Modular Decomposition of Complex Interactions using Games Networks

Matthieu Manceny
IBISC Laboratory
523 Place des Terrasses de l'Agora
91000 EVRY, FRANCE
matthieu.manceny@ibisc.univ-evry.fr

Franck Delaplace
IBISC Laboratory
523 Place des Terrasses de l'Agora
91000 EVRY, FRANCE
franck.delaplace@ibisc.univ-evry.fr

ABSTRACT

In this paper, we describe a *modular* game theoretical framework: the theory of *games networks*. Games networks extend non cooperative game theory by allowing agents to participate to several games simultaneously, what make possible the description of local interactions between agents. The theory enables us to formulate global interaction behaviors as composition of local interactions. This puts the emphasis on the way to decompose a game (viewed as a global structure of interactions) into a network of smaller games (viewed as local structures of interactions). The question of decomposition is significant for the understanding of complex systems whose dynamics is based on interactions between agents, such as biological networks. We describe an algorithm for this decomposition which modifies the network structure — how agents are connected to games — while preserving its dynamics — identified by games network equilibria (**Gne**) which extend the notion of Nash equilibria to games networks. Games within the decomposed network represent *basic building blocks* whose interactions may explain how the system works.

General Terms

Algorithms, Theory

Keywords

complex systems, modularity, locality, game theory, networks

1. INTRODUCTION

Game theory provides a framework to model and characterize complex interplays in a large variety of fields such as Economics ([6, 9]), Computer Science ([1, 15]) or Biology ([8, 13]). However, usual representation of game theory obscures locality of interactions because each player is assumed to play with all the others.

Our framework extends strategic game theory to represent interactions as a "network of games and players" where players are connected to the games they participate to. Games networks aim to analyse dynamical aspects of interactions and to model them as a set of modular activities. In a game networks, each game represents a module of interactions.

Finding modules may help us to understand the behavior of a system by decomposing it into basic building blocks. For instance, in post-genomic analysis ([2, 16]) modular analysis of interactions between molecular agents help to find the association between a support (a set of interacting agents) and a biological function. Modules have the following general properties:

- *Generative*: each module is constitutive of a system of which it defines a building block. From the assembly of the modules, the system is formed and acquires its properties.
- *Elementary*: this property refers to the atomicity of a module, i.e. the impossibility to extract a sub-module from a module.

Games networks theory provides a framework analysis of modules based on dynamics. It describes complexity of interplays by games which are assimilated to modules. Modular dynamics relies on locality assumptions (represented by games). From the local properties of games, e.g. local equilibria, we compute global properties, e.g. global equilibria computed by a "composition" of compatible local equilibria. Games networks have been used to model biological complex systems. In [2], authors deals with the Plasminogen Activation system (PAs). PAs is a process of signal transduction implied in the migration of cancer cells. Two global equilibria were found. The first one corresponds a non-migratory state for the cell; the second one corresponds to a pro-migratory state.

However the description may not represent a module because the initial description may not be necessary elementary. Indeed, elementary modules relies on the assumption that a module (a game) cannot be splited into sub-modules (sub-game). Hence, we propose an algorithm to automatically decompose a game into elementary modules. The automatic decomposition emphasizes new games structure of the former network and reveals real interactions between players. The algorithm is based on a dependence analysis between agents. If an agent is dependent to another one then it must consider the decision-making of the others to compute its payoff.

The paper is structured as follows. We present related works in section 2. The key notions of game theory and its extension, games networks theory, are presented in sections 3 and 4. In section 5 we developped the definition of operators which allow us to modify the structure of a game network. In section 6 we present the notion of dependence, and describe an algorithm which decompose a game in its elementary modules. We conclude in section 7.

GameNets '06, October 14, 2006, Pisa, Italy.

2. RELATED WORK

Research of steady states of a game, and so computation of Nash equilibria, is certainly one of the most studied field in game theory. McKelvey and McLennan note that the computation of Nash equilibria in n-players games is much harder, in many important ways, that the computation in 2-players games [10]. In order to reduce the complexity of Nash equilibria computation, some authors have investigated "*games with local interactions*" where games are no longer considered in their globality, but through the local interactions between agents. Moreover, games with local interactions allow to describe systems in a modular way and to study influence of local modifications on the global behavior of the system.

