

A 3-player game theoretic model of a choice
between two queueing systems with strategic
managerial decision making

Michalis Panayides

EURO 2021 Athens

About me



About me



THIS.

Queues - Examples

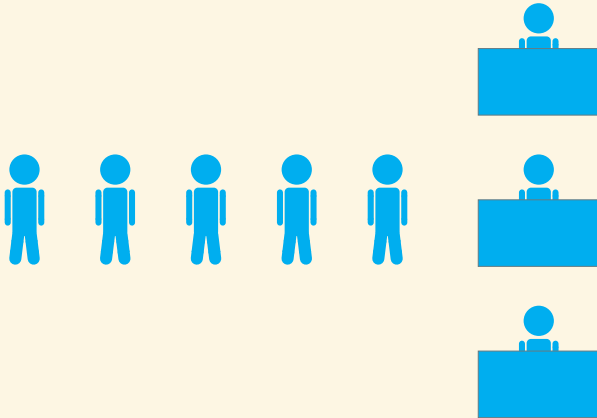


Queues - Examples

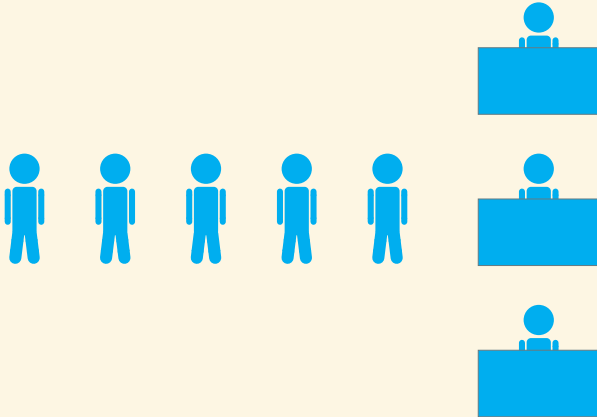


- ▶ Shone R, Knight VA, Williams JE. Comparisons between observable and unobservable M/M/1 queues with respect to optimal customer behaviour
- ▶ Kerner Y, Shmuel-Bittner O. Strategic behaviour and optimization in a hybrid M/M/1 queue with retrials.
- ▶ Gai Y, Liu H, Krishnamachari B. A packet dropping mechanism for efficient operation of M/M/1 queues with selfish users.

Queues - Examples



Queues - Examples

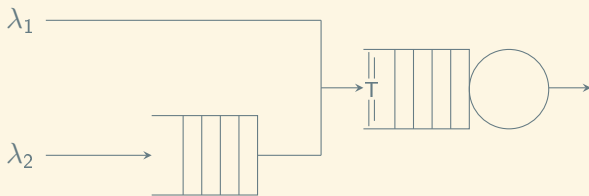


- ▶ Knight V, Harper P. The Impact of Choice on Public Services.
- ▶ Wang X, Song C, Zhuang J. Simulating a multi-stage screening network: A queueing theory and game theory application.

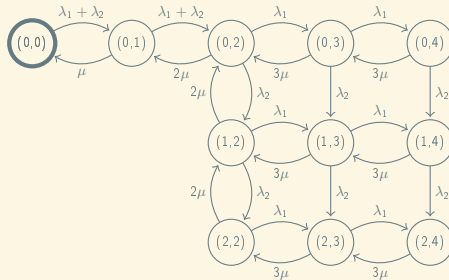
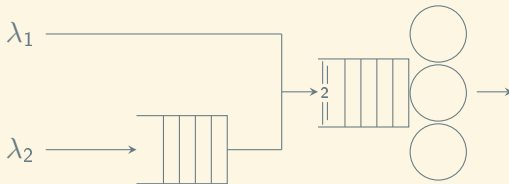
Queues - Examples



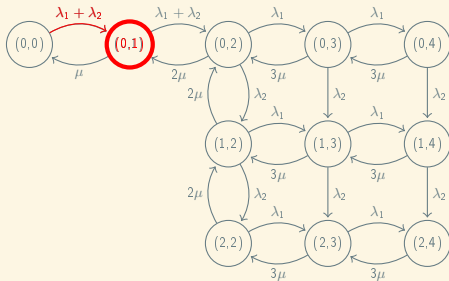
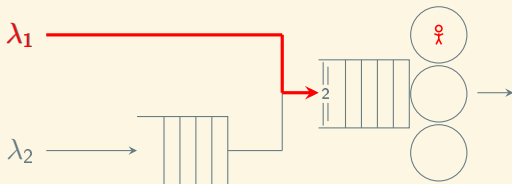
Queueing network structure



