Recovering from inefficiencies in queueing systems with two consecutive waiting zones

Michalis Panayides

Supervisors:
Dr. Vince Knight,
Prof. Paul Harper

Ambulance blockage problem in UK

Patients forced to wait for 24 hours in ambulances, data shows

Ambulance crews forced to wait outside A&Es for 24 hours, according to chiefs

Rebecca Thomas Health Correspondent • Tuesday 17 May 2022 08:26 • Comments

(AFP/Getty)

'Appalling' waits for ambulances in England leaving lives at risk

Exclusive: Royal College of Emergency Medicine president says NHS is breaking its agreement to treat sickest in a timely way

- 'For staff, this is heartbreaking': senior doctor's view on crisis
- 'I feel so let down': long waits for ambulances in south-west

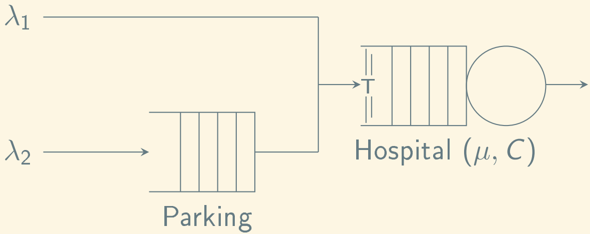
Ambulance handover delays highest since start of winter

NHS 'on its knees' as ambulance response times for life-threatening calls rise to record high

Average response time to deal with Category 1 cases – such as cardiac arrests – is now nine minutes and 20 seconds with rises across all categories.

Queues - Custom network of queues

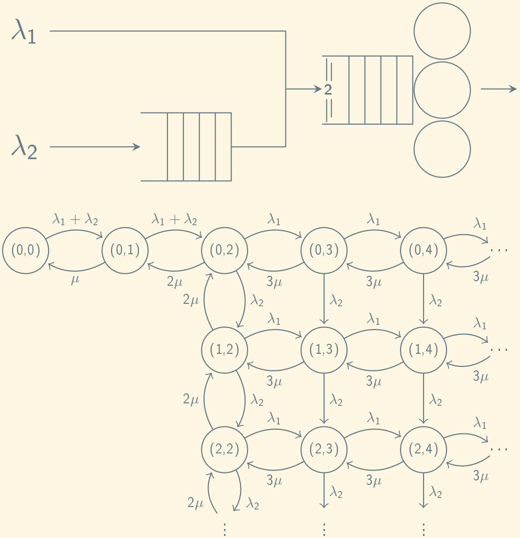
Queues - Custom network of queues



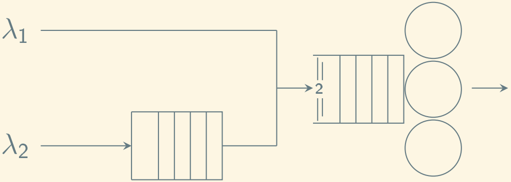
Parameters:

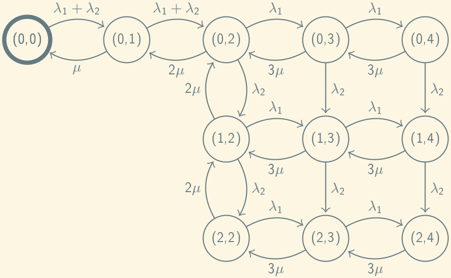
- λ_1: Arrival rate of type 1 individuals
- λ_2: Arrival rate of type 2 individuals
- μ: Service rate
- C: Number of servers
- T: Threshold

