# Distributed Nash Equilibrium Seeking On a Consensus based Gaming

Ryan J. Kung

ryankung@ieee.org

April 29, 2018

# Introduction

- Game
- Nash Equilibrium
- Distributed Game

# Game

Consider a game with $N$ players. The set of players is denoted by $\mathbb{N} = 1, 2, ..., N$. The payoff function of player $i$ is $f_i(x)$, where $x = [x_1, x_2, ..., x_N]^T \in R^N$ is the vector of players' actions and $x_i \in R$ is the action of player $i$, then player $i$ has no direct access to player $j$'s action.

# Nash Equilibrium

**Nash equilibrium is an action profile on which no player can gain more payoff by unilaterally changing its own action.**

$$\forall i, x_i \in S_i : f_i(x_i^*, x_{-i}^*) \geq f_i(x_i, x_{-i}^*)$$

# Cournot's duopoly Model

- the players are the firms
- the actions of each firm are the set of possible outputs $Q$
- the payoff of each firm is its profit.

where:

$$
\begin{aligned}
\text{p}:= &\quad \text{Price, Inverse demand function} \\
u_i:= &\quad \text{Profit of player i} \\
c_i:= &\quad \text{Total Cost Function}
\end{aligned}
$$

# Cournot's duopoly Model

| | |
|---|---|
| p:= | Price, Inverse demand function |
| $u_i$:= | Profit of player i |
| $c_i$:= | Total Cost Function |

We have:

$$u_i(q_1, q_2) = q_i p(q) - c_i(q_i)$$

Thus the response function $\dot{r}_i : Q_1 \rightarrow Q_2$:

$$\dot{r}_1 = \dot{r}_2 = \frac{1-c}{3}$$

# Distributed

A system is distributed if the message transmission delay is not negligible compared to the time between events in a single process. [1]

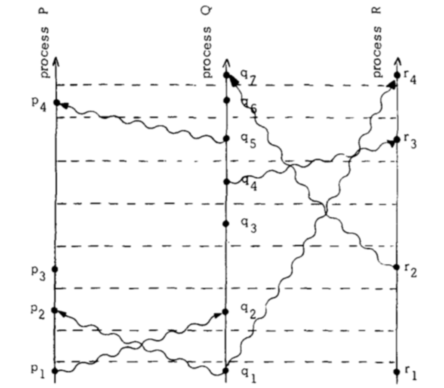



Figure: Lamport Timestamp

# Distributed Game

We defined Distributed Gaming as a series of Game Behaviors and Strategis which is Distributed. A Gaming is Distributed if the message transaction delay is not negligible compared to the time between event in classic gaming behavior.

# Formalize

Consider a game with $N$ players. The set of players is denoted by $\mathbb{N} = 1, 2, ..., N$. The payoff function of player $i$ is $f_i(x)$, where $x = [x_1, x_2, ..., x_N]^T \in R^N$ is the vector of players' actions and $x_i \in R$ is the action of player $i$, then player $i$ has no direct access to player $j$'s action.

**Suppose that if player $j$ is not a neighbor of player $i$, then player $i$ has no direct access to player $j's$ action**

# Distributed Nash Equilibrium

- Sensitive Game
- Distributed Network

# Example A

| | L | R |
|---|---|---|
| U | (1,3) | (-3,0) |
| M | (-2,0) | (1,3) |
| D | (0,1) | (0,1) |

With Repeated Advantage Solution:

| | L |
|---|---|
| U | (1,3) |

But when we discuss Mixed Strategy, if more than 1% $players_2$ chose $R$, $D$ is better than $U$:

# Example A :: Mixed Strategy

Mixed Strategy $\sigma$ is a probability distribution over pure strategy.

A Mixed Strategy for player $i$ can be present as a vector:

$$(\sigma_i(U), \sigma_i(M), \sigma_i(D))$$

Payoff Function $u_i$ over $\sigma$:

$$\sum_{i \in S} \left( \prod_{j=i}^{I} \sigma_i(s_u) \right) u_j(s)$$

# Example A :: Mixed Strategy

| | L | R |
|---|---|---|
| U | (1,3) | (-3,0) |
| M | (-2,0) | (1,3) |
| D | (0,1) | (0,1) |

Let:

$$\sigma_2 = (0.99, 0.01)$$

Then:

$$u_1 = -5.94 \iff \sigma_1 = (1, 0, 0)$$
$$u_1 = 2 \iff \sigma_1 = (0, 0, 1)$$

# Example B

Prisoner's Dilemma

| | M | S |
|---|---|---|
| M | (1,1) | (-1, 2) |
| S | (2,-1) | (0,0) |

If players are sensitive on the uncertainty of $s_{-1}$, they may not choose the rational strategy.