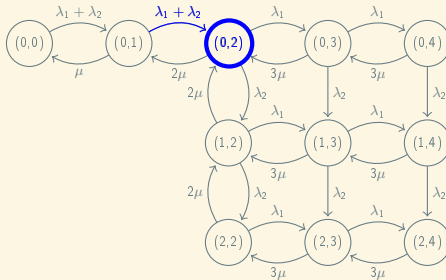
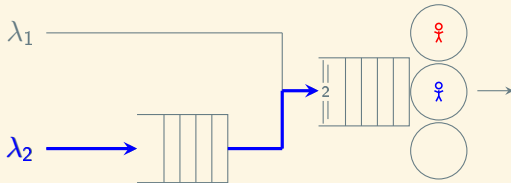
Markov Chain - Custom network



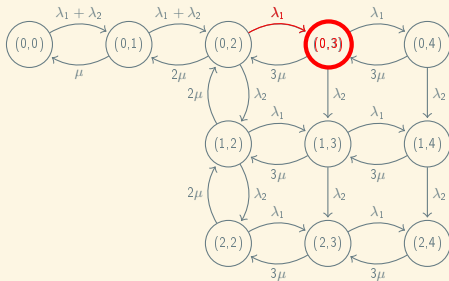
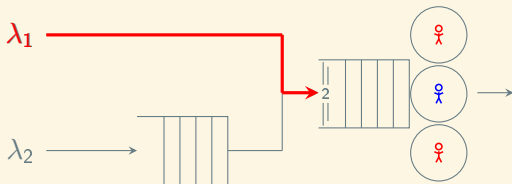
Markov Chain - Custom network



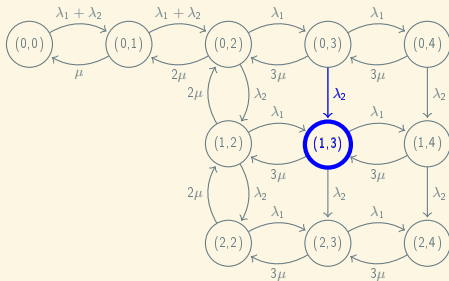
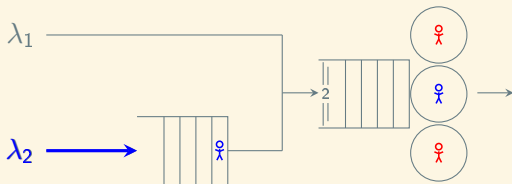
Markov Chain - Custom network



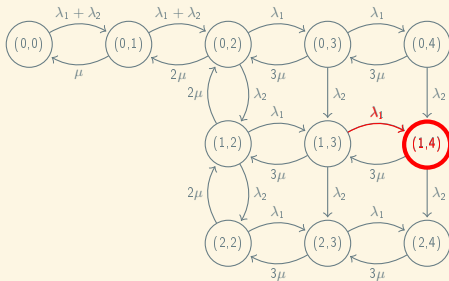
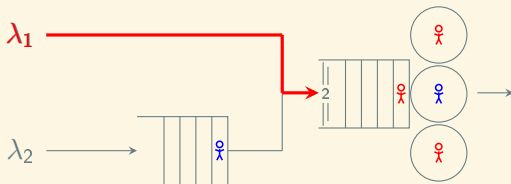
Markov Chain - Custom network



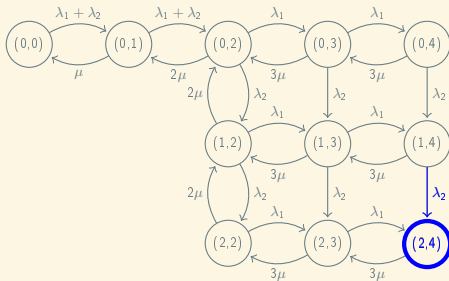
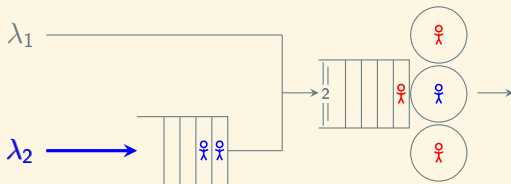
Markov Chain - Custom network