Markov Chain - Custom network

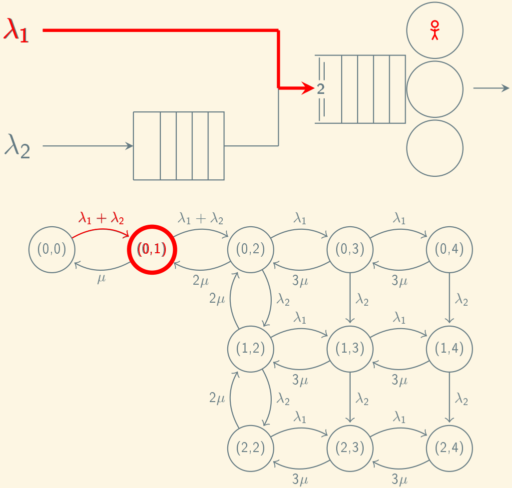


Markov Chain - Custom network - $N = 4, M = 2$

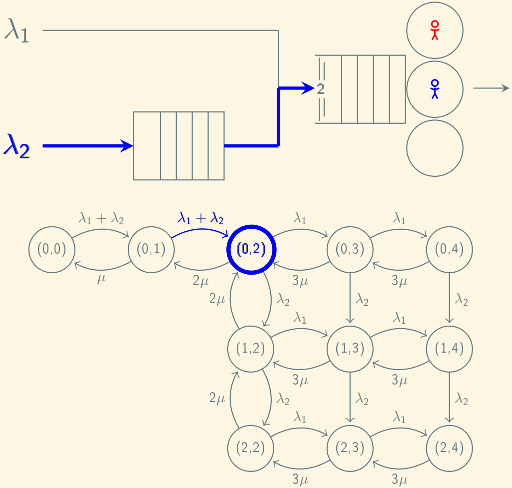




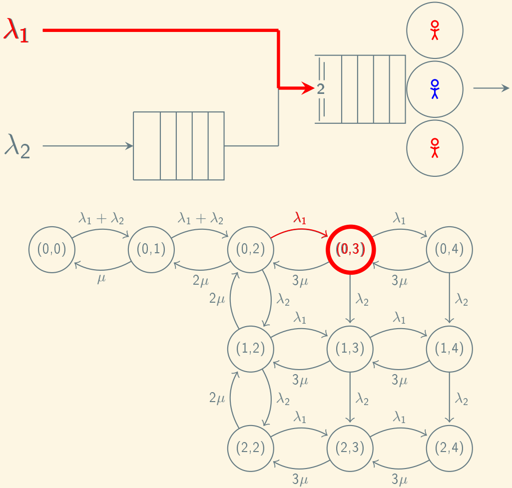
Markov Chain - Custom network - $N = 4, M = 2$



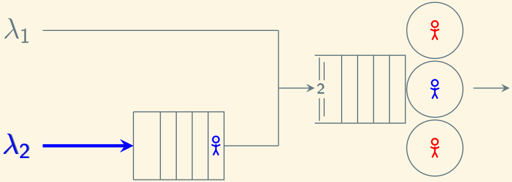
Markov Chain - Custom network - $N = 4, M = 2$

