A Study of the Coexistence of Heterogeneous Flows in Data Network

Adrian Sai-wah Tam

swtam3@ie.cuhk.edu.hk

Department of Information Engineering

The Chinese University of Hong Kong

Traffic Coexistence

Elastic vs **Inelastic**

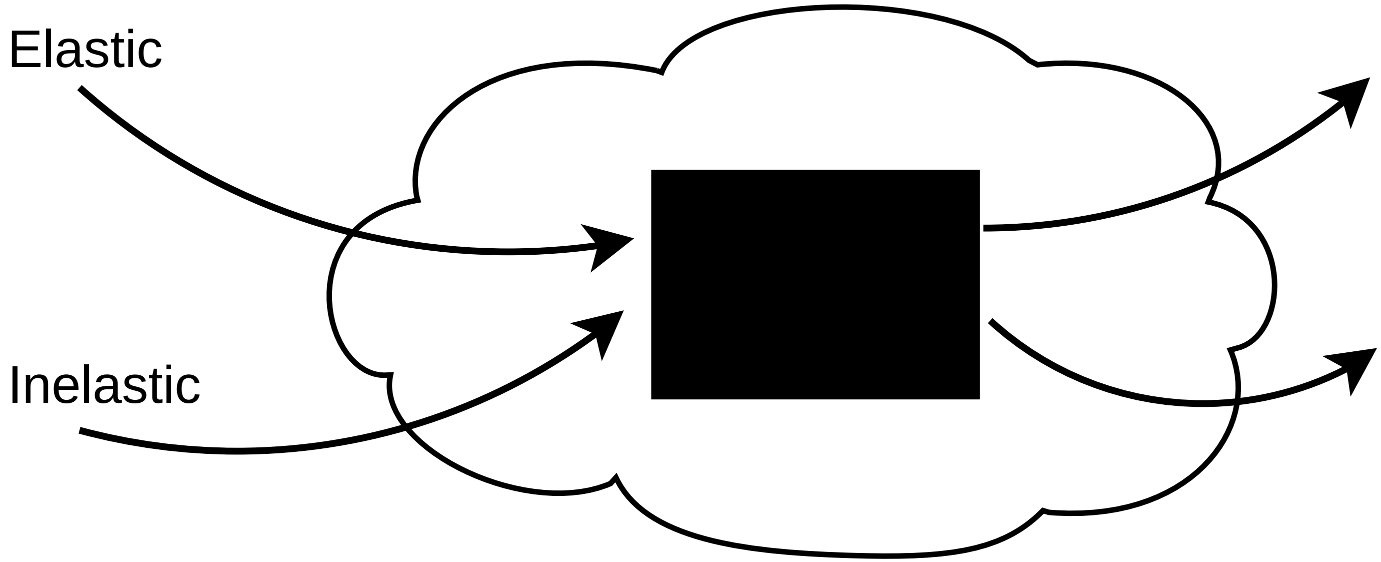
Elastic

- No explicit constrain on transfer
- Example: File transfer
- usually delivered by TCP

Inelastic

- With constrains (delay/rate/loss)
- Example: Media streaming
- usually delivered by UDP

What should happen inside the black box?



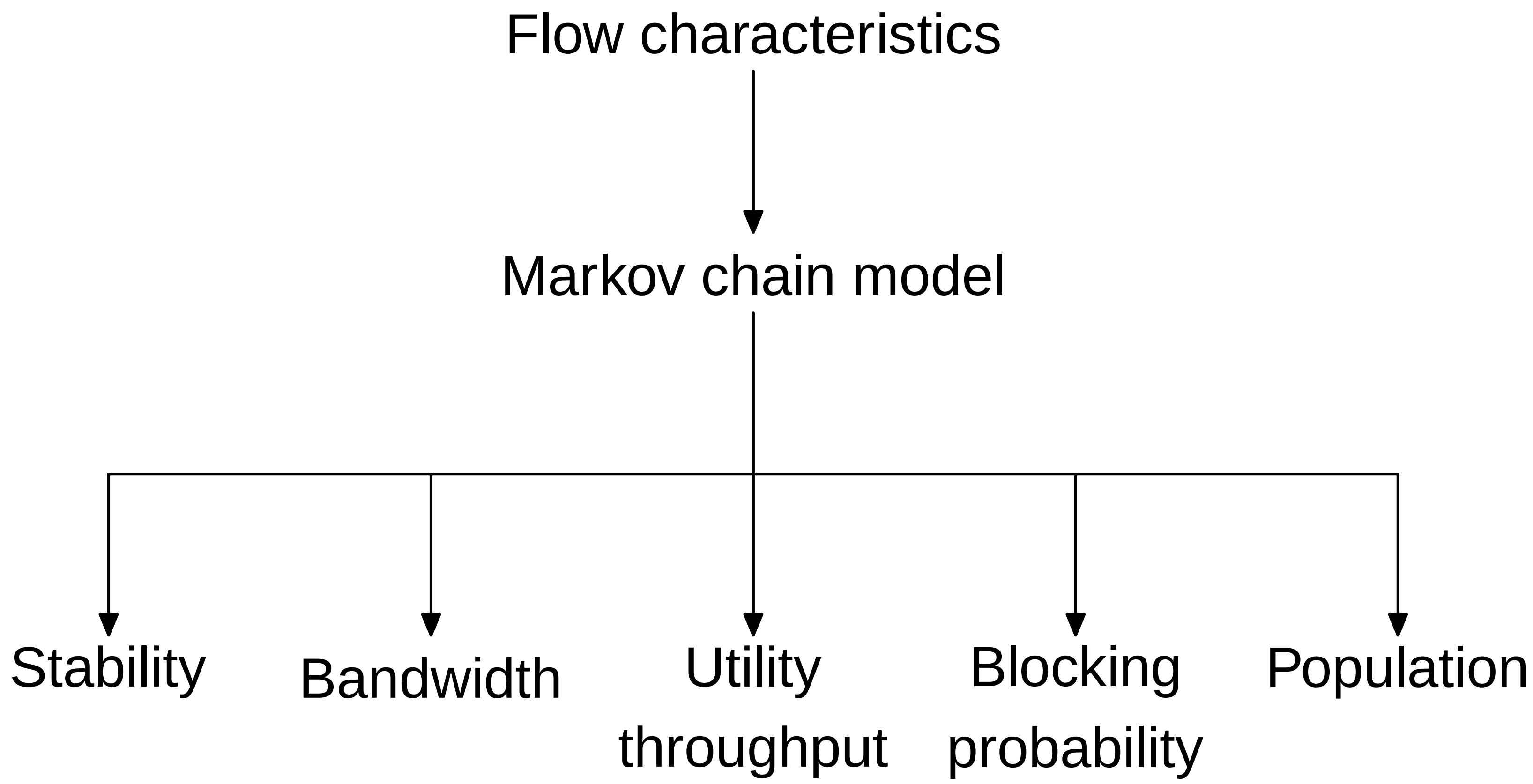
Different ‘regulations’ in the black box

We call them *control schemes*:

- No control
- Congestion control
- Admission control
- Admission control with continuous assurance

Result:
Admission control is no worse than congestion control

Roadmap


Flow characteristics
Markov chain model
Stability
Bandwidth
Utility throughput
Blocking probability
Population

Dichotomy of Flows

- Elastic flow can adapt to network conditions
  - It still functions if the network is slow, low bandwidth, high delay, . . .
  - Example: HTTP, FTP
- Inelastic flow cannot adapt
  - If bandwidth/delay is below the desired level, it is nearly useless
  - Example: VoIP, streaming

Problem Statement

- Elastic flows are adaptive to the available bandwidth
- Inelastic flows do not react to congestion,
  with constrain on min. data rate and delay

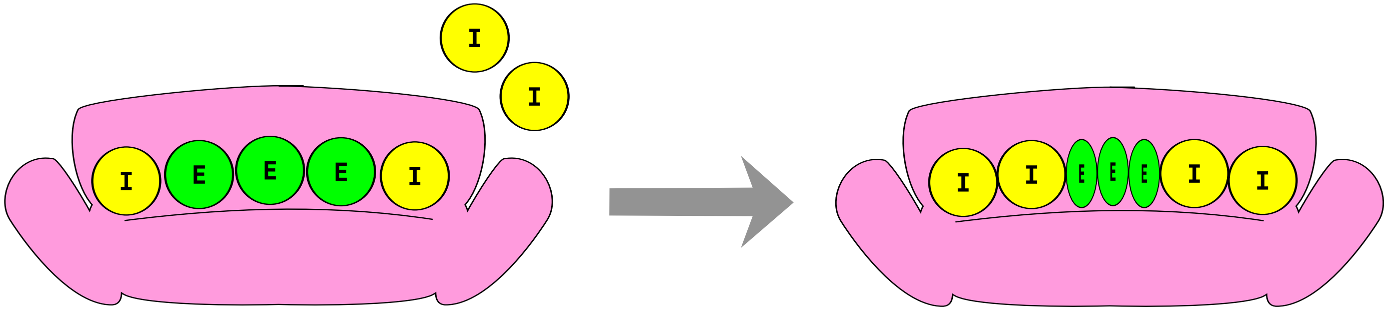
Problem Statement

- Elastic flows are adaptive to the available bandwidth
- Inelastic flows do not react to congestion,
  with constrain on min. data rate and delay

How should the elastic and inelastic flows coexist in the Internet?