To express locality in games, authors usually considered the notion of *dependence* either between actions [7, 5] or agents [4].

- La Mura, in [7] introduces a new game representation, more structured and more compact than classical representations in game theory. Considering the strategic separabilities in its representation, La Mura presents convergence methods to compute Nash equilibria.

- Koller and Milch in [5] propose a representation language for general multi-player games named Multi-Agent Influence Diagrams. They insist on the importance of dependence relationship among variables to detect structures in games and decrease the computational cost of finding Nash equilibria.

- Kearns, Littman and Singh in [4] introduce a compact graph-theoretic representation for multi-party game theory. Their main result is an efficient algorithm for computing approximate Nash equilibria in one-stage games represented by trees or sparse graphs.

Some authors are interested in *spatial locality* of agents. Informally, in such games, an agent can only interact with its neighbors. See [12] for a survey of these "spatial games".

In this paper, we focus on interactions localized to a given process. Our games network representation, compared to La Mura's one, is not another game-theoretic representation but an extension of strategic representation. The closest representation is that of Kearns, Littman and Singh. However, in quite a some way as Koller and Milch, we are interested in the influence of the network organization, in terms of dependences between agents. We more particularly focus on the research of elementary modules which compose a game.

3. STRATEGIC GAMES

In this section we give definitions of game theory used in the article. The reader may refer to the books [14] or [11] for a complete overview of game theory and its applications.

3.1 Definition of a strategic game

Strategic game is a model of interplays where each agent chooses its plan of action (or strategy) once and for all, and these choices are made simultaneously. Moreover, each agent is rational and perfectly informed of the payoff functions of other agents. Thus, agents aim at maximizing their payoff while knowing the expectation of other agents.

Definition 1. Normal or strategic representation. A strategic game G is a $3-$tuple $\langle A, C, u\rangle$ where:

- A is a set of players or agents.

- $C = \{C_i\}_{i\in A}$ is a set of strategy sets; $C_i = \{c_i^1, ..., c_i^{m_i}\}$ represents the set of the m_i strategies available for agent i.

- $u = (u_i)_{i\in A} : \prod_{i\in A} C_i \to \mathbb{R}^{|A|}$ is a payoff function; $u_i : \prod_{i\in A} C_i \to \mathbb{R}$ associates a payoff for agent i to each game configuration $c = (c_i)_{i\in A} \in \prod_{i\in A} C_i$.

3.2 Mixed (or randomized) strategies

Given a strategic game $G = \langle A, C, u\rangle$, a *mixed-strategy* for any player i is a probability distribution over C_i. We let $\Delta(C_i)$ denote the set of all possible mixed strategies for player i:

$$\Delta(C_i) = \{(p_j)_{j\in[1:m_i]} \,|\forall j \in [1:m_i] \quad 0 \le p_j \le 1 \wedge \sum_{j=1}^{m_i} p_j = 1\}$$

A *mixed-strategy configuration* σ is any vector that specifies one mixed strategy $\sigma_i \in \Delta(C_i)$ for each agent $i \in A$. We let $\Delta(C)$ denotes the set of all possible mixed-strategy configuration:

$$\Delta(C) = \prod_{i\in A} \Delta(C_i)$$

For any mixed-strategy configuration $\sigma \in \Delta(C)$, let $u_i(\sigma)$ denotes the payoff for player $i \in A$:

$$u_i(\sigma) = \sum_{c\in\prod_{i\in A} C_i} (\prod_{j\in A} \sigma_j(c_j)) u_i(c)$$

where $\sigma_j(c_j)$ represents the probability that agent j plays c_j.

3.3 Nash equilibrium

Nash equilibrium is a central concept of game theory. This notion captures the steady states of the play of a strategic game in which each agent holds the rational expectation about the other players behavior.