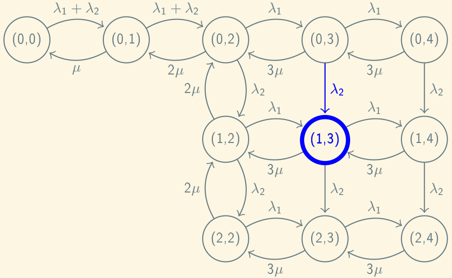


Markov Chain - Custom network - $N = 4, M = 2$

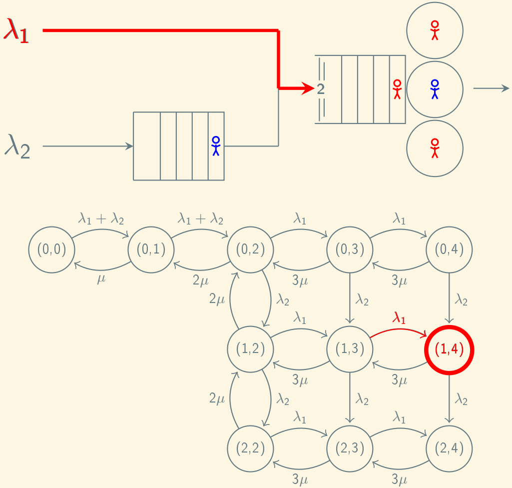


Markov Chain - Custom network - $N = 4, M = 2$

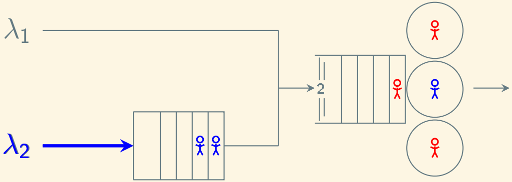


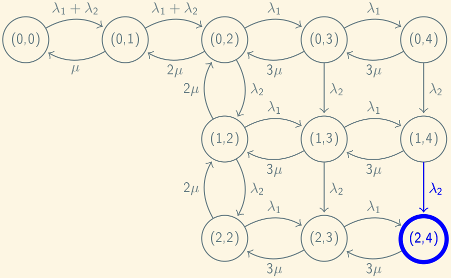


Markov Chain - Custom network - $N = 4, M = 2$

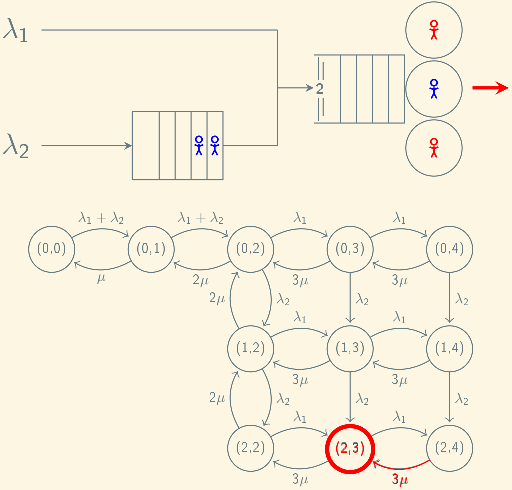


Markov Chain - Custom network - $N = 4$, $M = 2$

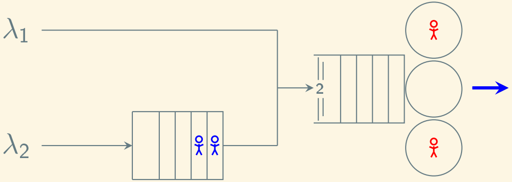


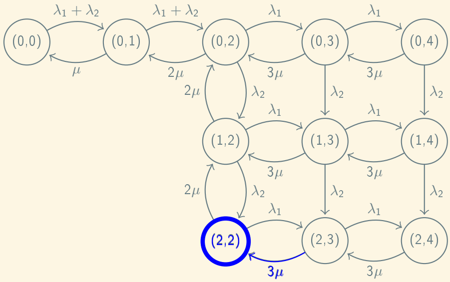


Markov Chain - Custom network - $N = 4, M = 2$

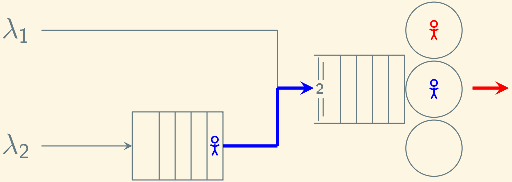


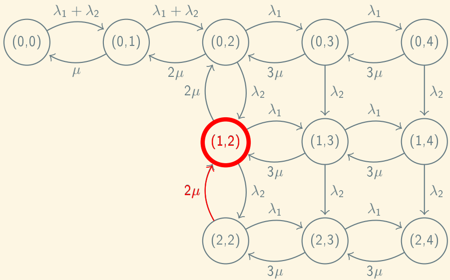
Markov Chain - Custom network - $N = 4$, $M = 2$