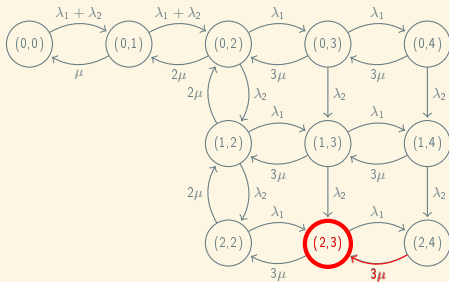
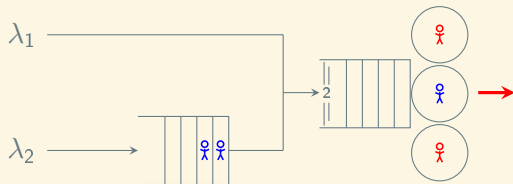
Markov Chain - Custom network



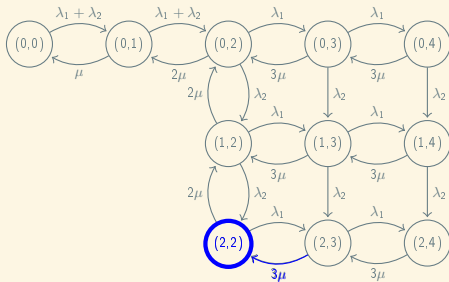
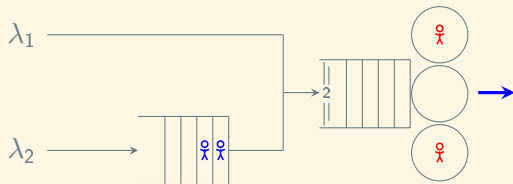
Markov Chain - Custom network



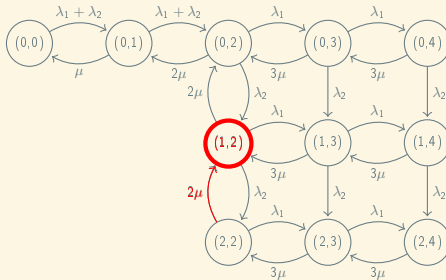
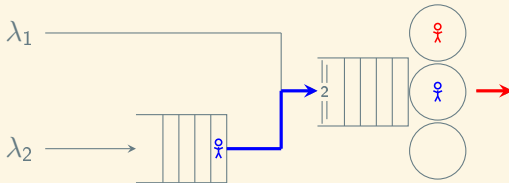
Markov Chain - Custom network



Markov Chain - Custom network

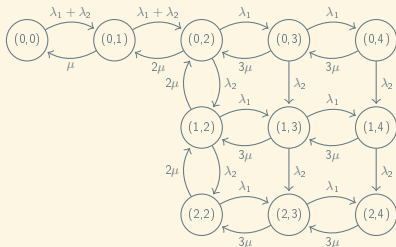


Markov Chain - Custom network



Steady state probabilities - Custom network

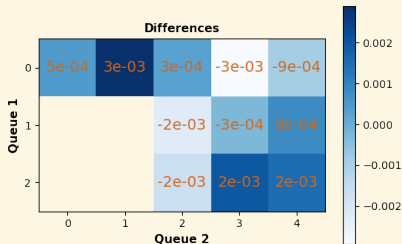
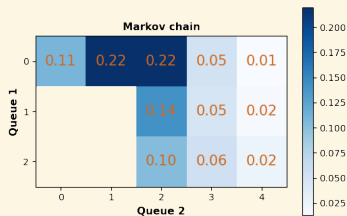
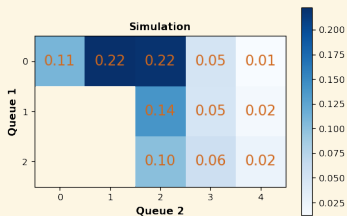
$$Q = \begin{matrix} & \begin{matrix} (0,0) & (0,1) & (0,2) & \dots & (2,3) & (2,4) \end{matrix} \\ \begin{matrix} (0,0) \\ (0,1) \\ (0,2) \\ \vdots \\ (2,3) \\ (2,4) \end{matrix} & \begin{pmatrix} -\lambda_1 - \lambda_2 & \lambda_1 + \lambda_2 & 0 & \dots & 0 & 0 \\ \mu & -\mu - \lambda_1 - \lambda_2 & \lambda_1 + \lambda_2 & \dots & 0 & 0 \\ 0 & 2\mu & -2\mu - \lambda_1 - \lambda_2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -\lambda_1 - 3\mu & \lambda_1 \\ 0 & 0 & 0 & \dots & 3\mu & -3\mu \end{pmatrix} \end{matrix}$$



