DISC Course: Multi-agent Network Dynamics and Games

Homework 4

Hai Zhu
Delft University of Technology
h.zhu@tudelft.nl

April 23, 2018

Problem 1

Solution: a) The replicator dynamics can be written as

$$\dot{x} = \begin{bmatrix} -x_1(ux_3 - x_4 + x_1x_2 + x_1x_4 + x_2x_3 + x_3x_4 - 2ux_1x_3 - 2ux_2x_4) \\ -x_2(ux_4 - x_1 + x_1x_2 + x_1x_4 + x_2x_3 + x_3x_4 - 2ux_1x_3 - 2ux_2x_4) \\ -x_3(ux_1 - x_2 + x_1x_2 + x_1x_4 + x_2x_3 + x_3x_4 - 2ux_1x_3 - 2ux_2x_4) \\ -x_4(ux_2 - x_3 + x_1x_2 + x_1x_4 + x_2x_3 + x_3x_4 - 2ux_1x_3 - 2ux_2x_4) \end{bmatrix} \tag{1}$$

Let $\dot{x} = 0$ and then we can get the equilibrium points of the replicator dynamics:

x_1	x_2	x_3	x_4
$\frac{1}{2}$	0	$\frac{1}{2}$	0
0	$\frac{1}{2}$	0	$\frac{1}{2}$
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
$\frac{\mu}{\mu^2+2\mu-1}$	$\frac{\mu(\mu+1)}{\mu^2+2\mu-1}$	$\frac{-1}{\mu^2+2\mu-1}$	0
$\frac{\mu(\mu+1)}{\mu^2+2\mu-1}$	$\frac{-1}{\mu^2+2\mu-1}$	0	$\frac{\mu}{\mu^2+2\mu-1}$
0	$\frac{\mu}{\mu^2+2\mu-1}$	$\frac{\mu(\mu+1)}{\mu^2+2\mu-1}$	$\frac{-1}{\mu^2+2\mu-1}$
$\frac{-1}{\mu^2+2\mu-1}$	0	$\frac{\mu}{\mu^2+2\mu-1}$	$\frac{\mu(\mu+1)}{\mu^2+2\mu-1}$

b) According to the results of question a), the only interior equilibrium point is $[\frac{1}{4},\ \frac{1}{4},\ \frac{1}{4},\ \frac{1}{4}]$. Thus we can get its Jacobian matrix:

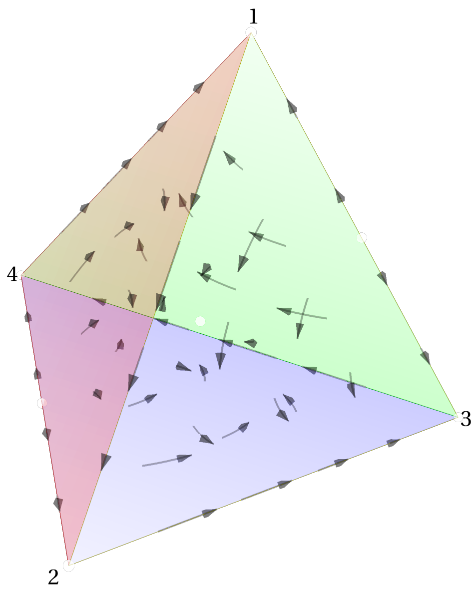
$$J = \begin{bmatrix} \frac{\mu}{8} - \frac{1}{8} & \frac{\mu}{8} - \frac{1}{8} & -\frac{\mu}{8} - \frac{1}{8} & \frac{\mu}{8} + \frac{1}{8} \\ \frac{\mu}{8} + \frac{1}{8} & \frac{\mu}{8} - \frac{1}{8} & \frac{\mu}{8} - \frac{1}{8} & -\frac{\mu}{8} - \frac{1}{8} \\ -\frac{\mu}{8} - \frac{1}{8} & \frac{\mu}{8} + \frac{1}{8} & \frac{\mu}{8} - \frac{1}{8} & \frac{\mu}{8} - \frac{1}{8} \\ \frac{\mu}{8} - \frac{1}{8} & -\frac{\mu}{8} - \frac{1}{8} & \frac{\mu}{8} + \frac{1}{8} & \frac{\mu}{8} - \frac{1}{8} \end{bmatrix} \tag{2}$$

The eigenvalues can be calculated then, which are $\lambda = \frac{\mu}{8} - \frac{1}{8}$ with its algebraic multiplicity 4.

c) To plot the phase portraits of the replicator dynamics, I used a open-source software, Dynamo. Here are three cases:

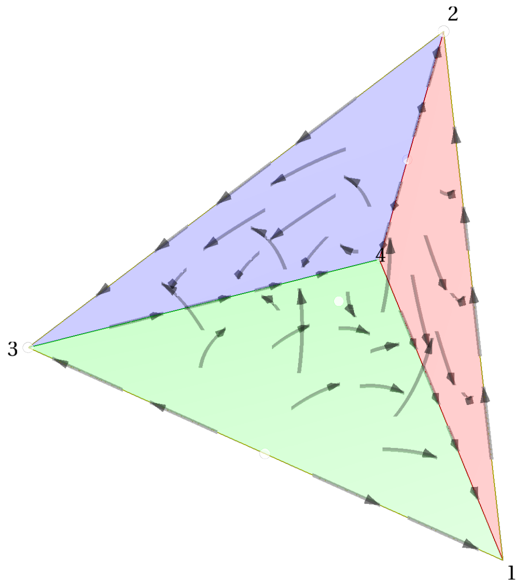
- Case 1: The interior equilibrium is stable.

  Apparently, if $\mu < 1$, then all the eigenvalues are negative and thus the interior equilibrium is stable. The following is the phase portrait when $\mu = 0.5$.



- Case 2: The interior equilibrium is unstable.

  If $\mu < 1$, then all the eigenvalues are positive and thus the interior equilibrium is unstable. The following is the phase portrait when $\mu = 2$.

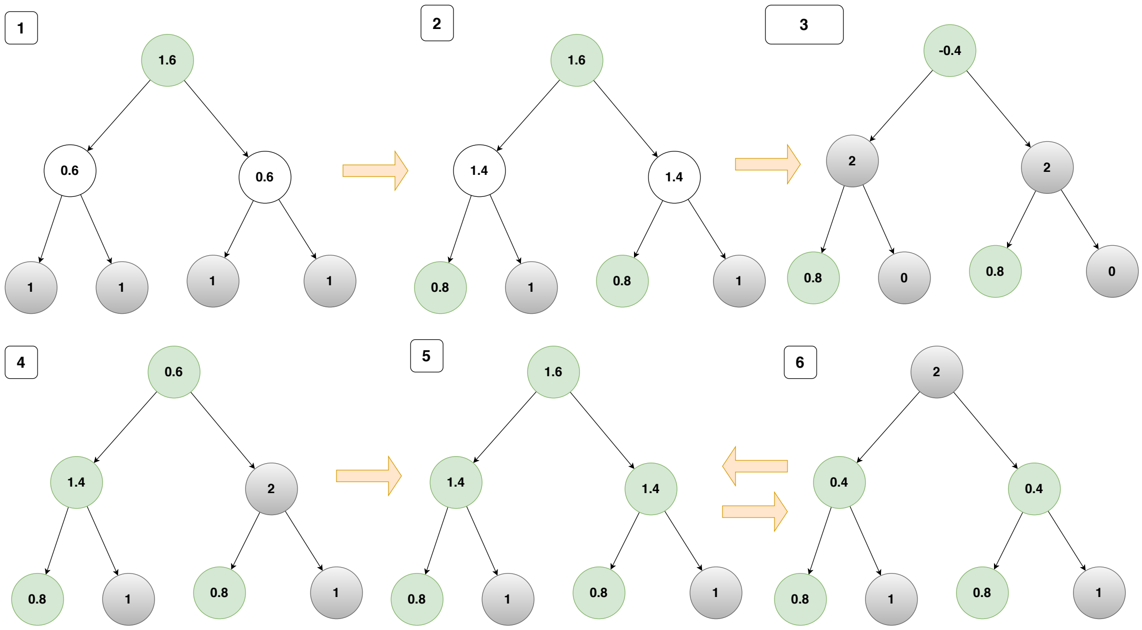


- Case 3: There is a periodic attractor near th interior equilibrium.

Problem 2

Solution: **a)** From the payoff matrix, it can be seen that to defect is the strongly dominant strategy. Thus, it is always a best response to whatever strategy its neighbors choose. Any agent that is not controlled will switch its strategy to defect in the next stage. Hence, under the best response rule, to render the 7 agents to be cooperative asymptotically, all the 7 agents have to be controlled.

b) Riehl and Cao [4] proposed an algorithm to compute the minimum number of controlled agents to drive all the agents to desired strategy under the imitating rule. Here I follow their algorithm as follows:



where the green circles are agents being controlled, the white circles are agents to co-operative (playing strategy A) and the grey circles are agents to defect (playing strategy B).