Definition 2. Nash equilibrium of a strategic game. Let $G = \langle A, C, u\rangle$ be a strategic game. We denote by **Nash**(G) the set of all Nash equilibria for G:

$$\mathbf{Nash}(G) = \{\sigma^* \in \Delta(C) \,|\forall i \in A,\ \forall \sigma_i \in \Delta(C_i) \quad u_i(\sigma^*_{-i}, \sigma_i) \le u_i(\sigma^*_{-i}, \sigma^*_i)\}$$

with $(\sigma^*_{-i}, \sigma_i)$ equivalent to σ^* but where player i plays its strategy σ_i rather than σ^*_i.

In other words, *no agent can unilaterally deviate of a Nash equilibrium without decreasing its payoff.*

4. GAMES NETWORK

In this section, we address the main definitions of a games network. Games networks correspond to an extension of game theory which defines *modular interactions* localized to different subsets of agents. Each module corresponds to a specific game defined by a payoff function. Parameters of the payoff function are strategies of agents involved in the game. Agents are shared between different modules and played different games in parallel. However, they have the same set of strategies for every games they played. Moreover, we assume that agents follow the *single played strategy* rule: an agent plays the same strategy for each game it is connected to. The reader may refer to [3] for a more complete overview.

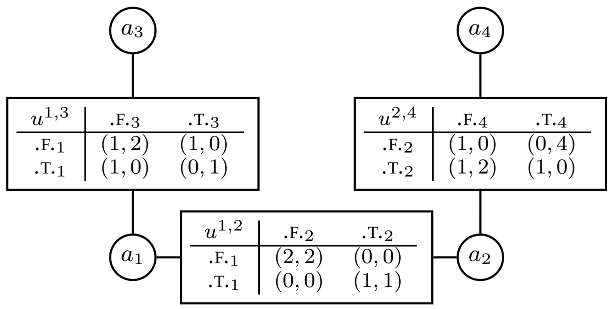


Figure 1: A "4-agents-3-games" games networks

4.1 Definition of a games network

Definition of a games network mainly consists of defining a set of agents connected to a set of games.

Definition 3. Games network. A games network is a $3-$tuple $\langle \mathcal{A}, C, \mathcal{U} \rangle$ where

- $\mathcal{A}$ is a set of agents or players.
- $C = \{C_i\}_{i \in \mathcal{A}}$ is a set of strategy sets.
- $\mathcal{U} = \{\langle A_j, u^j \rangle\}_j$ is a set of *game nodes* where each $A_j \subseteq \mathcal{A}$ is a set of agents and $u^j = (u_i^j : \Pi_{i \in A_j} C_i \mapsto \mathbb{R})_{i \in A}$ is a vector of payoff functions.

A games network offers a synthetic representation to define the different interplays between several players. The structure $\langle A, u \rangle$ totally determines a game played by a subset of agents since it is useless to include the strategies which are perfectly defined in C.

A games network is represented by a bipartite graph $\langle \mathcal{A}, \mathcal{U}, E \rangle$, with $E \subseteq \mathcal{A} \times \mathcal{U}$ and where an edge $(i, \langle A, u \rangle)$ is a member of E if and only if $i \in A$ (See fig. 1 for an illustration of a "4-agents-3-games" games network).

4.2 Games network equilibrium

Games networks allow a two level description of complex systems: local interactions are combined and result in a global behavior. Local steady states are identified to *local equilibria*, i.e. Nash equilibria of game nodes.

Global equilibria are equilibria at the scale of the whole games network. Such equilibria are called *games network equilibria*, or **Gne**. A **Gne** corresponds to a compatible association of local equilibria:

*Definition 4. Games network equilibrium (***Gne***).* Considering $\Gamma = \langle \mathcal{A}, C, \mathcal{U} \rangle$ a games network, we denote by **Gne**(Γ) the set of all global equilibria for Γ:

$$\textbf{Gne}(\Gamma) = \{\sigma^* \in \Delta(C) \mid \forall \langle A_j, u^j \rangle \in \mathcal{U} \quad p(\sigma^*, A_j) \in \textbf{Nash}(\langle A_j, \{C_i\}_{i \in A_j}, u^j \rangle)\}$$

with $p(\sigma^*, A_j) = (\sigma_i^*)_{i \in A_j}$, the projection of σ^* to a subset $A_j \subseteq \mathcal{A}$ of agents.

