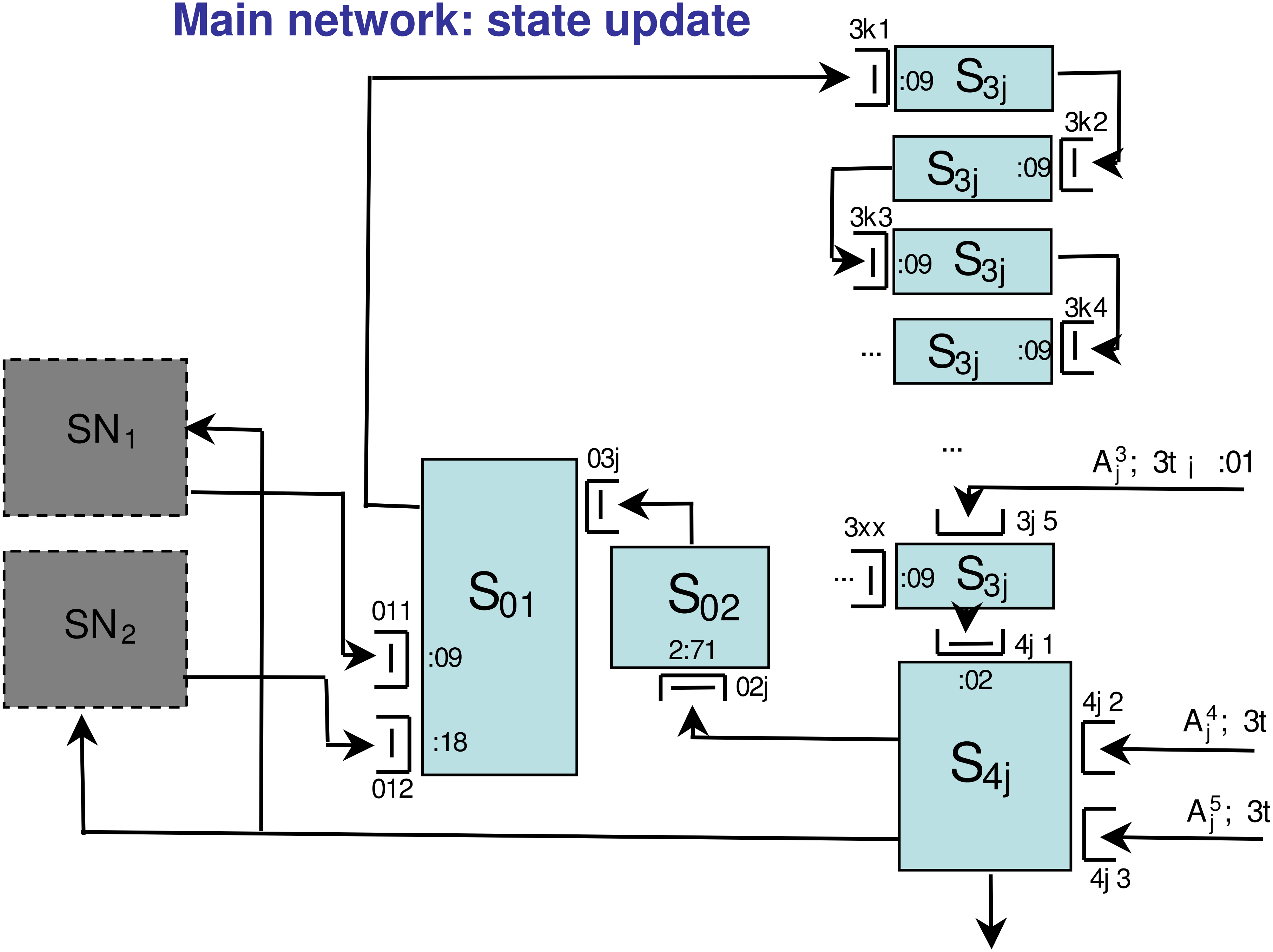
Counter subnetwork j=1,2.



Counter subnetwork j=1,2.



Main network: state update



Extension: Skorohod mapping problem and stability

$$Z(t) = X(t) + RY(t),$$
$$\int Z_i(t) dY_i(t) = 0, \quad i = 1, 2, \ldots, J$$
$$Z, Y \geq 0, Y(0) = 0,$$

Theorem. Williams et al. [1994]. Solution to the Skorohod problem exists iff R is completely-S.

Theorem. Deciding whether a given Skorohod problem with *a given initial state* has a stable solution is *undecidable*.

Proof: a counter machine with m states can be embedded into a Skorohod mapping problem with $J = 5m + 9$ in such a way that one step of the counter machine corresponds to 5 steps of the Skorohod mapping.

Skorohod mapping problem and stability

$$
\begin{aligned}
Z(t) &= X(t) + RY(t), \\
\int Z_i(t)dY_i(t) &= 0, \quad i = 1, 2, \ldots, J \\
Z, Y \geq 0, Y(0) &= 0,
\end{aligned}
$$

Details: **R** has positive upper triangular part including the diagonal. As a result it is completely-S.

Open: stability for arbitrary initial state.

Summary and future work

• Stability of queueing networks under priority type scheduling policies with finite/infinite buffers is undecidable.

• Proof: reduction from some other “known” undecidable problem (counter machine)

• **Challenge:** FIFO policy, infinite buffers, non-zero service times