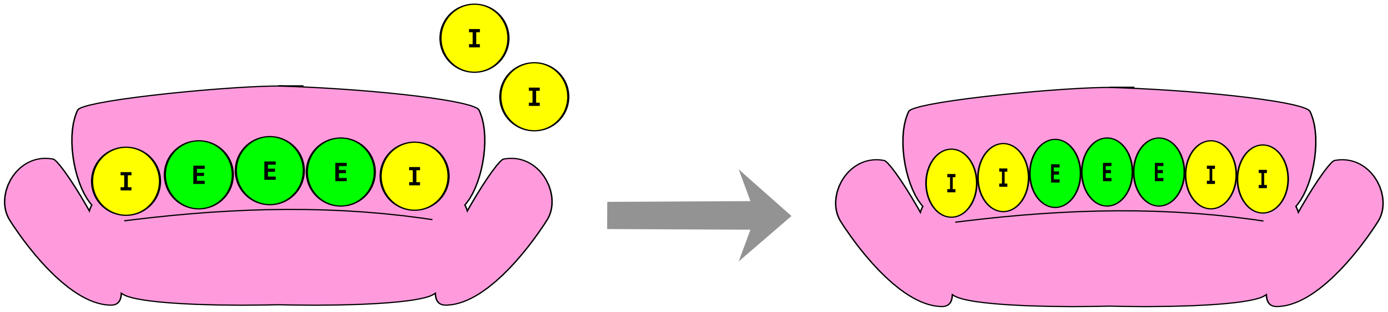
Solution A: No control

- Use UDP for multimedia transfer
- RTP over UDP to trace packet arrival times
- Problem: fairness with elastic flows is not guaranteed
  - Fear of congestion collapse



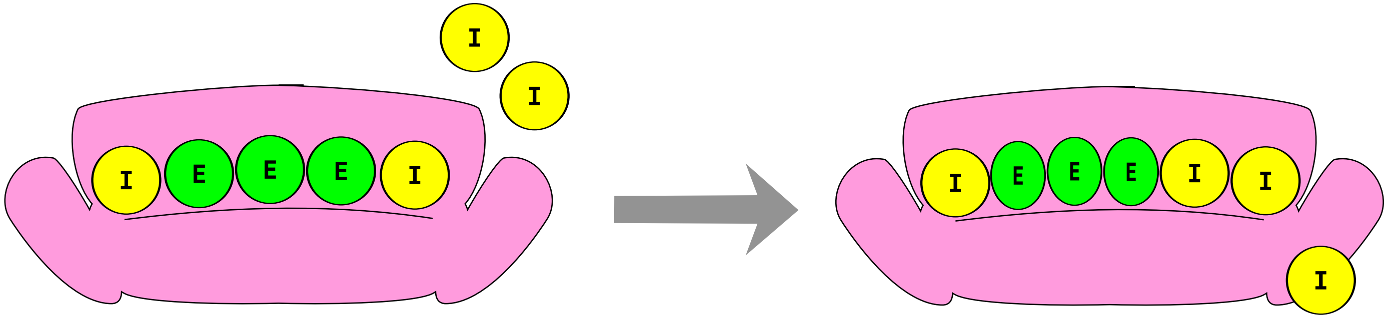
Solution B: Congestion Control

- IETF is working on TCP-friendly congestion control
- Requires inelastic flows to adapt, but allows them to adapt smoothly
- Inelastic flows *need* to be fair when using the network



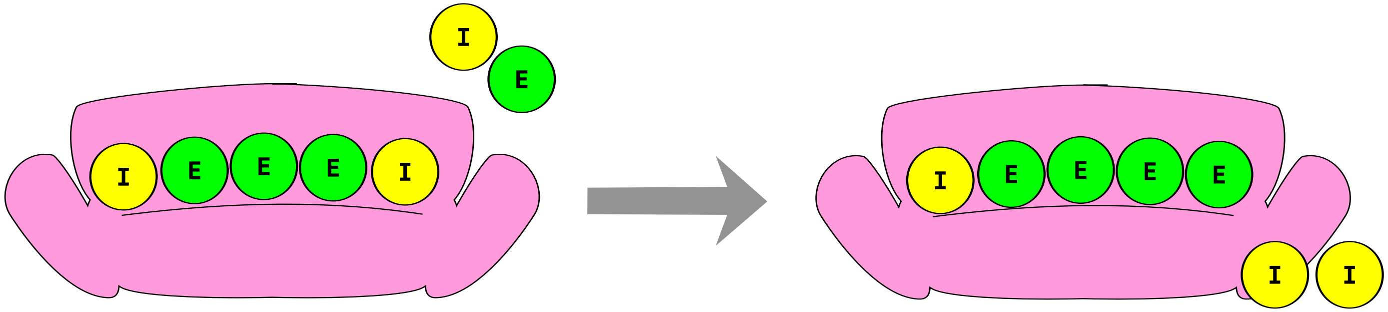
Solution C: Admission Control

- Similar to circuit switching: All or nothing
- Multimedia stay inelastic
  - Do not insist equal sharing of bandwidth
- Ensure the network can support before you use



Solution D: Admission Control with Continuous Assurance

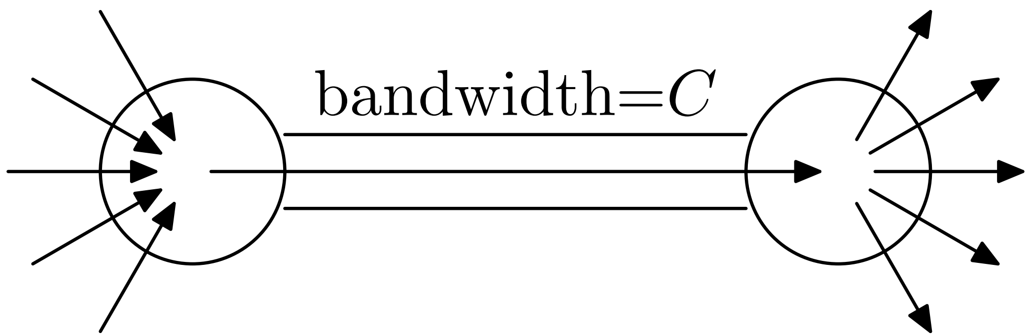
- Modified from Admission Control
- Consider inelastic flows:
  - Ensure the network can support before you use
  - When you are using, also make sure you don't make the network too congested



Model for Evaluation

Approximation by fluid model

- Network conditions are sensed by the flows instantly and the controls take effect immediately
- Single bottleneck link network



Markov Chain Model

- Applied with the fluid assumption
- State space: no. of elastic and inelastic flows, (n, m)
- Stochastic arrival, but the service rate depends on the flow controls

Flow Controls for Inelastic

1. No Control — multimedia over UDP
2. Congestion Control — TCP-friendly
3. Admission control in an “aggressive” way
4. Admission control in a “conservative” way
5. Admission control w/continue assurance in an “aggressive” way
6. Admission control w/continue assurance in a “conservative” way

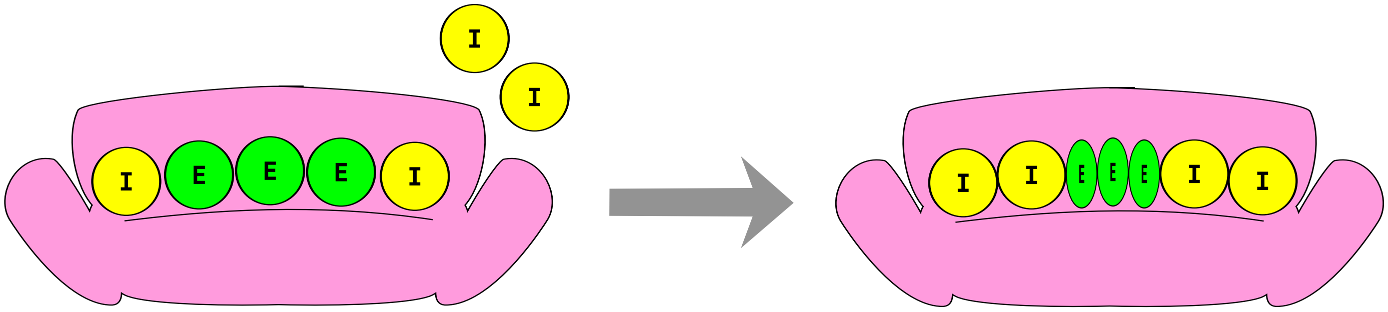
NC: No Control

Each inelastic flow uses α of bandwidth

- If there are n elastic and m inelastic flows,

	No.	Each	Total
Inelastic	m	α	$m\alpha$
Elastic	n	$\frac{1-m\alpha}{n}$	$1-m\alpha$
Total			1

- If $m\alpha > 1$, elastic flows get nothing and each inelastic flow has $1/m$

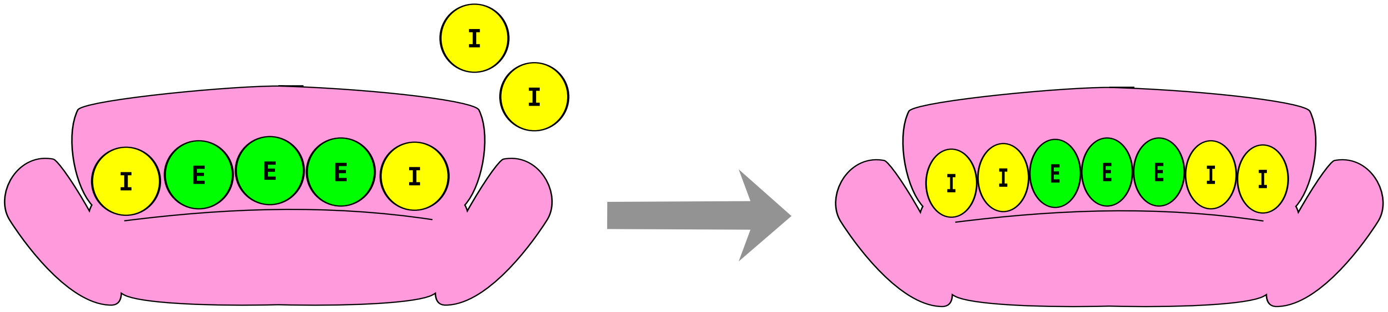


CC: Fair Share Congestion Ctrl

- If there are n elastic and m inelastic flows,

	No.	Each	Total
Inelastic	m	$\frac{1}{m+n}$	$\frac{m}{m+n}$
Elastic	n	$\frac{1}{m+n}$	$\frac{n}{m+n}$
Total			1

- If $\frac{1}{m+n} > \alpha$, each inelastic flow will use only α. Then each elastic flow will have $\frac{1-m\alpha}{n} > \frac{1}{m+n}$