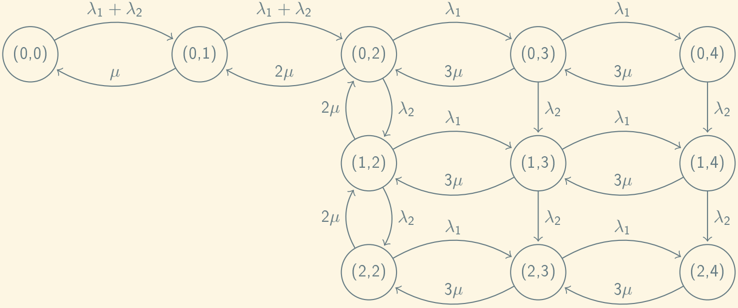


Markov Chain - Custom network - $N = 4, M = 2$





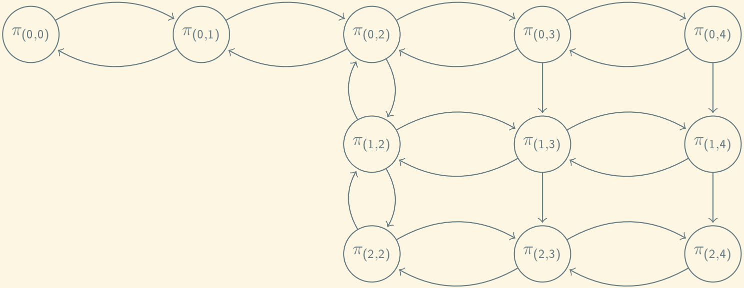
Steady state probabilities - Generator matrix



From \To	**(0,0)**	**(0,1)**	**(0,2)**		**(2,3)**	**(2,4)**
(0,0)	$-\lambda_1 - \lambda_2$	$\lambda_1 + \lambda_2$	0	$\dots$	0	0
(0,1)	μ	$-\mu - \lambda_1 - \lambda_2$	$\lambda_1 + \lambda_2$	$\dots$	0	0
(0,2)	0	2μ	$-2\mu - \lambda_1 - \lambda_2$	$\dots$	0	0
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
(2,3)	0	0	0	$\dots$	$-\lambda_1 - 3\mu$	λ_1
(2,4)	0	0	0	$\dots$	3μ	-3μ

Steady state probabilities - Generator matrix (Q)

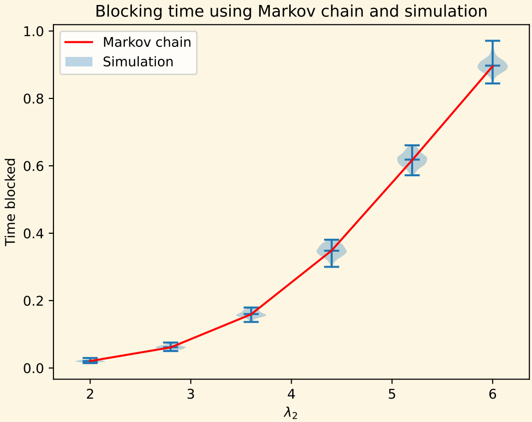
$$\pi = \begin{bmatrix} \pi_{(0,0)} & \pi_{(0,1)} & \pi_{(0,2)} & \dots & \pi_{(2,3)} & \pi_{(2,4)} \end{bmatrix}, \qquad \sum \pi_{(u,v)} = 1$$



$$\frac{d\pi}{dt} = \pi Q = 0$$

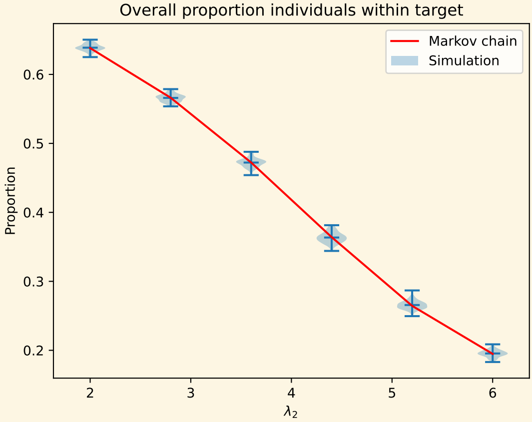
Performance Measures - Blocking time

$$B = \frac{\sum_{(u,v) \in S_A^{(2)}} \pi_{(u,v)} \; b(u,v)}{\sum_{(u,v) \in S_A^{(2)}} \pi_{(u,v)}}$$