Example 1. A "4-agents-3-games" games network.
Let us consider $\Gamma = \langle \mathcal{A}, C, \mathcal{U} \rangle$ the games network of fig.1. We have:

- $\mathcal{A} = \{a_1, a_2, a_3, a_4\}$, the agents
- $\forall i \in \mathcal{A} \quad C_i = \{.\text{F}._i, .\text{T}._i\}$, the agents' strategies
- $\mathcal{U} = \{\langle A_{1,3}, u^{1,3} \rangle, \langle A_{1,2}, u^{1,2} \rangle, \langle A_{2,4}, u^{2,4} \rangle\}$ which represents the game nodes. $A_{1,3} = \{a_1, a_3\}$, $A_{1,2} = \{a_1, a_2\}$, $A_{2,4} = \{a_2, a_4\}$ and the payoff functions are shown in fig.1.

To compute the **Gne** of Γ, let us compute the **Nash** of each game node.

$$\textbf{Nash}_{1,3} = \textbf{Nash}(\langle A_{1,3}, \{C_1, C_3\}, u^{1,3} \rangle) = \Big\{ \big((1,0),(1,0)\big) \ ; \ \big((\frac{1}{3}, \frac{2}{3}),(1,0)\big) \Big\}$$

$$\textbf{Nash}_{2,4} = \textbf{Nash}(\langle A_{2,4}, \{C_2, C_4\}, u^{2,4} \rangle) = \Big\{ \big((0,1),(1,0)\big) \ ; \ \big((\frac{1}{3}, \frac{2}{3}),(1,0)\big) \Big\}$$

$$\textbf{Nash}_{1,2} = \textbf{Nash}(\langle A_{1,2}, \{C_1, C_2\}, u^{1,2} \rangle) = \Big\{ \big((1,0),(1,0)\big) \ ; \ \big((0,1),(0,1)\big) \ ; \big((\frac{1}{3}, \frac{2}{3}),(\frac{1}{3}, \frac{2}{3})\big) \Big\}$$

Thus, we can compute the **Gne** of Γ by combining Nash equilibria of each game node:

$$\textbf{Gne}(\Gamma) = \Big\{ \big((\frac{1}{3}, \frac{2}{3}),(\frac{1}{3}, \frac{2}{3}),(1,0),(1,0)\big) \Big\}$$

Note that, in certain situations, it may be possible that no **Gne** exists.

5. STRUCTURAL MODIFICATIONS AND GAMES NETWORKS EQUIVALENCE

Games networks allow the combination of several games into a single network. This puts the emphasis on the *network structure*, i.e. how agents are connected to games, because different structures can be proposed to model the same situation. To compare these different structures we first define the notion of equivalence between games network (section 5.1). Then we detail operators to modify the structure (section 5.2). The use of structural operators reveals the importance of a specific function called observer. We deal with this notion in section 5.3.

5.1 Equivalence between games networks

The notion of equivalence is used to determine whether two network structures model the same situation. Informally, the network structure represents the statics of the networks, whereas identical situation have the same dynamics. We consider that global equilibria of a games network represent its steady states. Thus the equivalence between two stuctures is based on the equality of their equilibria:

Definition 5. **Gne** *equivalence.* Let $\Gamma_1 = \langle \mathcal{A}_1, C_1, \mathcal{U}_1 \rangle$, $\Gamma_2 = \langle \mathcal{A}_2, C_2, \mathcal{U}_2 \rangle$ be two games networks such that $\mathcal{A}_1 = \mathcal{A}_2, C_1 = C_2$. We denote by $\Gamma_1 \equiv_{\textbf{Gne}} \Gamma_2$ the equivalence between Γ_1 and Γ_2:

$$\Gamma_1 \equiv_{\textbf{Gne}} \Gamma_2 \Leftrightarrow \textbf{Gne}(\Gamma_1) = \textbf{Gne}(\Gamma_2)$$

5.2 Operators for structural modifications

Operators detailled here allow us to modify the structure of a games network. Indeed, restructuring games networks is expressed in terms of substituting game nodes. The join operation or, conversely, the separation are the basic operations for games networks reorganization. Operators for structural modifications can be applied either on games or games nodes.