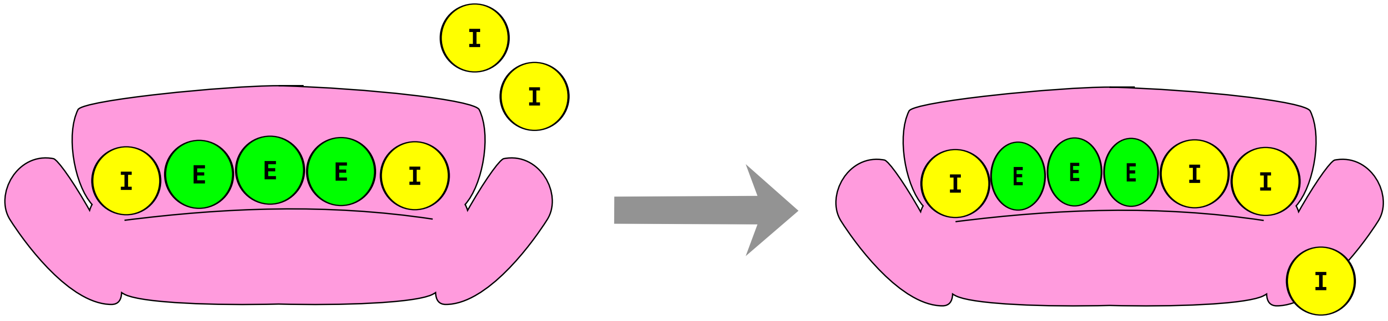


AC-A: Aggressive Admission Ctrl

- Assume an inelastic flow always take α of bandwidth
- Guarantee each elastic flow gets ϵ or more when admitting inelastic flows ($0 < \epsilon \ll \alpha$)

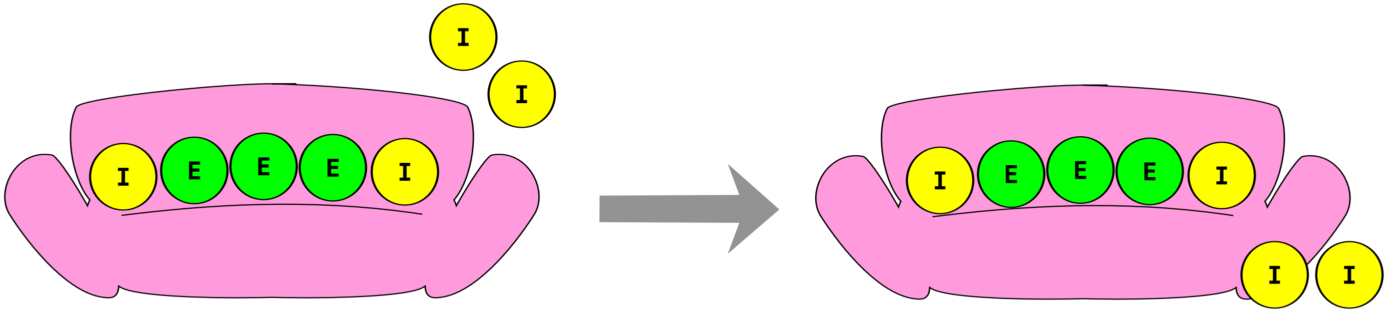
	No.	Each	Total
Inelastic	m	α	$m\alpha$
Elastic	n	$\frac{1-m\alpha}{n}$	$1 - m\alpha$
Total			1

- Admission only if $n\epsilon + (m+1)\alpha \leq 1$



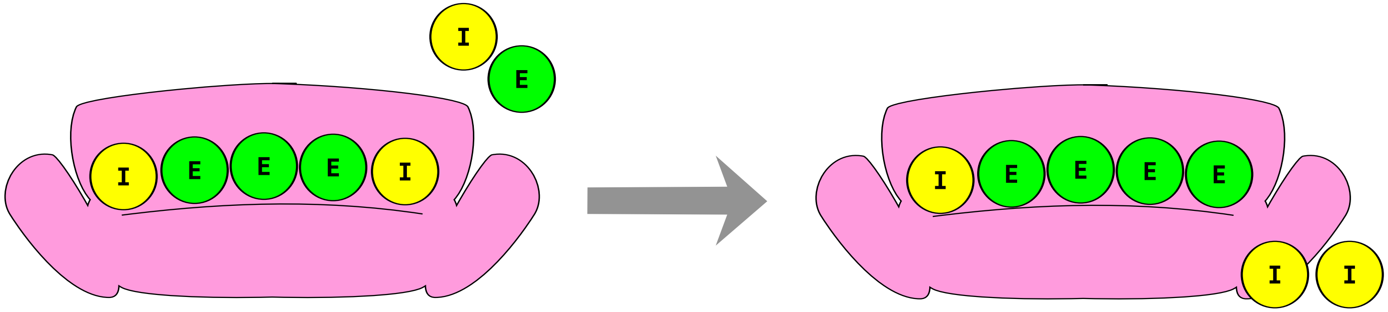
AC-C: Conservative Admission Ctrl

- $\epsilon = \alpha$
- Admission only if $(n + m + 1)\alpha \leq 1$
- We call this the “TCP-friendly admission control”