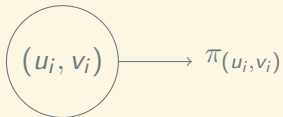
$$\frac{d\pi}{dt} = \pi Q = 0, \quad \sum \pi_{(u,v)} = 1$$

$$\pi = \begin{bmatrix} \pi(0,0) \\ \pi(0,1) \\ \pi(0,2) \\ \vdots \\ \pi(2,3) \\ \pi(2,4) \end{bmatrix}$$

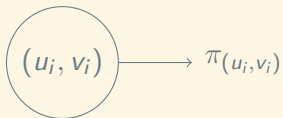
Steady state probabilities - Comparison



Performance Measures - Number of individuals



Performance Measures - Number of individuals



$$L = \sum_{i=1}^{|\pi|} \pi_i (u_i + v_i)$$

$$L_1 = \sum_{i=1}^{|\pi|} \pi_i v_i$$

$$L_2 = \sum_{i=1}^{|\pi|} \pi_i u_i$$

Performance Measures - Waiting time

$$W = \frac{\lambda_1 P_{L'_1}}{\lambda_2 P_{L'_2} + \lambda_1 P_{L'_1}} W^{(1)} + \frac{\lambda_2 P_{L'_2}}{\lambda_2 P_{L'_2} + \lambda_1 P_{L'_1}} W^{(2)}$$

$$W^{(1)} = \frac{\sum_{\substack{(u,v) \in S_A^{(1)} \\ v \geq C}} \frac{1}{C_\mu} \times (v - C + 1) \times \pi(u, v)}{\sum_{(u,v) \in S_A^{(1)}} \pi(u, v)}$$

$$W^{(2)} = \frac{\sum_{\substack{(u,v) \in S_A^{(2)} \\ \min(v, T) \geq C}} \frac{1}{C_\mu} \times (\min(v + 1, T) - C) \times \pi(u, v)}{\sum_{(u,v) \in S_A^{(2)}} \pi(u, v)}$$

Performance Measures - Waiting time

$$W^{(1)} = \frac{\sum_{(u,v) \in S_A^{(1)}} w^{(1)}(u,v) \pi(u,v)}{\sum_{(u,v) \in S_A^{(1)}} \pi(u,v)}, \quad W^{(2)} = \frac{\sum_{(u,v) \in S_A^{(2)}} w^{(2)}(u,v) \pi(u,v)}{\sum_{(u,v) \in S_A^{(2)}} \pi(u,v)}$$

Performance Measures - Waiting time

$$W^{(1)} = \frac{\sum_{(u,v) \in S_A^{(1)}} w^{(1)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(1)}} \pi_{(u,v)}}, \quad W^{(2)} = \frac{\sum_{(u,v) \in S_A^{(2)}} w^{(2)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(2)}} \pi_{(u,v)}}$$

$$S_A^{(1)} = \{(u,v) \in S \mid v < N\}, \quad S_A^{(2)} = \begin{cases} \{(u,v) \in S \mid u < M\} & \text{if } T \leq N \\ \{(u,v) \in S \mid v < N\} & \text{otherwise} \end{cases}$$