5.2.1 Substitution

The *substitution operator* consists in replacing a set of game nodes by another:

Definition 6. Substitution. Let $\Gamma = \langle \mathcal{A}, C, \mathcal{U} \rangle$ be a games network, let $U \subseteq \mathcal{U}$ be a set of game nodes. Let $U' = \{\langle A_k, u^k \rangle\}_k$ such that $A_k \subseteq \mathcal{A}, \forall k$. We denote by $\Gamma_{[U/U']}$ the substitution of U by U' in Γ:

$$\Gamma_{[U/U']} = \langle \mathcal{A}, C, \mathcal{U} - U \cup U' \rangle$$

5.2.2 Join operation

The *join operation* consists in joining two games (or game nodes) in a single one:

Definition 7. Join according to ω. Let $G_1 = \langle A_1, C_1, u^1 \rangle$ and $G_2 = \langle A_2, C_2, u^2 \rangle$ be two games, let $\omega : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ be a function. We denote by $G_1 \bigvee^{\omega} G_2$ the join game of G_1 and G_2:

$$G_1 \bigvee^{\omega} G_2 = \langle A_1 \cup A_2, C_1 \cup C_2, u \rangle$$

with

$$\begin{aligned}
&\forall c \in C_{(A_1 \cup A_2)} \\
&\quad \forall i \in A_1 - A_2 \quad u_i(c) = {u^1}_i\big(p(c, A_1)\big) \\
&\quad \forall i \in A_2 - A_1 \quad u_i(c) = {u^2}_i\big(p(c, A_2)\big) \\
&\quad \forall i \in A_1 \cap A_2 \quad u_i(c) = \omega\Big({u^1}_i\big(p(c, A_1)\big), {u^2}_i\big(p(c, A_2)\big)\Big)
\end{aligned}$$

The join operation depends on a function ω, called *observer function*. For instance, the maximum function $(x, y) \mapsto \max(x, y)$ can be a candidate for giving a concrete definition of $\bigvee^{\omega}$ operation. This observer function explicits how an agent computes its global payoff considering local payoffs from the different games it participates to. Section 5.3 exemplifies the importance of the observer function.

5.2.3 Separation

Separation is the reciprocal operation of the join operation. It consists in spliting a game in a games network with two game nodes. The separation, according to a function ω, is defined as follows:

Definition 8. Separation according to ω. Let G be a game and Γ be a games network. We denote by $\bigwedge^{\omega} G = \Gamma$ the separation of G (according to ω) into Γ:

$$\bigwedge^{\omega} \langle A, C, u \rangle = \langle A, C, \{\langle A_1, u^1 \rangle, \langle A_2, u^2 \rangle\}\rangle \Leftrightarrow \langle A_1, u^1 \rangle \bigvee^{\omega} \langle A_2, u^2 \rangle = \langle A, u \rangle$$

By extension, we also called separation the games network which results of successive separations.

5.3 Structural modifications and importance of the observer

The structural modifications are dependant on a particular function ω, called *observer* and used in join or separation. Considering a games networks, the resulting joint game can be very different according to the observer we use. For example, let us consider the games network Γ from fig. 2 which is composed of two game nodes $g^{1,2}$ and $g^{1,3}$. If we use the *Max* function as the observer to join $g^{1,2}$ and $g^{1,3}$, the resulting game is $\Gamma^M = g^{1,2} \bigvee^{Max} g^{1,3}$. The payoff function of Γ^M is shown in the following table.

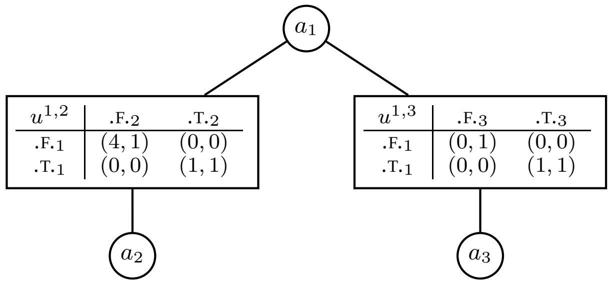


Figure 2: Γ, a small games network to instance the importance of the observer function

a_1	a_2	a_3	u_1^M	u_2^M	u_3^M
.F.1	.F.2	.F.3	4	1	1
.F.1	.F.2	.T.3	4	1	0
.F.1	.T.2	.F.3	0	0	1
.F.1	.T.2	.T.3	0	0	0
.T.1	.F.2	.F.3	0	0	0
.T.1	.F.2	.T.3	0	0	1
.T.1	.T.2	.F.3	1	1	0
.T.1	.T.2	.T.3	1	1	1