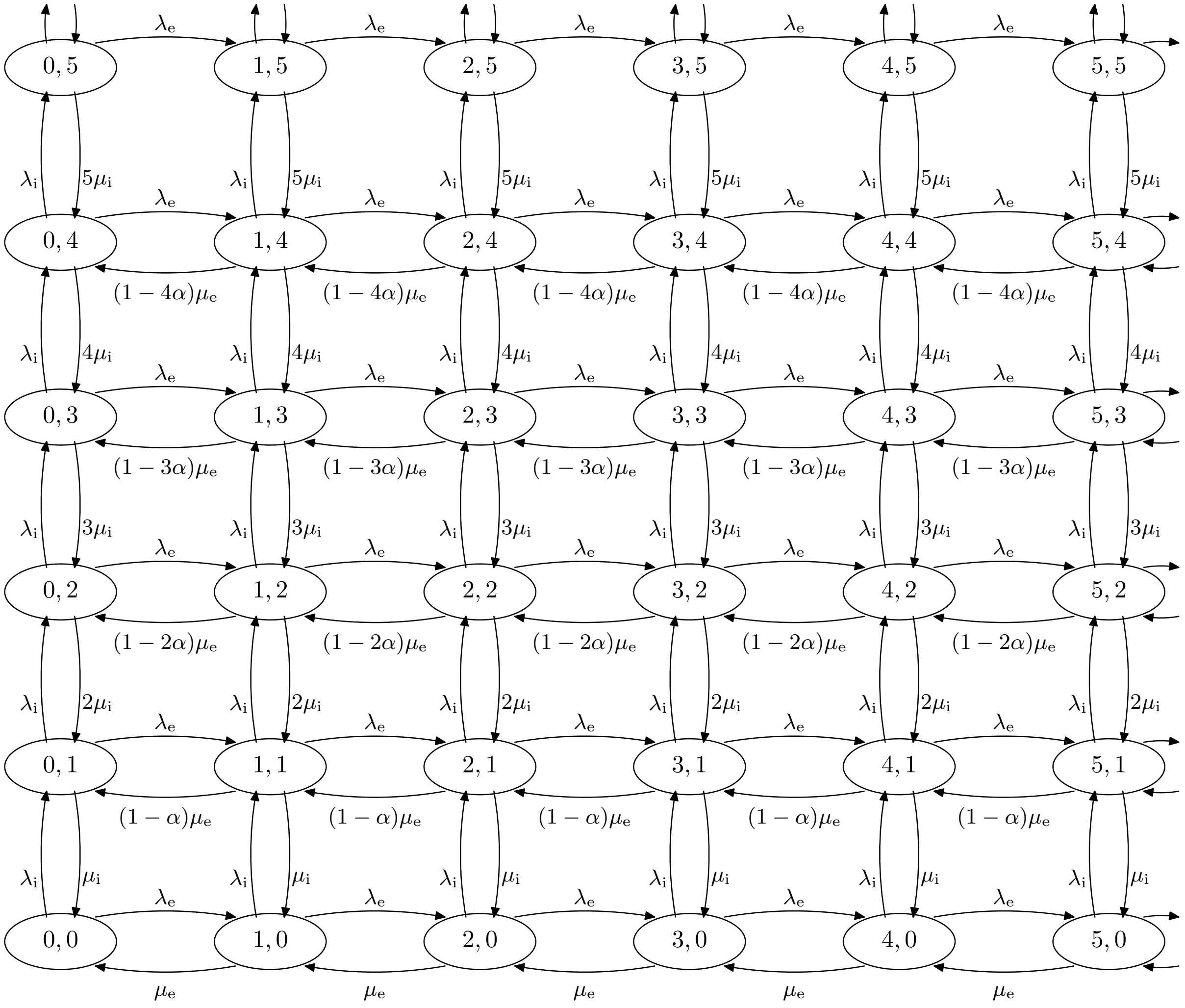


AA-A and AA-C: AC with Continuous Assurance

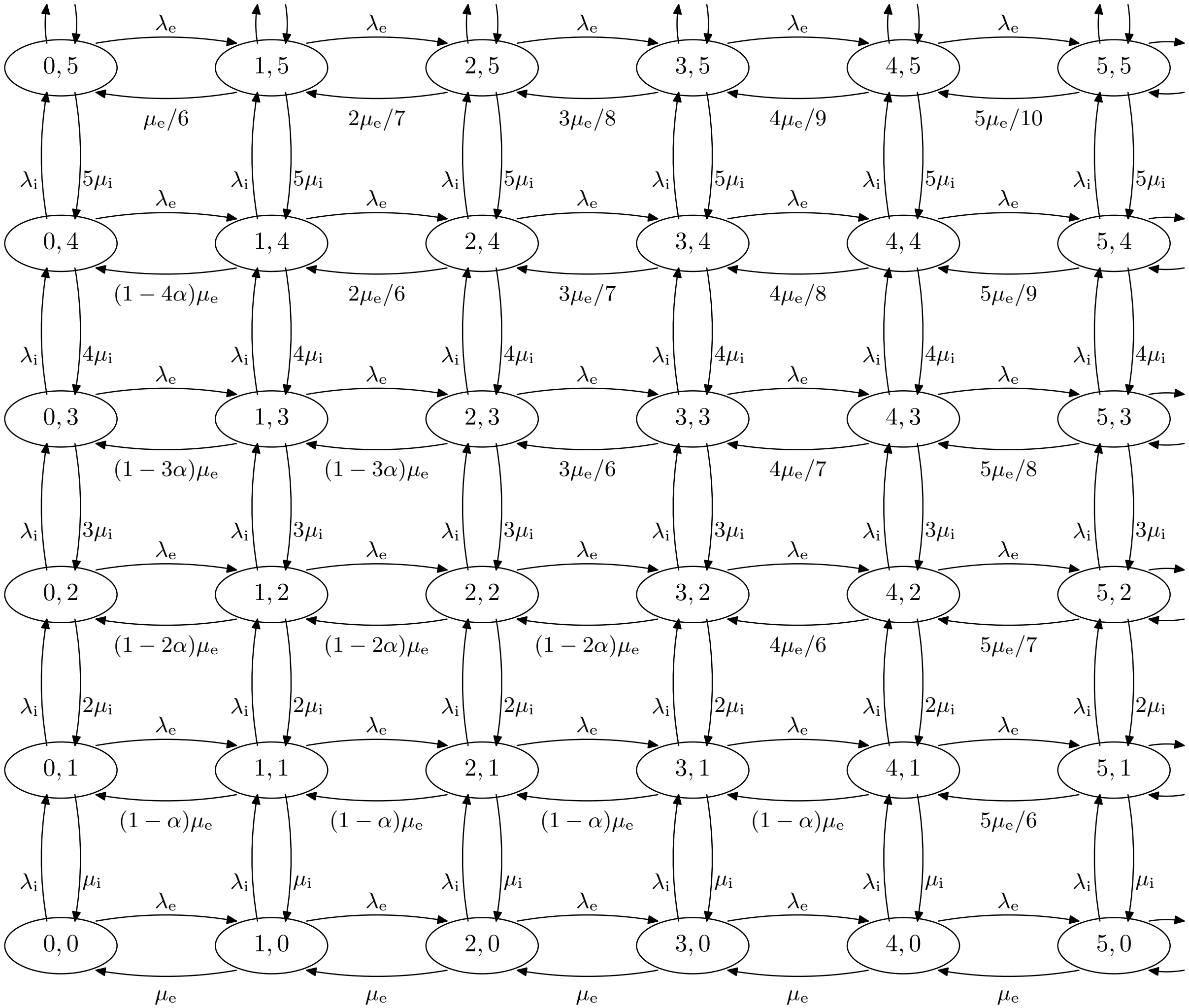
- Extension of AC-A and AC-C
- Also allows the inelastic flows to admit only if $n\epsilon + (m+1)\alpha \leq 1$
- Requires inelastic flows to continuously ensure $n\epsilon + m\alpha \leq 1$
  - Assure ϵ to each elastic flows continuously