Performance Measures - Waiting time

$$W^{(1)} = \frac{\sum_{(u,v) \in S_A^{(1)}} w^{(1)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(1)}} \pi_{(u,v)}}, \quad W^{(2)} = \frac{\sum_{(u,v) \in S_A^{(2)}} w^{(2)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(2)}} \pi_{(u,v)}}$$

$$S_A^{(1)} = \{(u,v) \in S \mid v < N\}, \quad S_A^{(2)} = \begin{cases} \{(u,v) \in S \mid u < M\} & \text{if } T \leq N \\ \{(u,v) \in S \mid v < N\} & \text{otherwise} \end{cases}$$

$$S_W = \{(u,v) \in S \mid v > C\}$$

Performance Measures - Waiting time

$$W^{(1)} = \frac{\sum_{(u,v) \in S_A^{(1)}} w^{(1)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(1)}} \pi_{(u,v)}}, \quad W^{(2)} = \frac{\sum_{(u,v) \in S_A^{(2)}} w^{(2)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(2)}} \pi_{(u,v)}}$$

$$S_A^{(1)} = \{(u,v) \in S \mid v < N\}, \quad S_A^{(2)} = \begin{cases} \{(u,v) \in S \mid u < M\} & \text{if } T \leq N \\ \{(u,v) \in S \mid v < N\} & \text{otherwise} \end{cases}$$

$$S_W = \{(u,v) \in S \mid v > C\}$$

$$c^{(1)}(u,v) = \begin{cases} 0, & \text{if } u > 0 \text{ and } v = T \\ \frac{1}{\min(v,C)\mu}, & \text{otherwise} \end{cases}, \quad c^{(2)}(u,v) = \begin{cases} 0, & \text{if } u > 0 \\ \frac{1}{\min(v,C)\mu}, & \text{otherwise} \end{cases}$$

Performance Measures - Waiting time

$$W^{(1)} = \frac{\sum_{(u,v) \in S_A^{(1)}} w^{(1)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(1)}} \pi_{(u,v)}}, \quad W^{(2)} = \frac{\sum_{(u,v) \in S_A^{(2)}} w^{(2)}(u,v) \pi_{(u,v)}}{\sum_{(u,v) \in S_A^{(2)}} \pi_{(u,v)}}$$

$$S_A^{(1)} = \{(u,v) \in S \mid v < N\}, \quad S_A^{(2)} = \begin{cases} \{(u,v) \in S \mid u < M\} & \text{if } T \leq N \\ \{(u,v) \in S \mid v < N\} & \text{otherwise} \end{cases}$$

$$S_W = \{(u,v) \in S \mid v > C\}$$

$$c^{(1)}(u,v) = \begin{cases} 0, & \text{if } u > 0 \text{ and } v = T \\ \frac{1}{\min(v,C)\mu}, & \text{otherwise} \end{cases}, \quad c^{(2)}(u,v) = \begin{cases} 0, & \text{if } u > 0 \\ \frac{1}{\min(v,C)\mu}, & \text{otherwise} \end{cases}$$

$$w^{(i)}(u,v) = \begin{cases} 0, & \text{if } (u,v) \notin S_W \\ c^{(i)}(u,v) + w^{(i)}(u-1,v), & \text{if } u > 0 \text{ and } v = T \\ c^{(i)}(u,v) + w^{(i)}(u,v-1), & \text{otherwise} \end{cases}$$

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 - 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 - Blocking time

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

$$b(u, v) = \begin{cases} 0, & \text{if } (u, v) \notin S_b \\ c(u, v) + b(u-1, v), & \text{if } v = N = T \\ c(u, v) + b(u, v-1), & \text{if } v = N \neq T \\ c(u, v) + p_s(u, v)b(u-1, v) + p_a(u, v)b(u, v+1), & \text{if } u > 0 \text{ and } v = T \\ c(u, v) + p_s(u, v)b(u, v-1) + p_a(u, v)b(u, v+1), & \text{otherwise} \end{cases}$$

Performance Measures - Blocking time

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

$$b(u, v) = \begin{cases} 0, & \text{if } (u, v) \notin S_b \\ c(u, v) + b(u-1, v), & \text{if } v = N = T \\ c(u, v) + b(u, v-1), & \text{if } v = N \neq T \\ c(u, v) + p_s(u, v)b(u-1, v) + p_a(u, v)b(u, v+1), & \text{if } u > 0 \text{ and } v = T \\ c(u, v) + p_s(u, v)b(u, v-1) + p_a(u, v)b(u, v+1), & \text{otherwise} \end{cases}$$

$$S_b = \{(u, v) \in S \mid u > 0\}$$

Performance Measures - Blocking time

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

$$b(u, v) = \begin{cases} 0, & \text{if } (u, v) \notin S_b \\ c(u, v) + b(u-1, v), & \text{if } v = N = T \\ c(u, v) + b(u, v-1), & \text{if } v = N \neq T \\ c(u, v) + p_s(u, v)b(u-1, v) + p_a(u, v)b(u, v+1), & \text{if } u > 0 \text{ and } v = T \\ c(u, v) + p_s(u, v)b(u, v-1) + p_a(u, v)b(u, v+1), & \text{otherwise} \end{cases}$$

$$S_b = \{(u, v) \in S \mid u > 0\}$$

$$c(u, v) = \begin{cases} \frac{1}{\min(v, C)\mu}, & \text{if } v = N \\ \frac{1}{\lambda_1 + \min(v, C)\mu}, & \text{otherwise} \end{cases}$$