The steps are:

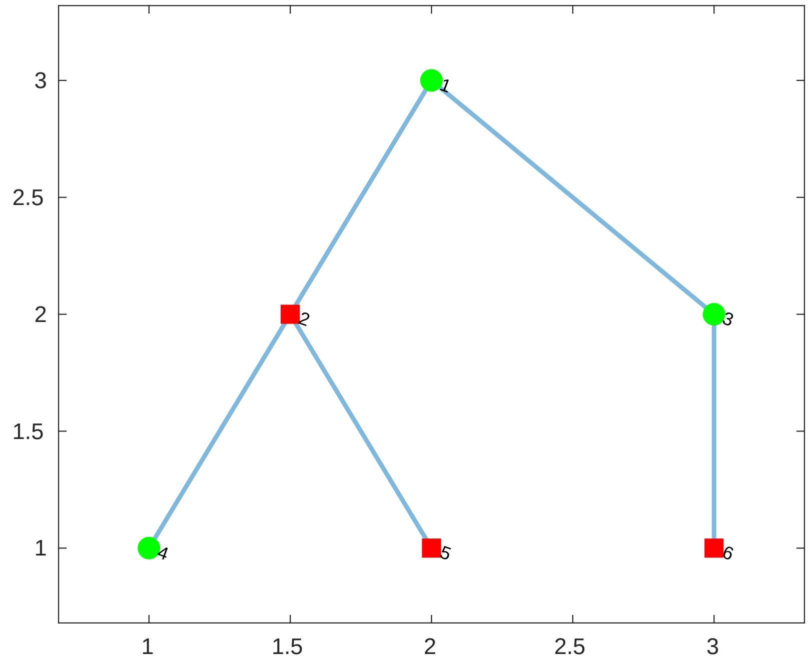
- Assume the root agent is controlled and all other agents are playing A.
- From the bottom level, assuming each agent plays B and compute their payoff and their parents' payoff.
- If any child's payoff is higher than its parents' payoff. Determine the minimum number and who should be controlled such that the parents' payoff is higher than the children. Repeat this for every sub-branch.
- Move to one upper level (second level in this case). Do the previous step again to determine the agents who should be controlled in this level.
- Repeat the previous two steps level by level until the root.
- Remove the root from the controlled agents set and check if the network can be driven to desired state. In this case the answer is not. So the root has to stay in the controlled agents set.

In conclude, by applying the algorithm, we can obtain that the minimum number of controlled agents is 6 and they are shown in the above figure.

Problem 3

Solution: To find the presented examples, I used Matlab to search and simulate the evolutionary process of networks, which is listed at the end of this solution. In all examples, a graph is given to represent a network, in which the green circular nodes are coordinating agents and the red rectangular nodes are anti-coordinating agents. All agents have two strategies to play, A and B and update their strategies synchronously.

a) Yes. Here is an example:

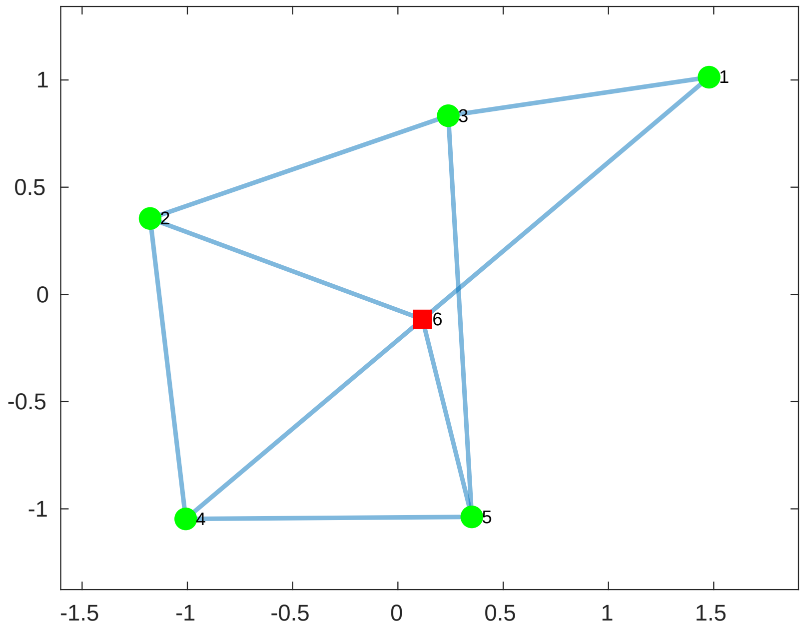


where 1,3,4 are coordinating agents and 2,5,6 are anti-coordinating agents and their initial playing strategies are $\{A, A, A, B, B, B\}$. The threshold is set to be $\tau_i = 0.5, i = 1, \dots, 6$. For all agents, if $n_i^A(k) = \tau_i \text{deg}_i$, they keep their current strategy, i.e. $z_i = x_i(k)$. According to the simulation, we can obtain the evolution of this network:

$$
\begin{aligned}
&\{A, A, A, B, B, B\} \\
\rightarrow &\{A, A, A, A, B, B\} \\
\rightarrow &\{A, B, A, A, B, B\} \\
\rightarrow &\{A, B, A, B, A, B\} \\
\rightarrow &\{A, B, A, B, A, B\}
\end{aligned}
$$