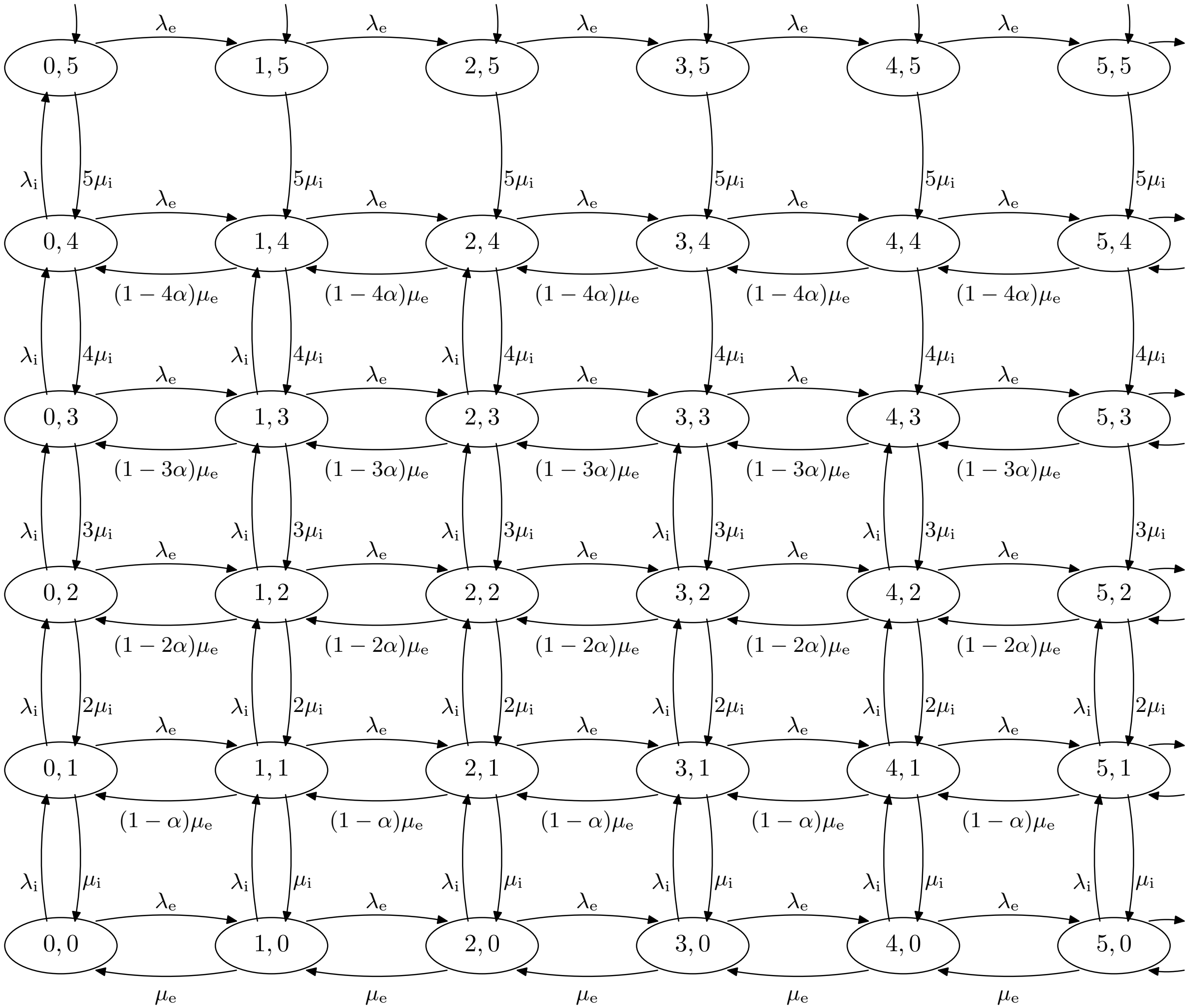
Markov Chain: NC



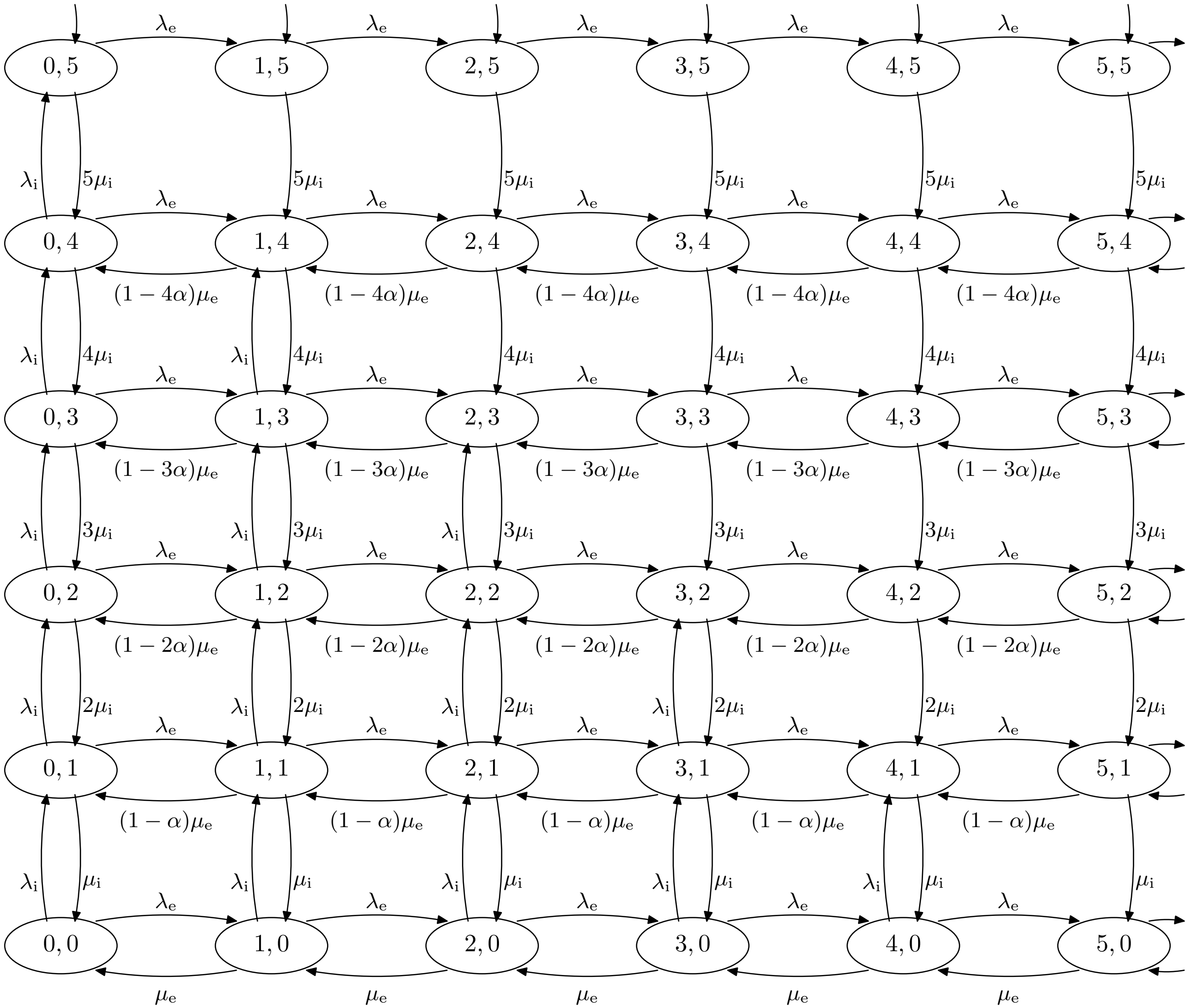
Markov Chain: CC



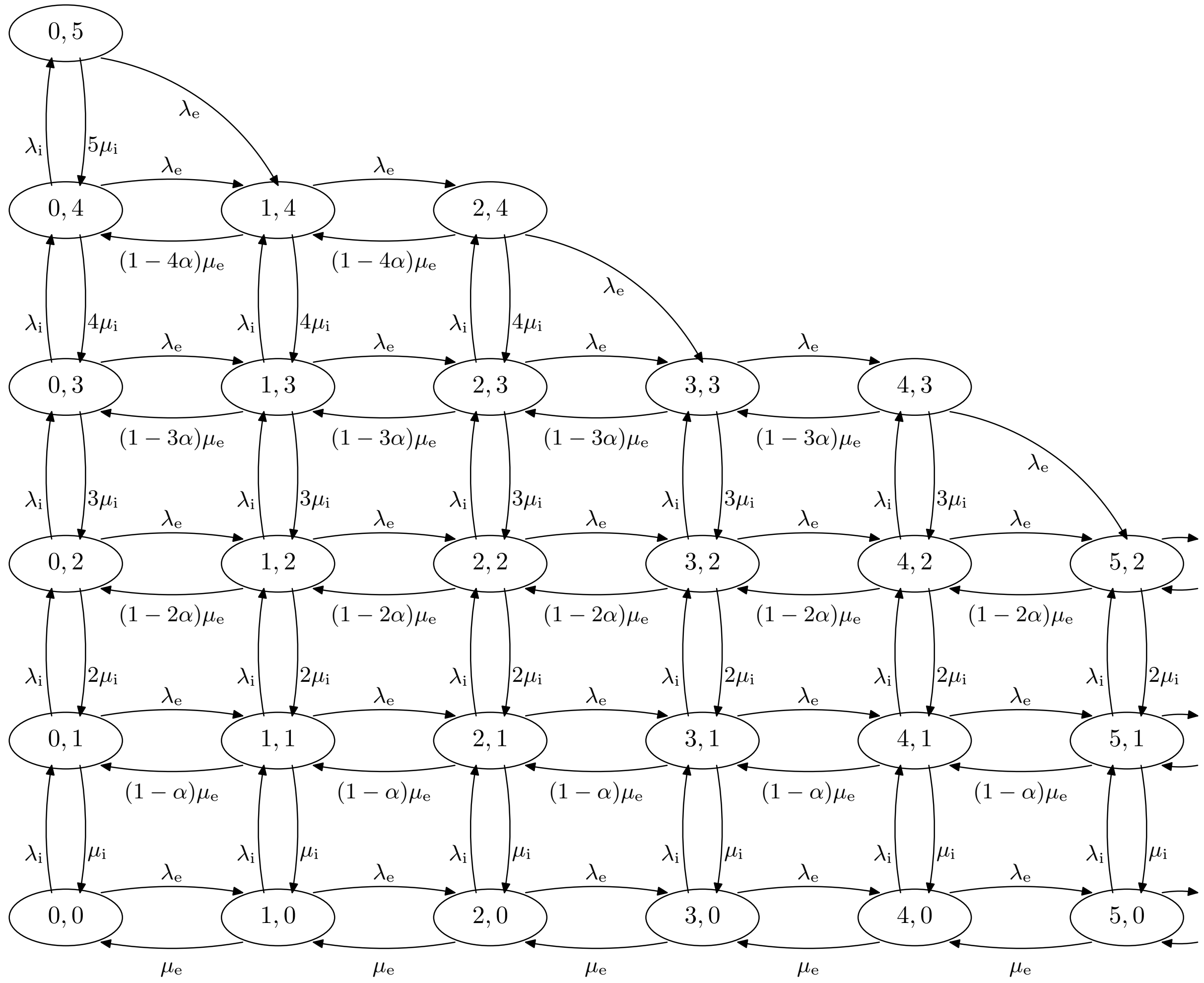
Markov Chain: AC-A



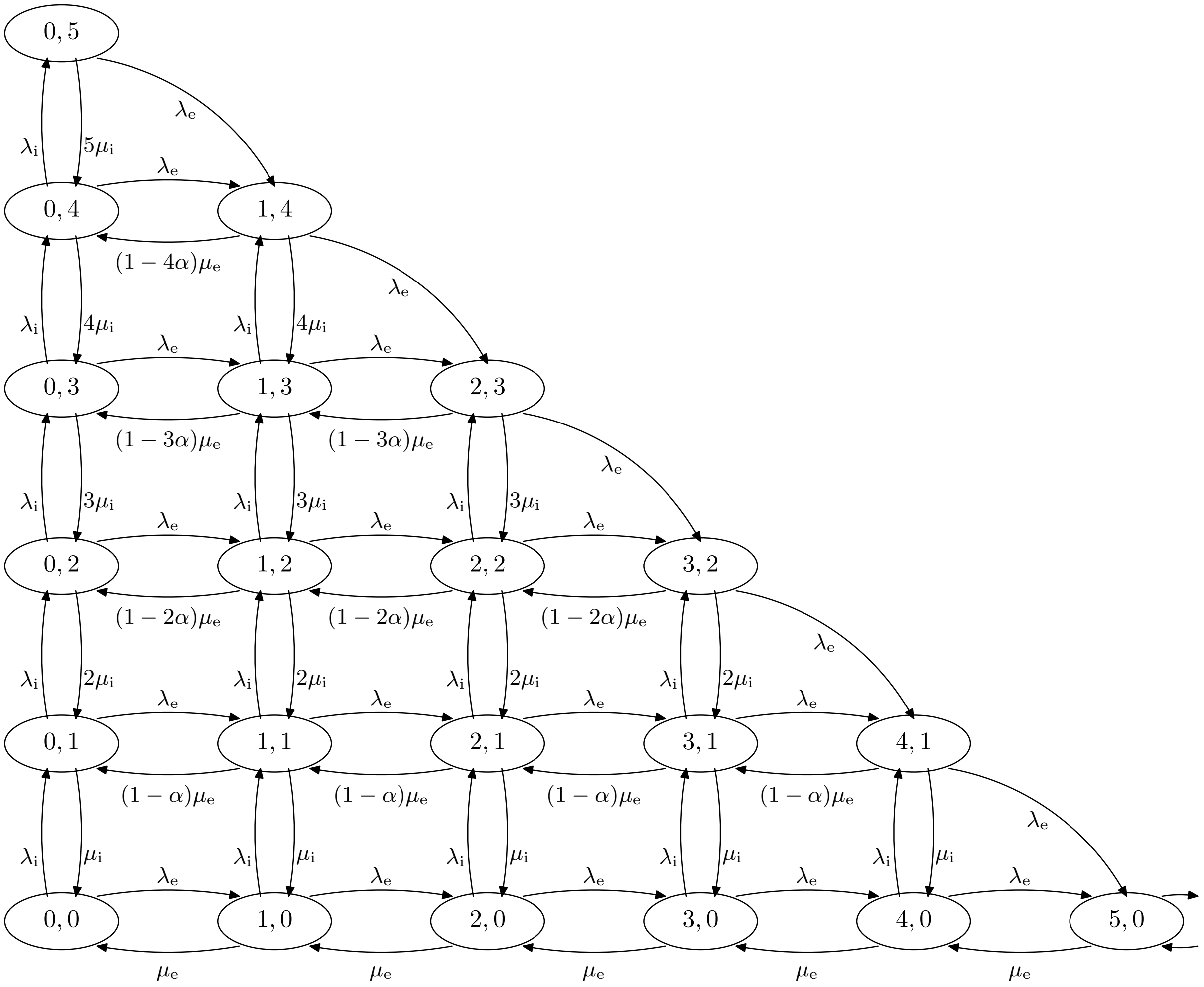
Markov Chain: AC-C



Markov Chain: AA-A



Markov Chain: AA-C



Markov Chain: Summary

Transition rates of Markov Chain:

		$(n,m) \to (n+1,m)$	$(n,m) \to (n,m+1)$	$(n,m) \to (n-1,m)$	$(n,m) \to (n,m-1)$	$(n,m) \to (n+1,m-1)$
NC	$m\alpha < 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
	$m\alpha \geq 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	0	$m\mu_{\rm i}$	0
CC	$(n+m)\alpha < 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
	$(n+m)\alpha \geq 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	$\frac{n}{m+n}\mu_{\rm e}$	$m\mu_{\rm i}$	0
AC-A	$n\epsilon+(m+1)\alpha \leq 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
	$n\epsilon+(m+1)\alpha > 1$	$\lambda_{\rm e}$	0	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
AC-C	$(n+m+1)\alpha \leq 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
	$(n+m+1)\alpha > 1$	$\lambda_{\rm e}$	0	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
AA-A	$n\epsilon+(m+1)\alpha \leq 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
	$1-\alpha < n\epsilon+m\alpha \leq 1-\epsilon$	$\lambda_{\rm e}$	0	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
	$(n+1)\epsilon+m\alpha > 1$	0	0	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	$\lambda_{\rm e}$
AA-C	$(n+m+1)\alpha \leq 1$	$\lambda_{\rm e}$	$\lambda_{\rm i}$	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	0
	$(n+m+1)\alpha > 1$	0	0	$(1-m\alpha)\mu_{\rm e}$	$m\mu_{\rm i}$	$\lambda_{\rm e}$

Define: $\rho = \rho_e + \alpha\rho_i; \quad \rho_e = \lambda_e/\mu_e; \quad \rho_i = \lambda_i/\mu_i$

Evaluations

- Stability
- Bandwidth allocation
- Utility throughput
- Blocking probability
- Population

Eval. 1: Stability

- Network as a server and flows as customers
- Pure elastic flows network: M/M/1-PS queue
- Pure inelastic flows network: M/M/∞-PS queue
- How is their mix?