Performance Measures - Blocking time

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

$$b(u, v) = \begin{cases} 0, & \text{if } (u, v) \notin S_b \\ c(u, v) + b(u-1, v), & \text{if } v = N = T \\ c(u, v) + b(u, v-1), & \text{if } v = N \neq T \\ c(u, v) + p_s(u, v)b(u-1, v) + p_a(u, v)b(u, v+1), & \text{if } u > 0 \text{ and } v = T \\ c(u, v) + p_s(u, v)b(u, v-1) + p_a(u, v)b(u, v+1), & \text{otherwise} \end{cases}$$

$$S_b = \{(u, v) \in S \mid u > 0\}$$

$$c(u, v) = \begin{cases} \frac{1}{\min(v, C)\mu}, & \text{if } v = N \\ \frac{1}{\lambda_1 + \min(v, C)\mu}, & \text{otherwise} \end{cases}$$

$$p_s(u, v) = \frac{\min(v, C)\mu}{\lambda_1 + \min(v, C)\mu}, \quad p_a(u, v) = \frac{\lambda_1}{\lambda_1 + \min(v, C)\mu}$$

Performance Measures - Proportion within time

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

$$P(X^{(1)} < t) = \frac{\sum_{(u,v) \in S_A^{(1)}} P(X_{(u,v)}^{(1)} < t) \pi_{u,v}}{\sum_{(u,v) \in S_A^{(1)}} \pi_{u,v}}$$

$$P(X^{(2)} < t) = \frac{\sum_{(u,v) \in S_A^{(2)}} P(X_{(u,v)}^{(2)} < t) \pi_{u,v}}{\sum_{(u,v) \in S_A^{(2)}} \pi_{u,v}}$$

Performance Measures - Proportion within time

$$P(X^{(i)} < t) = \frac{\sum_{(u,v) \in S_A^{(i)}} P(X_{u,v}^{(i)} < t) \pi_{u,v}}{\sum_{(u,v) \in S_A^{(i)}} \pi_{u,v}}, \quad \text{for } i = \{1, 2\}$$

Performance Measures - Proportion within time

$$P(X^{(i)} < t) = \frac{\sum_{(u,v) \in S_A^{(i)}} P(X_{u,v}^{(i)} < t) \pi_{u,v}}{\sum_{(u,v) \in S_A^{(i)}} \pi_{u,v}}, \quad \text{for } i = \{1, 2\}$$

$$X_{(u,v)}^{(1)} \sim \begin{cases} \text{Erlang}(\nu, \mu), & \text{if } C = 1 \text{ and } \nu > 1 \\ \text{Hypo}([\nu - C, 1], [C\mu, \mu]), & \text{if } C > 1 \text{ and } \nu > C \\ \text{Erlang}(1, \mu), & \text{if } \nu \leq C \end{cases}$$

$$X_{(u,v)}^{(2)} \sim \begin{cases} \text{Erlang}(\min(\nu, T), \mu), & \text{if } C = 1 \text{ and } \nu, T > 1 \\ \text{Hypo}([\min(\nu, T) - C, 1], [C\mu, \mu]), & \text{if } C > 1 \text{ and } \nu, T > C \\ \text{Erlang}(1, \mu), & \text{if } \nu \leq C \text{ or } T \leq C \end{cases}$$