# Distributed Nash Equilibrium

The action of player $i$ is updated according to

$$\dot{x} = k_i \frac{\partial f_i}{\partial x_i}(y_i), i \in \mathbb{N} \tag{0}$$

where

$$\dot{y_i} = [y_{i1}, y_{i2}, ..., y_{iN}]^T \in R$$
$$k_i = \delta \bar{k}$$

$\delta$ as a small positive parameter, $\bar{k_i}$ is a fixed positive parameter. $y_{ij}, \forall i, j \in \mathbb{N}$ is player $i$'s estimate on player $j$'s action, which are generated by:

$$\dot{y_{ij}} = -\left( \sum_{k=1}^{N} a_{ik}(y_{ik} - y_{kj}) + a_{ij}(y_{ij} - x_y) \right)$$

# Distributed Nash Equilibrium

Let $\tau = \delta t$, at the $\tau$-time scale:

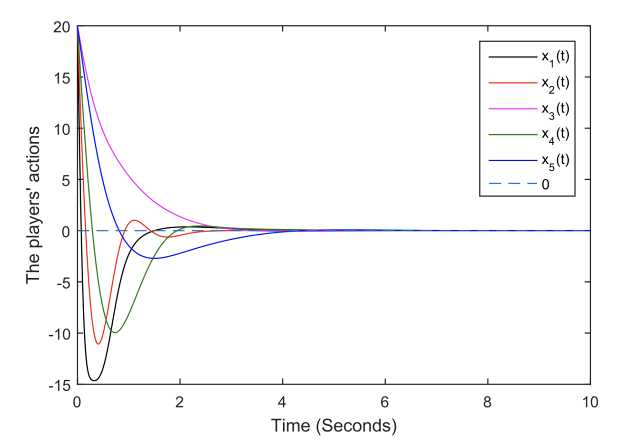
$$\frac{dx_i}{d_\tau} = \bar{k}_i \frac{\partial f_i}{\partial x_i}(y_i)$$

$$\delta \frac{dy_{ij}}{d\tau} = -\left( \sum_{k=1}^{N} a_{ik}(y_{ik} - y_{kj}) + a_{ij}(y_{ij} - x_y) \right) [2] \qquad (1)$$

# Distributed Nash Equilibrium

Easy to know, by setting $\delta$ to zero, The reduced system is:

$$\frac{dx_i}{d\tau} = \bar{k}_i \frac{\partial f_i}{\partial x_i}(x) \quad \forall i \in \mathbb{N} \tag{2}$$



# Under Switching Topologies

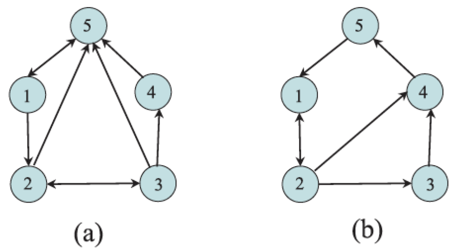
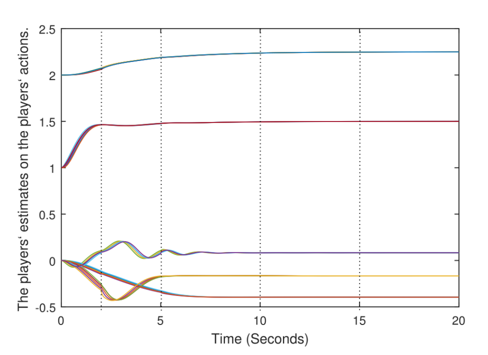



Fig. 1. Communication graph in (a) is $\mathcal{G}^1$ and the communication graph in (b) is $\mathcal{G}^2$.

Player $i$ can be regarded as a virtual leader, who provides its action as a reference signal to be followed by $-i$, For each $X^*$, there is a constant $\delta$ that for every $\delta \in (\delta_{max}, \delta_{min})$, Nash Equilibrium $X^*$ is asymptotically stable. [3]

# On loss of Communication




Player $i$ can be regarded as a virtual leader, who provides its action as a reference signal to be followed by $-i$, For each $X^*$, there is a constant $\delta$ that for every $\delta \in (\delta_{max}, \delta_{min})$, Nash Equilibrium $X^*$ is asymptotically stable. [3]

# Conclusions

- Exists Distributed Nash Equilibrium
- Distributed Nash Equilibrium can be sensitive
- Distributed Nash Equilibrium is stable under different network Topologies

# Reference


Leslie Lamport.
Time, clocks and the ordering of events in a distributed system.
pages 558–565, July 1978.

M. Ye and G. Hu.
Distributed nash equilibrium seeking by a consensus based approach.
IEEE Transactions on Automatic Control, 62(9):4811–4818, Sept 2017.

M. Ye and G. Hu.
Distributed nash equilibrium seeking in multiagent games under switching communication topologies.
IEEE Transactions on Cybernetics, PP(99):1–10, 2017.

# Thanks

Repository of this Slides:

- https://github.com/RyanKung/dneq4eos