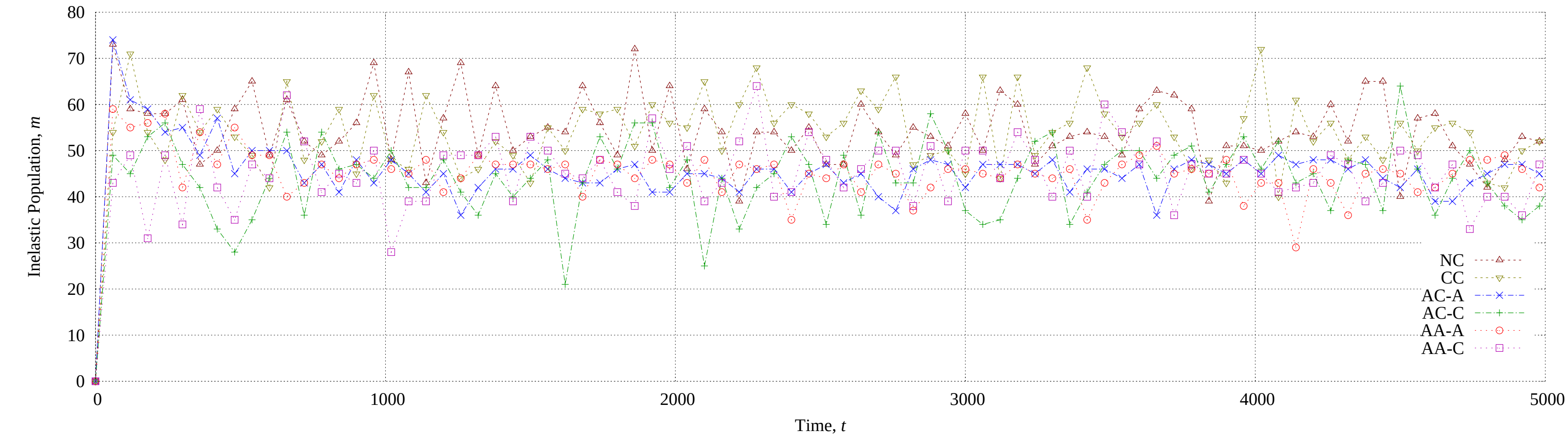
Eval. 1: Stability

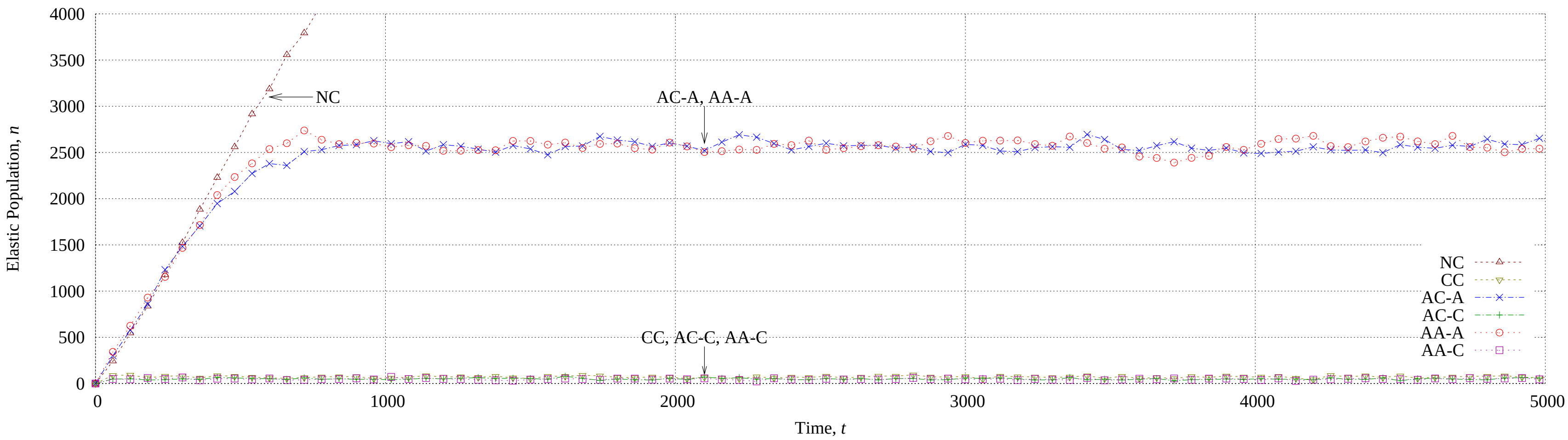
- Stability of queue: Avg queue length doesn't increase
- Inelastic: Leave whenever playback time expired
  - Never accumulate
- Elastic: Leave only if they finish the file transfer
  - Accumulate if not enough bandwidth

Eval. 1: Stability

- If the network is too congested,
  - NC: bandwidth to elastic flows can be zero
  - Other: limits the use of bandwidth by inelastic flows
- Therefore, NC is stable if the offered load $\rho < 1$
- Other is stable if the offered load by elastic flows $\rho_{\mathrm{e}} < 1$