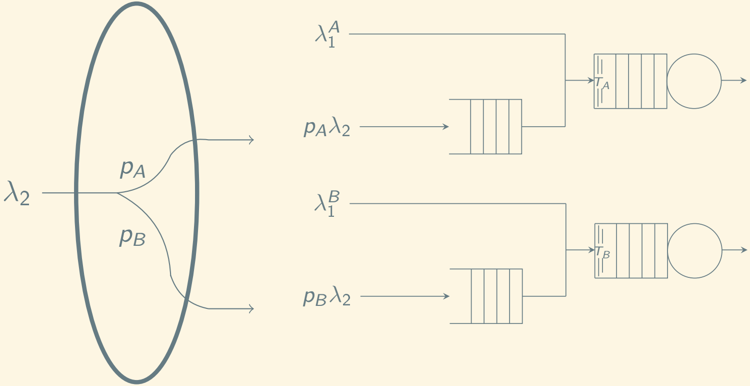
Performance Measures - Proportion within target

$$P(W < t) = \frac{\lambda_1 P_{L'_1}}{\lambda_2 P_{L'_2} + \lambda_1 P_{L'_1}} P(W^{(1)} < t) + \frac{\lambda_2 P_{L'_2}}{\lambda_2 P_{L'_2} + \lambda_1 P_{L'_1}} P(W^{(2)} < t)$$

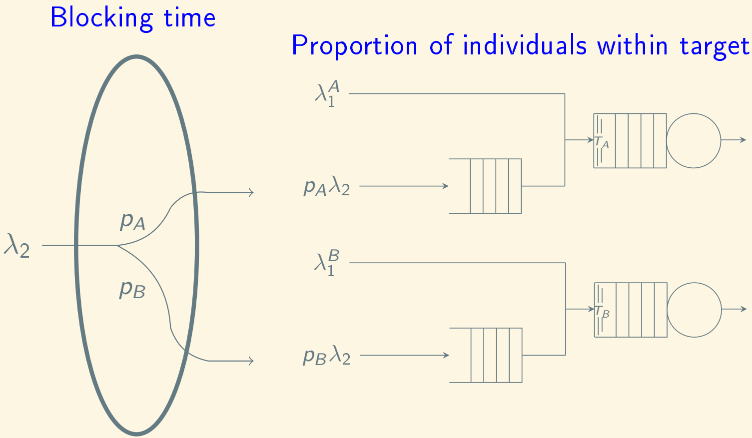


Game - Definition

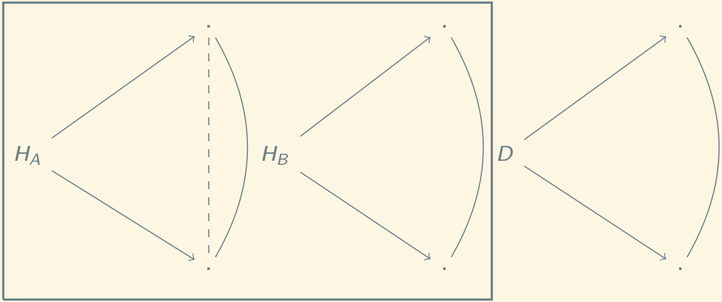
Game - Diagrammatic representation



Game - Diagrammatic representation



Game - Formulation



Hospital's utility

$$U^{(i)}_{T_A,T_B} = 1 - \left[\left(P(W^{(i)} < t) - 0.95\right)^2 \right]$$

Game - Payoff matrices

$$A = \begin{pmatrix} U^A_{1,1} & U^A_{1,2} & \cdots & U^A_{1,N_B} \\ U^A_{2,1} & U^A_{2,2} & \cdots & U^A_{2,N_B} \\ \vdots & \vdots & \ddots & \vdots \\ U^A_{N_A,1} & U^A_{N_A,2} & \cdots & U^A_{N_A,N_B} \end{pmatrix}, \quad B = \begin{pmatrix} U^B_{1,1} & U^B_{1,2} & \cdots & U^B_{1,N_B} \\ U^B_{2,1} & U^B_{2,2} & \cdots & U^B_{2,N_B} \\ \vdots & \vdots & \ddots & \vdots \\ U^B_{N_A,1} & U^B_{N_A,2} & \cdots & U^B_{N_A,N_B} \end{pmatrix}$$