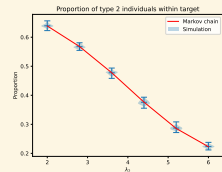
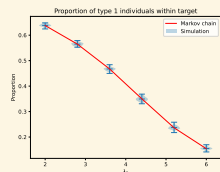
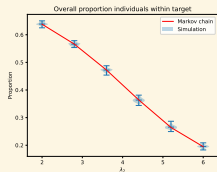
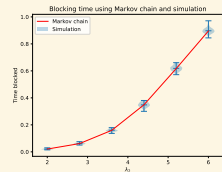
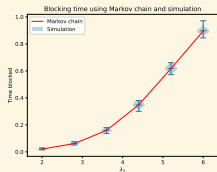
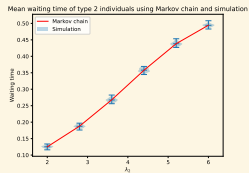
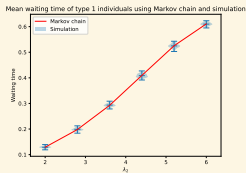
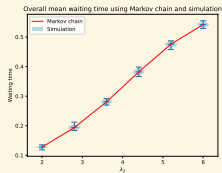
Performance Measures - Proportion within time

$$P(X^{(i)} < t) = \frac{\sum_{(u,v) \in S_A^{(i)}} P(X_{u,v}^{(i)} < t) \pi_{u,v}}{\sum_{(u,v) \in S_A^{(i)}} \pi_{u,v}}, \quad \text{for } i = \{1, 2\}$$

$$P(X_{(u,v)}^{(1)} < t) = \begin{cases} 1 - \sum_{i=0}^{v-1} \frac{1}{i!} e^{-\mu t} (\mu t)^i, & \text{if } C = 1 \text{ and } v > 1 \\ 1 - (\mu C)^{v-C} \mu \sum_{k=1}^{\lceil \vec{r} \rceil} \sum_{l=1}^{r_k} \frac{\Psi_{k,l}(-\lambda_k) t^{r_k-l} e^{-\lambda_k t}}{(r_k-l)!(l-1)!}, & \text{if } C > 1 \text{ and } v > C \\ \quad \text{where } \vec{r} = (v-C, 1) \text{ and } \vec{\lambda} = (C\mu, \mu) & \\ 1 - e^{-\mu t}, & \text{if } v \leq C \end{cases}$$

$$P(X_{(u,v)}^{(2)} < t) = \begin{cases} 1 - \sum_{i=0}^{\min(v,T)-1} \frac{1}{i!} e^{-\mu t} (\mu t)^i, & \text{if } C = 1 \text{ and } v, T > 1 \\ 1 - \mu (\mu C)^{\min(v,T)-C} \times \sum_{k=1}^{\lceil \vec{r} \rceil} \sum_{l=1}^{r_k} \frac{\Psi_{k,l}(-\lambda_k) t^{r_k-l} e^{-\lambda_k t}}{(r_k-l)!(l-1)!}, & \text{if } C > 1 \text{ and } v, T > C \\ \quad \text{where } \vec{r} = (\min(v, T) - C, 1) \text{ and } \vec{\lambda} = (C\mu, \mu) & \\ 1 - e^{-\mu t}, & \text{if } v \leq C \text{ or } T \leq C \end{cases}$$

Comparisons



Game - Definition

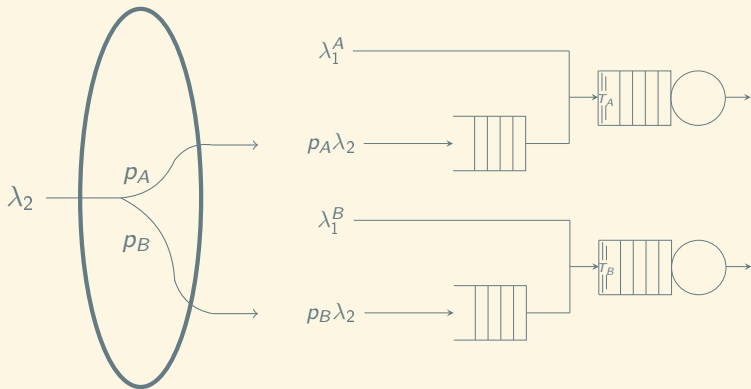


Game - Definition

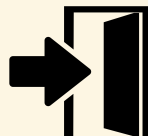
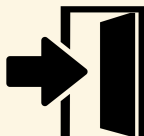


- ▶ Misra S, Sarkar S. Priority-based time-slot allocation in wireless body area networks during medical emergency situations: An evolutionary game-theoretic perspective
- ▶ Song J, Wen J. A non-cooperative game with incomplete information to improve patient hospital choice