Eval. 1: Stability

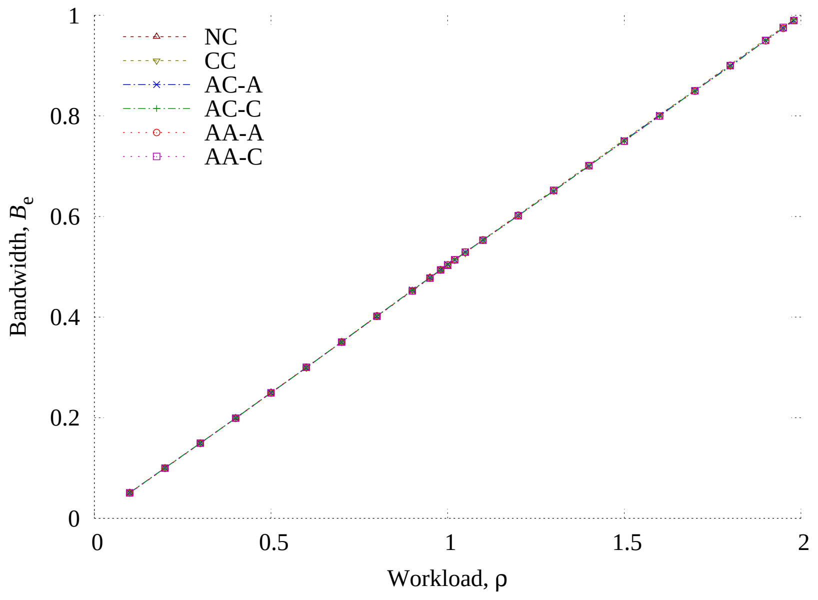


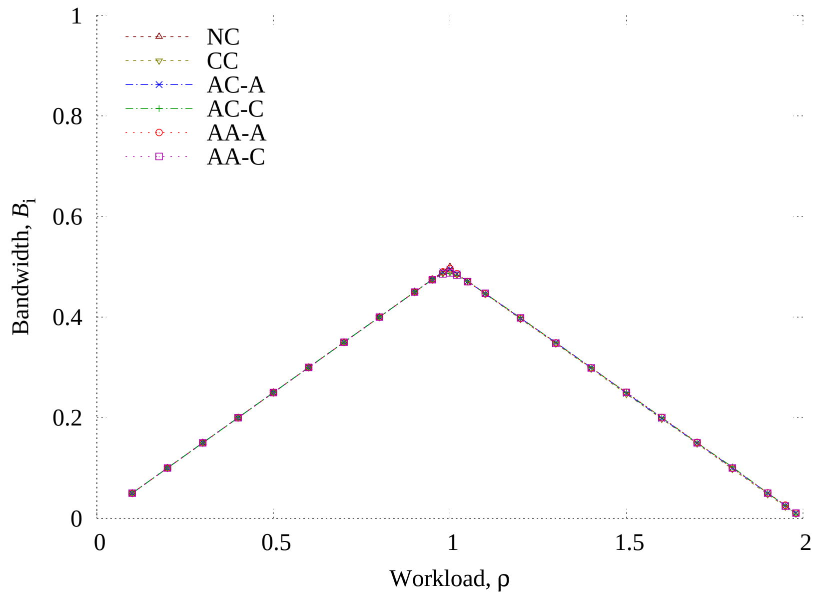


$$\rho = \rho_{\mathrm{e}} + \alpha\rho_{\mathrm{i}} = 1.1$$

Eval. 2: Bandwidth allocation

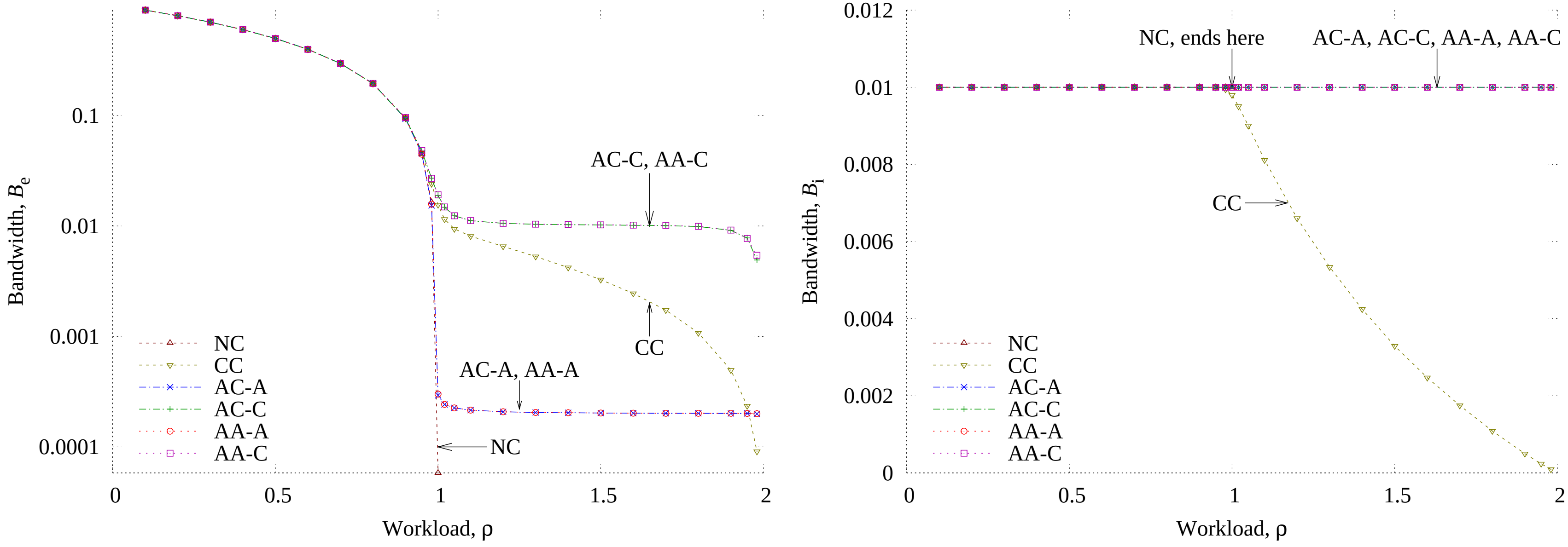
Aggregated bandwidth allocation





Eval. 2: Bandwidth allocation

Per-flow bandwidth allocation



Eval. 3: Utility throughput

- The network is serving many flows
- Each flow has some utility function
- Different controls $\Rightarrow$ Different bw. allocation
- The network's utility = Sum of the flows' utility
- Add up the utility of different flows—the better traffic control should yield higher total utility

Eval. 3: Utility throughput

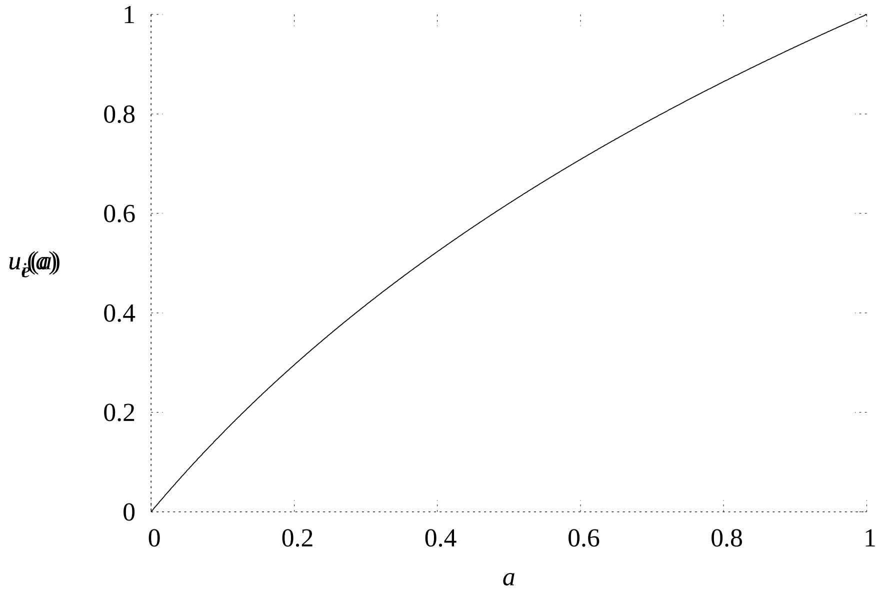
- Elastic: $u(x) = \log(x)$
  - Following Frank Kelly (proportional fairness, paper in 1997)
  - A concave function and monotonically increasing

Eval. 3: Utility throughput

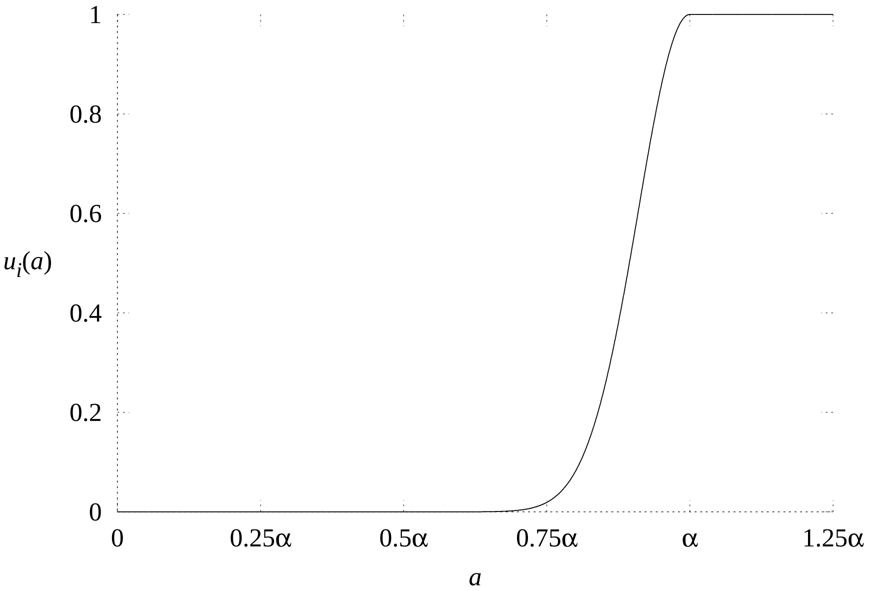
- Elastic: $u(x) = \log(x)$
  - Following Frank Kelly (proportional fairness, paper in 1997)
  - A concave function and monotonically increasing
- Inelastic: $u(x) = \sin^k(x)$
  - Steep decay in utility if the allocation is lower than desired rate
  - Over-allocation yields no value
  - This is known as a sigmoidal function

Eval. 3: Utility throughput

$$u_e(x) = \frac{\log(1+x)}{\log 2}$$



$$u_i(x) = \sin^{50}\left(\frac{\pi}{2}\frac{\min(x,\alpha)}{\alpha}\right)$$



Eval. 3: Utility throughput

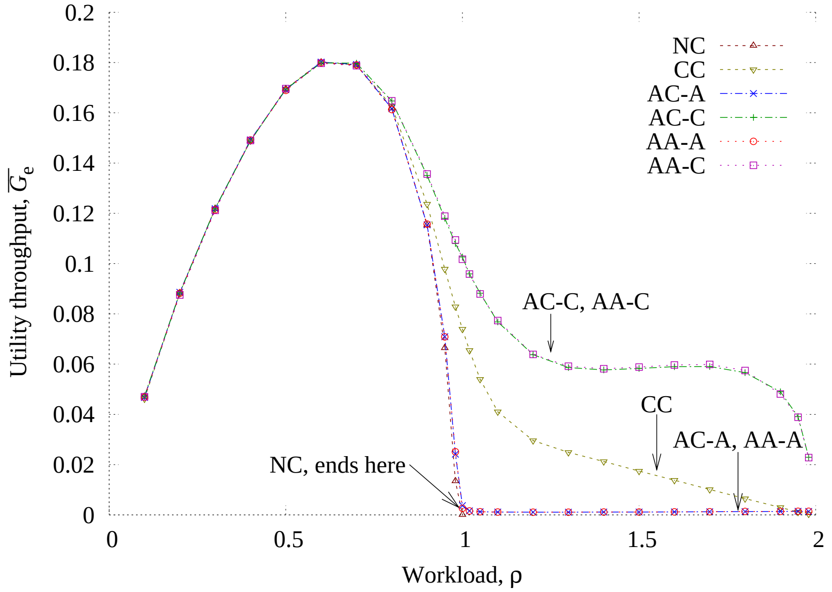
Utility throughput:
Expected aggregated utility gain per unit time

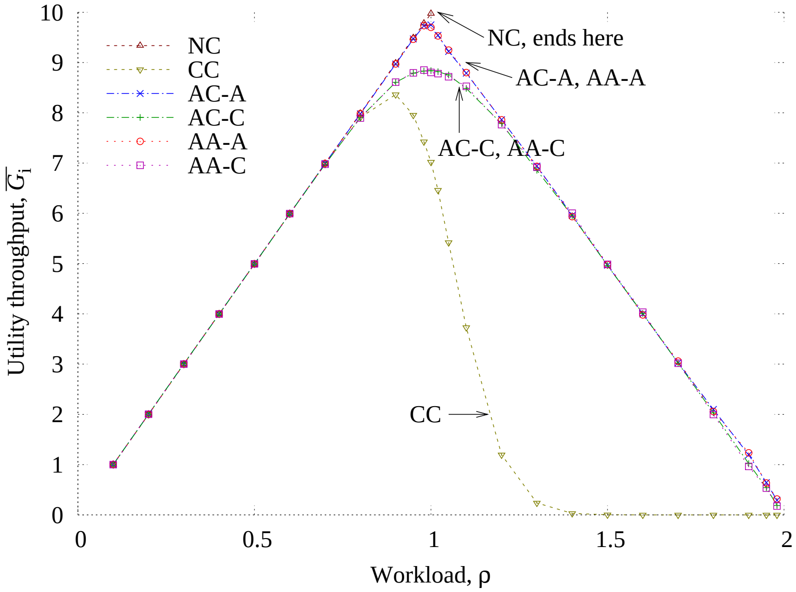
$$\bar{G}_{\mathrm{e}} = \sum_{n \neq 0} \sum_{m} n a_{\mathrm{e}}(n,m) u_{\mathrm{e}}\left(a_{\mathrm{e}}(n,m)\right) P[n,m]$$

$$\bar{G}_{\mathrm{i}} = \sum_{n} \sum_{m \neq 0} m u_{\mathrm{i}}\left(a_{\mathrm{i}}(n,m)\right) P[n,m]$$