If we use the *min* function as observer to join $g^{1,2}$ and $g^{1,3}$, we obtain Γ^m as result game: $\Gamma^m = g^{1,2} \bigvee^{min} g^{1,3}$. The payoff function of Γ^m is shown in the following table.

a_1	a_2	a_3	u_1^m	u_2^m	u_3^m
.F.1	.F.2	.F.3	0	1	1
.F.1	.F.2	.T.3	0	1	0
.F.1	.T.2	.F.3	0	0	1
.F.1	.T.2	.T.3	0	0	0
.T.1	.F.2	.F.3	0	0	0
.T.1	.F.2	.T.3	0	0	1
.T.1	.T.2	.F.3	0	1	0
.T.1	.T.2	.T.3	1	1	1

With the payoff functions defined, we can compute global and Nash equilibria.

- For the Γ games network:

$$\textbf{Gne}(\Gamma) = \Big\{\big((1,0),(1,0),(1,0)\big); \big((0,1),(0,1),(0,1)\big); \big((\tfrac{1}{2},\tfrac{1}{2}),(\tfrac{1}{5},\tfrac{4}{5}),(1,0)\big)\Big\}$$

- For the Γ^m game:

$$\textbf{Nash}(\Gamma^m) = \Big\{\big((1,0),(1,0),(1,0)\big); \big((0,1),(0,1),(0,1)\big)\Big\}$$

- For the Γ^M game:

$$\textbf{Nash}(\Gamma^M) \supset \textbf{Gne}(\Gamma) \cup \Big\{\big((\tfrac{1}{2},\tfrac{1}{2}),(\tfrac{1}{5},\tfrac{4}{5}),(0,1)\big)\Big\}$$

Finally, we have $\textbf{Gne}(\Gamma^m) \subsetneq \textbf{Gne}(\Gamma) \subsetneq \textbf{Gne}(\Gamma^M)$; and none of the games network Γ and its two joint games Γ^M and Γ^m has the same equilibria. Thus, choice of the observer function is central especially if we are interested in finding a games network with a different structure but equivalent to the initial one.

6. ELEMENTARY MODULES

In this section, given a games network, we are interested in finding an *equivalent* one, but with a simpler structure. More precisely, we are looking for a *normal form* which is composed of the smallest possible game nodes (in sense of number of agents involved in game nodes). Game nodes of such a games network are called *elementary games* or *elementary modules*. The normal form can be used to obtain a better understanding of complex systems, where elementary modules underline how agents interplay.

Normal form can be obtained using successive separations, but figure 4 presents an algorithm which directly separates a game in its elementary modules. Thus, in order to find a games network normal form, we apply the separate algorithm to each game node.

Separation (as join operation) depends on a function ω (section 5.3). This function determines how payoffs of the original game are distributed in the different separated games. In the separation obtained with algorithm from figure 4, there exists, for each agent, one particular game which contains the whole payoffs of this agent. The *dependence* notion determines which other agents have to participate to this particular game. Dependence underlines interactions between agents; and intuitively, agents involved in the same elementary module are agents of the original game which are highly interacting.

Not all ω functions are compatible with the separation presented in figure 4. For example, *addition* is compatible but not the *max* function. More precisely, *functions with a neutral element*[1] are compatible. Thus, in the separated games, each agent is involved in two types of games: one game which contains all its payoffs, and possibly several other games where its payoffs are the neutral element of the separation function.