Finally it converge to an equilibrium $\{A, B, A, B, A, B\}$.

b) Yes. Here is an example:

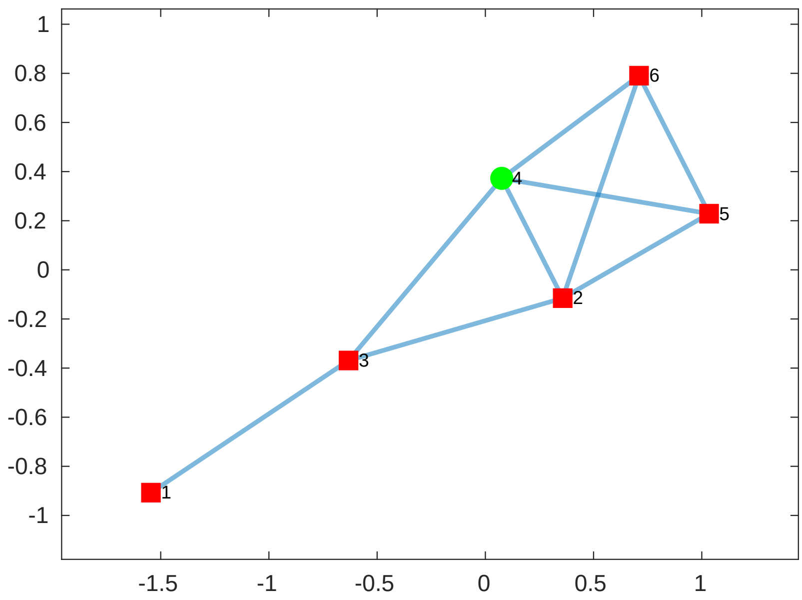


where 1,2,3,4,5 are coordinating agents and 6 is an anti-coordinating agents and their initial playing strategies are $\{A, A, A, B, B, B\}$. The threshold is set to be $\tau = [0.3848, 0.1626, 0.7968, 0.1138, 0.1588, 0.3558]^T$. For all agents, if $n_i^A(k) = \tau_i \text{deg}_i$, they keep their current strategy, i.e. $z_i = x_i(k)$. According to the simulation, we can obtain the evolution of this network:

$$
\begin{aligned}
&\{A, A, A, B, B, B\}\\
\rightarrow&\{A, A, B, A, A, B\}\\
\rightarrow&\{B, A, A, A, A, B\}\\
\rightarrow&\{A, A, B, A, A, B\}\\
\rightarrow&\{B, A, A, A, A, B\}
\end{aligned}
$$

Finally it converges to a periodic solution of period 2: $\{A, A, B, A, A, B\} \rightarrow \{B, A, A, A, A, B\}$.

c) Yes. Here is an example:



where 4 is a coordinating agents and 1,2,3,5,6 are anti-coordinating agents and their initial playing strategies are $\{A, A, A, B, B, B\}$. The threshold is set to be $\tau_i = 0.8532, i = 1, \dots, 6$. For all agents, if $n_i^A(k) = \tau_i \text{deg}_i$, they keep their current strategy, i.e. $z_i = x_i(k)$. According to the simulation, we can obtain the evolution of this network:

$$
\begin{aligned}
&\{A, A, A, B, B, B\} \\
\to&\{B, A, A, B, A, A\} \\
\to&\{B, A, A, A, A, A\} \\
\to&\{B, B, A, A, B, B\} \\
\to&\{B, A, A, B, A, A\} \\
\to&\{B, A, A, A, A, A\} \\
\to&\{B, B, A, A, B, B\}
\end{aligned}
$$

Finally it converges to a periodic solution of period 2: $\{B, A, A, B, A, A\} \to \{B, A, A, A, A, A\} \to \{B, B, A, A, B, B\}$.

d) No. First, I have searched for periodic solution of period 2, 3, 4, 5 and 6 and I can always find an example. This indicates that it is almost surely converges to an equilibrium or a periodic solution. However, this does not mean that there is no case in which it will not converge. Hence, a rigorous proof is needed to prove that it will always converge. Nevertheless, I have no idea how to give such a rigorous proof. But I can show the property in a naive way.