Eval. 3: Utility throughput

- Simulating the Markov chain
- Result: AA-C, AC-C > AA-A, AC-A, CC > NC





Eval. 4: Blocking probability

- Focus: How to tune-up the admission control
- Comparing different admission controls do not need utility functions
- The performance of admission control is determined solely by blocking probability

Eval. 4: Blocking probability

- Consider only the admission control models
- Make use of Poisson Counter Driven Stochastic Differential Equation
- Defining
  - τ to be the total number of bytes yet to be transferred by all the existing flows, and
  - N_{i}, N_{e} to be Poisson counters marking the arrival of inelastic and elastic flows

Eval. 4: Blocking probability

Equation:

$$d\tau = -\mathbf{1}(\tau > 0)dt + S_{\mathrm{e}} dN_{\mathrm{e}} + I(n,m) S_{\mathrm{i}} dN_{\mathrm{i}}$$

evaluates to:

$$R = 1 - P_{\text{block}} = \frac{\Pr[\tau > 0] - \rho_{\mathrm{e}}}{\alpha \rho_{\mathrm{i}}}$$

Eval. 4: Blocking probability

$$R = 1 - P_{\text{block}} = \frac{\Pr[\tau > 0] - \rho_{\mathrm{e}}}{\alpha \rho_{\mathrm{i}}}$$

- $\Pr[\tau > 0]$ is the probability that the network is not idle
- Intuitively, we can approximate by:

$$\Pr[\tau > 0] \approx \min(\rho, 1)$$
$$\rho = \rho_{\mathrm{e}} + \alpha \rho_{\mathrm{i}}$$

Eval. 4: Blocking probability

$$R = 1 - P_{\text{block}} = \frac{\Pr[\tau > 0] - \rho_{\mathrm{e}}}{\alpha \rho_{\mathrm{i}}}$$

- $\Pr[\tau > 0]$ is the probability that the network is not idle
- Intuitively, we can approximate by:

$$\Pr[\tau > 0] \approx \min(\rho, 1)$$

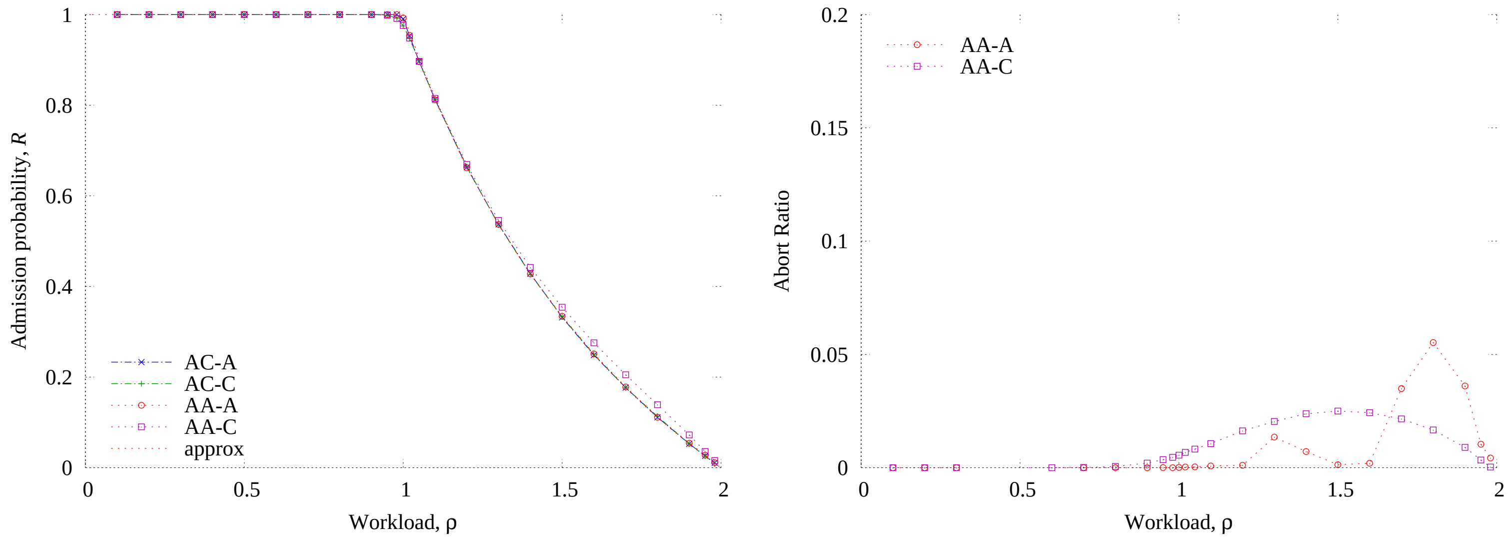
$$\rho = \rho_{\mathrm{e}} + \alpha \rho_{\mathrm{i}}$$

$$\therefore \quad R \approx \frac{\min(\rho, 1) - \rho_{\mathrm{e}}}{\alpha \rho_{\mathrm{i}}}$$

Eval. 4: Blocking probability

$$R \approx \frac{\min(\rho, 1) - \rho_{\mathrm{e}}}{\alpha \rho_{\mathrm{i}}}$$

- No ϵ in the equation!
- Whichever AC models, the same R



Eval. 5: Population

- Avg population = Avg no. of flows using the network
- Higher population $\Rightarrow$ Longer queue, longer delay
- Better control scheme shall give lower population (if the offered load is the same)

Eval. 5: Population

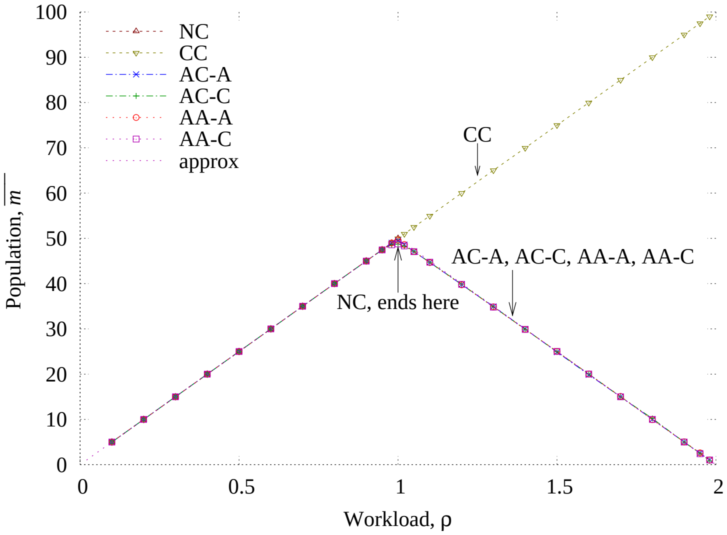
- Admission probability: $R \approx \dfrac{\min(\rho, 1) - \rho_{\mathrm{e}}}{\alpha \rho_{\mathrm{i}}}$
- Effective offered load by inelastic flows:

$$\rho_{\mathrm{i,eff.}} = R\rho_{\mathrm{i}} = \frac{\min(\rho, 1) - \rho_{\mathrm{e}}}{\alpha} = \bar{m}$$

- $\bar{m}$ is the mean no. of inelastic flows

Eval. 5: Population

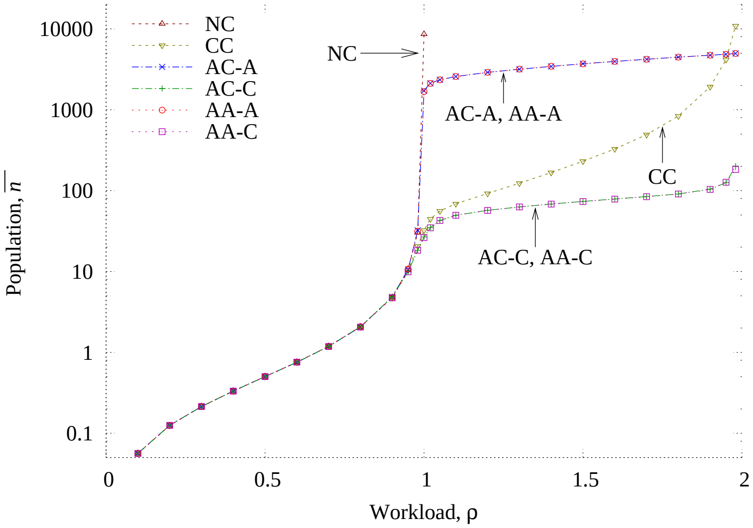
Inelastic population:



No different between different AC/AA control schemes
NC unstable at $\rho > 1$; CC keeps increasing

Eval. 5: Population

Elastic population:



Big difference between different control schemes

Eval. 5: Population

Recite: $R \approx \dfrac{\min(\rho, 1) - \rho_{\mathrm{e}}}{\alpha \rho_{\mathrm{i}}}$

- Being aggressive and selfish does not improve the performance
- In terms of social welfare, AC-C or AA-C should be chosen instead of AC-A or AA-A
  - pseudo-Nash equilibrium

Conclusion

- We argue for multimedia flows it is better to use admission control than TCP-friendly congestion control

Conclusion

- We argue for multimedia flows it is better to use admission control than TCP-friendly congestion control
- To make admission control TCP-friendly is easy:
  - Work as if you are normal TCP first
  - If (attained the rate you want)
    continue with your desired rate
    otherwise
    quit

Conclusion

- It does not pay to be too aggressive! You won't get any advantage in the long run

References

- *Network fairness for heterogeneous applications*

  Dah Ming Chiu and Adrian Tam
  In Proc. SIGCOMM Asia Workshop 2005
- *A case for TCP-friendly admission control*

  Adrian Tam, Dah Ming Chiu, John Lui, Y. C. Tay
  Submitted to Networking 2006
- *Redefining fairness in the study of TCP-friendly traffic controls*

  Dah Ming Chiu and Adrian Tam
  Submitted to IEEE Trans. on Networking