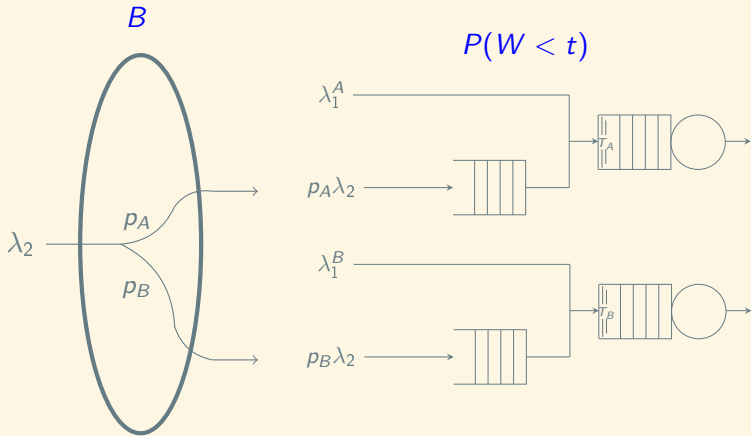
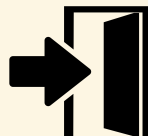
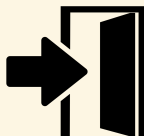


Game - Players and objectives



Game - Strategies



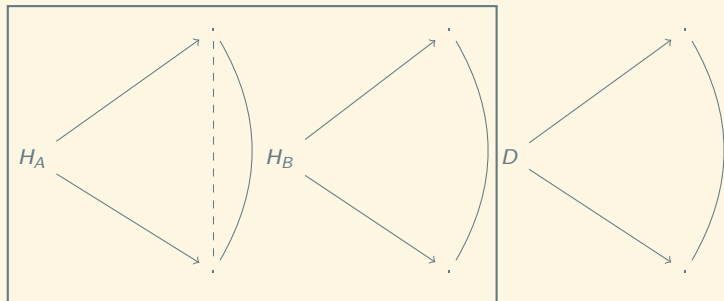
$$p_A, p_B \in [0, 1]$$

$$p_A + p_B = 1$$

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

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

Game - Formulation



Game - Payoff matrices

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

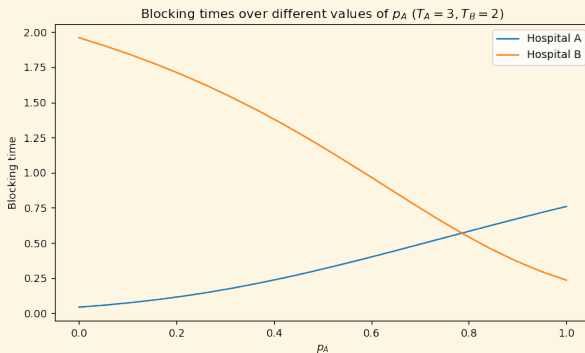
$$R = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N_B} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,N_B} \\ \vdots & \vdots & \ddots & \vdots \\ p_{N_A,1} & p_{N_A,2} & \cdots & p_{N_A,N_B} \end{pmatrix}$$

Ambulance's Decision

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

Ambulance's Decision

$$\begin{matrix} T_A = 3 \\ T_B = 2 \end{matrix} \rightarrow \begin{pmatrix} - & - & - & - \\ - & - & - & - \\ - & \times & - & - \\ - & - & - & - \end{pmatrix} \rightarrow B_A(p_A) = B_B(1 - p_A)$$



Hospitals' Decision

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

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

Nash Equilibrium

$$R = \begin{pmatrix} 0.5 & 0.1 & 0 & 0 \\ 0.9 & 0.5 & 0.2 & 0 \\ 1 & 0.8 & 0.5 & 0.3 \\ 1 & 1 & 0.7 & 0.5 \end{pmatrix}$$

$$A = \begin{pmatrix} 0.99998394 & 0.99998394 & 0.99998394 & 0.99998394 \\ 0.99998955 & 0.99998848 & 0.99998649 & 0.9999845 \\ 0.99999952 & 0.9999987 & 0.99999596 & 0.99999199 \\ 0.99994372 & 0.99995113 & 0.99998603 & 0.99999911 \end{pmatrix}$$

$$B = \begin{pmatrix} 0.99998394 & 0.99998955 & 0.99999952 & 0.99994372 \\ 0.99998394 & 0.99998848 & 0.9999987 & 0.99995113 \\ 0.99998394 & 0.99998649 & 0.99999596 & 0.99998603 \\ 0.99998394 & 0.9999845 & 0.99999199 & 0.99999911 \end{pmatrix}$$

Nash Equilibria: $(0, 0, 0.4, 0.6)$ $(0, 0, 0.4, 0.6)$

Asymmetric Replicator Dynamics

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

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

Inefficiency measure

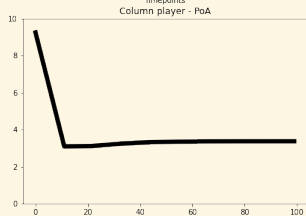
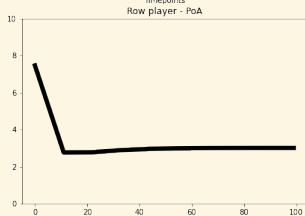
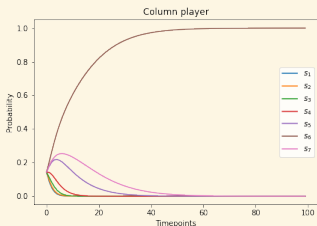
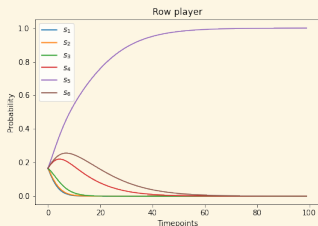
$$PoA = \frac{\max_{s \in E} Cost(s)}{\min_{s \in S} Cost(S)}$$

Inefficiency measure

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

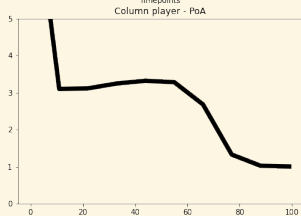
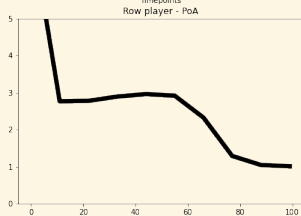
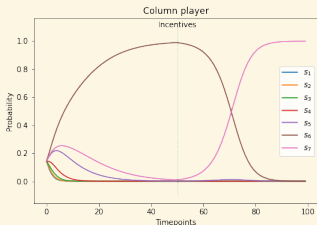
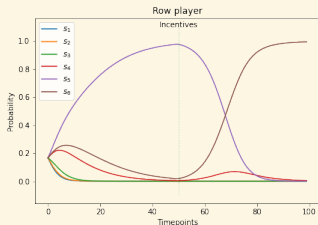
$$PoA_A(s_r) = \frac{Cost(s_r)}{\min_{s \in S} Cost(S)}, \quad PoA_B(s_c) = \frac{Cost(s_c)}{\min_{s \in S} Cost(S)}$$

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”

```
$ pip install ambulance_game  
https://github.com/11michalis11/AmbulanceDecisionGame
```

✉ PanayidesM@cardiff.ac.uk

🐦 @Michalis_Pan

👤 @11michalis11