Given a network, denote $x(k) = \{x_i(k), i = 1, \dots, 6\}$ the state vector of the network at stage k and $x(k) \in \mathcal{X}$, where $\mathcal{X}$ is the feasible set of x. Then it is apparent that $\mathcal{X}$ is finite. Thus, for any $x(k)$, we can always find such a stage in the future who has the same state as current stage, i.e.

$$\exists \lambda \in \mathbb{N}, \quad s.t. \;\; x(k + \lambda) = x(k)$$

Thus, under the linear threshold model, the network will always converge to an equilibrium or a periodic solution with finite period.

Matlab Code for Simulation

```
clear all
close
clc

%% W4P3_sim_random
Num = 6;                        % number of agents
Gen = 100;                      % number of simulation generation
Age_his = zeros(Gen,Num);       % history of playing strategies
Try_num = 1000;                 % number of try to find a solution with period
period = 6;                     % to find a perodic solution
period_max = 6;
flag_stop = 0;                  % if stop the serching
more = 2*period;                % evolution more generations after finding a solution
ind_more = 0;                   % index

%% finding an example
for k = 1 : Try_num
    % threshold and coordination spercifying
    tau = rand(Num,1);          % threshold for each agents
%     tau = rand(1)*ones(Num,1);
    map = randi([0 1],Num,1);   % coordinating (1) or anti-coordinating (0)
    % creating the graph
    G_A = randi([0 1],Num,Num);
    G_A = G_A - tril(G_A,-1) + triu(G_A,1)';
    G_A = G_A - diag(diag(G_A));
    G_names = {'1' '2' '3' '4' '5' '6'};
    G = graph(G_A,G_names);
    %% simulation
%     Age_0 = randi([0 1],Num,1); % ininital playing strategies
                                % agents with its playing strategy,
                                % 1 indicates A, 0 indicates B
    Age_0 = [1;1;1;0;0;0];
    Age_pre = Age_0;            % initialization
    Age_next = Age_0;
    for i = 1 : Gen
        for j = 1: Num          % loop for each agent
            % calculate the number of playing A of its neighbors
            neighbor = neighbors(G,j);          % neighbors of the agent
```

```
            num_A = sum(Age_pre(neighbor));     % number of neighbors playing A
            thr_nei = tau(j)*length(neighbor);  % threshold
            % updating the strategy of the agent
            if map(j) == 1                      % for coordinating agents
                if num_A == thr_nei
                    Age_next(j) = Age_pre(j);
    %                 Age_next(j) = 1;
                else
                    Age_next(j) = (num_A > thr_nei);
                end
            end
            if map(j) == 0                      % for anti-coordinating agents
                if num_A == thr_nei
                    Age_next(j) = Age_pre(j);
    %                 Age_next(j) = 0;
                else
                    Age_next(j) = (num_A < thr_nei);
                end
            end
        end
        Age_his(i,:) = Age_next';
        if period == 1 && isequal(Age_his(i,:), Age_his(i-1,:))
            flag_stop = 1;
        elseif (i > period) && (isequal(Age_his(i,:), Age_his(i-period,:))) && (~isequal(Age_his(i,:), Age_his(i-1,:)))
            flag_stop = 1;
        end
        if flag_stop == 1 && ind_more > more
            break;
        end
        Age_pre = Age_next;
        ind_more = ind_more + 1;
    end
    if flag_stop == 1 && ind_more > more
        break;
    end
    Age_his = zeros(Gen,Num);
    ind_more = 0;
end

if flag_stop == 1
    h_G = plot(G,'MarkerSize',9,'Linewidth',2);
    highlight(h_G,find(map),'NodeColor','g');
```

```
    highlight(h_G,find(~map),'NodeColor','r','Marker','s');
end
```

References

[1] M. Maschler, E. Solan, and S. Zamir. *Game Theory.* Cambridge: Cambridge University Press, 2013.

[2] Weibull, Jörgen W. *Evolutionary Game Theory*. MIT Press, 1997.

[3] W. H. Sandholm, E. Dokumaci, and F. Franchetti. *Dynamo: Diagrams for Evolutionary Game Dynamics*. 2012. http://www.ssc.wisc.edu/ whs/dynamo.

[4] Riehl, J. R., and Cao, M. Minimal-agent control of evolutionary games on tree networks. In The 21st International Symposium on Mathematical Theory of Networks and Systems (Vol. 148), 2014.

[5] H.K. Khalil. *Nonlinear systems*. Prentice Hall, Upper Saddle River, USA, third edition, 2002.