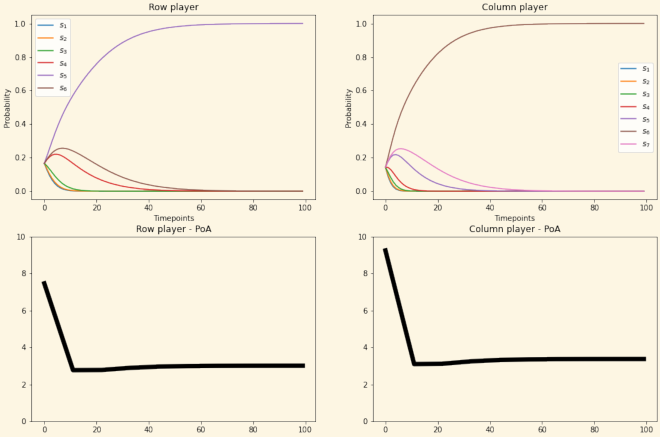
$$R = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N_B} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,N_B} \\ \vdots & \vdots & \ddots & \vdots \\ p_{N_A,1} & p_{N_A,2} & \cdots & p_{N_A,N_B} \end{pmatrix}$$

Asymmetric Replicator Dynamics

$$\frac{dx}{dt}_i = x_i((f_x)_i - \phi_x), \qquad \text{for all } i$$

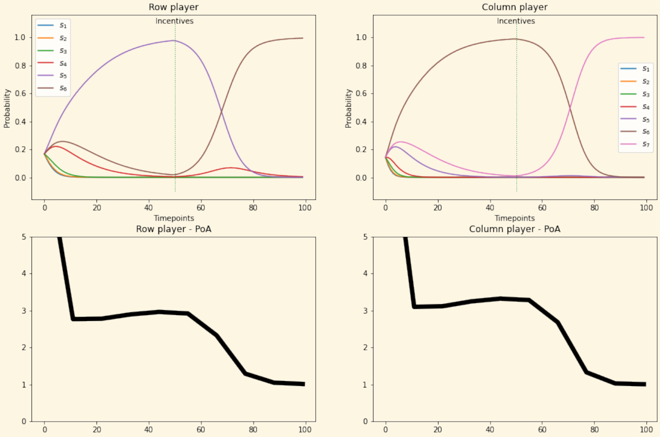
$$\frac{dy}{dt}_i = y_i((f_y)_i - \phi_y), \qquad \text{for all } i$$

Learning algorithms - Asymmetric replicator dynamics



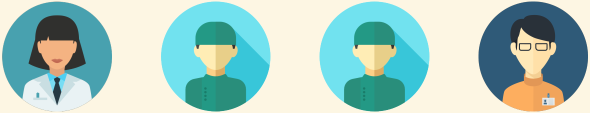
Inefficiencies can be learned and emerge naturally

Learning algorithms - Asymmetric replicator dynamics



Targeted incentivisation of behaviours can help escape learned inefficiencies

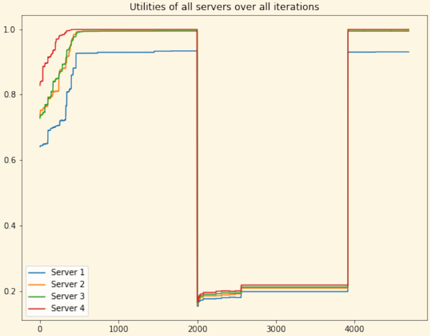
Reinforcement Learning - Server's behaviour

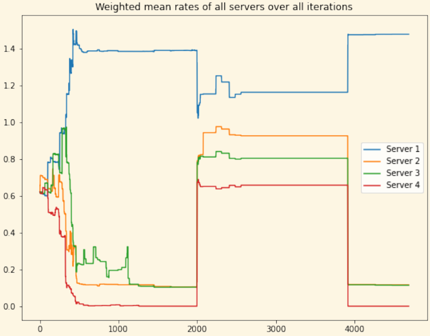


1 2 2 3

$$\overbrace{\text{Idle time} \quad + \quad \text{Lost inds}}^{\text{Utility}}$$

Reinforcement Learning - Server's behaviour





Thank you!

$ pip install ambulance_game

https://github.com/11michalis11/AmbulanceDecisionGame

PanayidesM@cardiff.ac.uk

@Michalis_Pan

@11michalis11