Game - Players



Game - Strategies



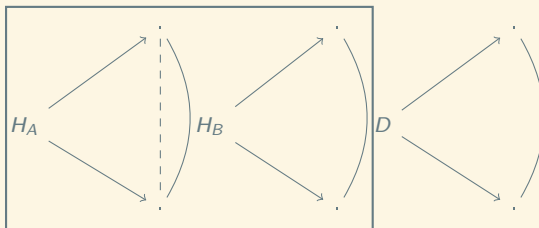
$$p_A, p_B \in [0, 1]$$

$$T_A \in [1, N_A]$$

$$T_B \in [1, N_B]$$

$$p_A + p_B = 1$$

Game - Formulation



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

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

Solution concepts in games

$$A \in \mathbb{R}^{m \times n}, \quad B \in \mathbb{R}^{m \times n}$$

Solution concepts in games

$$A \in \mathbb{R}^{m \times n}, \quad B \in \mathbb{R}^{m \times n}$$

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

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

Solution concepts in games

$$A \in \mathbb{R}^{m \times n}, \quad B \in \mathbb{R}^{m \times n}$$

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

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

- Fudenberg, Drew, et al. The theory of learning in games. Vol. 2. MIT press, 1998.
- Elvio, Accinelli and Carrera, Edgar. 2011. Evolutionarily Stable Strategies and Replicator Dynamics in Asymmetric Two-Population Games. 10.1007/978-3-642-11456-4_3.

Inefficiency measure

$$PoA = \frac{\max_{s \in E} Cost(s)}{\min_{s \in S} Cost(S)}$$

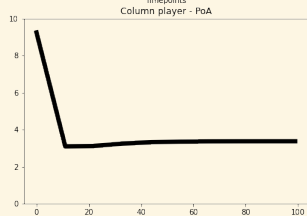
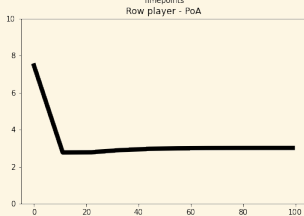
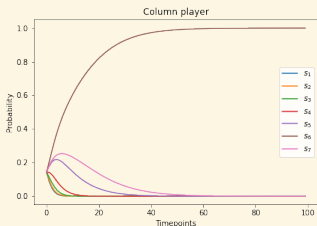
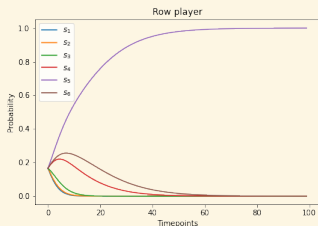
Inefficiency measure

$$PoA = \frac{\max_{s \in E} Cost(s)}{\min_{s \in S} Cost(S)}$$

$$PoA_A(s_r) = \frac{Cost(s_r)}{\min_{s \in S} Cost(S)},$$

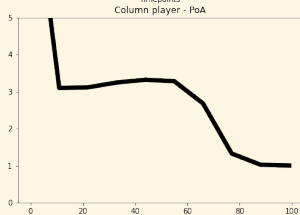
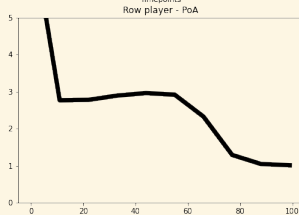
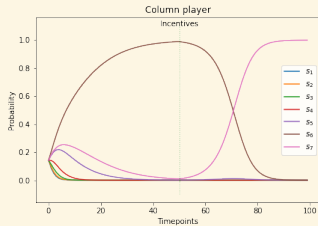
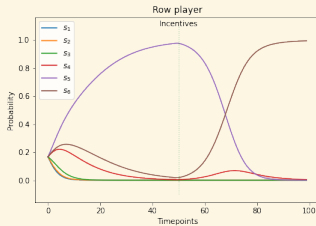
$$PoA_B(s_c) = \frac{Cost(s_c)}{\min_{s \in S} Cost(S)}$$

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”

Thank you!

“Inefficiencies can be learned and emerge naturally”

“Targeted incentivisation of behaviours can help escape learned inefficiencies”

✉ PanayidesM@cardiff.ac.uk

🐦 @Michalis_Pan

🐙 @11michalis11

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