In order to rebuild the original game from the normal form, the definition of join operation considering a function with neutral element is changed as follows:

Definition 9. Join with neutral element. Let ω a function with e_ω as neutral element. Let $\Gamma = \langle \mathcal{A}, C, \mathcal{U}\rangle$ be a games network, let $G_1 = \langle A_1, u^1\rangle$ and $G_2 = \langle A_2, u^2\rangle$ be two game nodes of Γ ($G_1 \in \mathcal{U}, G_2 \in \mathcal{U}$), we define:

$$G_1 \overset{\omega}{\bigvee} G_2 = \langle A_1 \cup A_2, u\rangle$$

with:

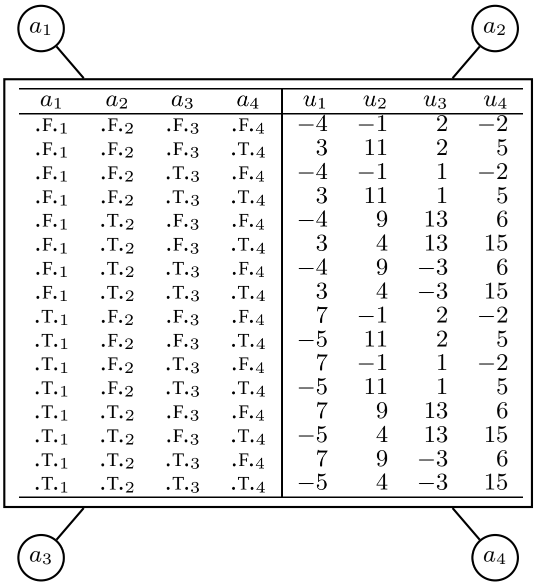
$$\forall c \in C_{(A_1 \cup A_2)},$$
$$\forall i \in A_1 - A_2 \quad u_i(c) = {u^1}_i\big(p(c, A_1)\big)$$
$$\forall i \in A_2 - A_1 \quad u_i(c) = {u^2}_i\big(p(c, A_2)\big)$$
$$\forall i \in A_1 \cap A_2$$
$$u_i(c) = \begin{cases} {u^2}_i\big(p(c,A_2)\big) & \text{if } {u^1}_i\big(p(c,A_1)\big) = e_\omega \\ {u^1}_i\big(p(c,A_1)\big) & \text{if } {u^2}_i\big(p(c,A_2)\big) = e_\omega \\ {u^1}_i\big(p(c,A_1)\big) & \text{if } {u^2}_i\big(p(c,A_2)\big) = {u^1}_i\big(p(c,A_1)\big) \\ e_\omega & \text{otherwise} \end{cases}$$

Section 6.1 precises the notion of dependance used to underline how agents are interacting. Section 6.2 explicits the separation algorithm from fig 4. These two ideas are ilustrated using game from fig. 3.

6.1 Dependence

Dependence provides an overview of the agent interplays in a game without having carefully studying the payoff function. Informally, an agent A depends on another agent B (or B influences A) if A's payoffs are altered by B's strategies.

[1]Recall that $e \in \mathbb{R}$ is a neutral element for $\omega : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ if and only if $\forall r \in \mathbb{R} \quad \omega(e, r) = \omega(r, e) = r$



a_1	a_2	a_3	a_4	u_1	u_2	u_3	u_4
$.F._1$	$.F._2$	$.F._3$	$.F._4$	-4	-1	2	-2
$.F._1$	$.F._2$	$.F._3$	$.T._4$	3	11	2	5
$.F._1$	$.F._2$	$.T._3$	$.F._4$	-4	-1	1	-2
$.F._1$	$.F._2$	$.T._3$	$.T._4$	3	11	1	5
$.F._1$	$.T._2$	$.F._3$	$.F._4$	-4	9	13	6
$.F._1$	$.T._2$	$.F._3$	$.T._4$	3	4	13	15
$.F._1$	$.T._2$	$.T._3$	$.F._4$	-4	9	-3	6
$.F._1$	$.T._2$	$.T._3$	$.T._4$	3	4	-3	15
$.T._1$	$.F._2$	$.F._3$	$.F._4$	7	-1	2	-2
$.T._1$	$.F._2$	$.F._3$	$.T._4$	-5	11	2	5
$.T._1$	$.F._2$	$.T._3$	$.F._4$	7	-1	1	-2
$.T._1$	$.F._2$	$.T._3$	$.T._4$	-5	11	1	5
$.T._1$	$.T._2$	$.F._3$	$.F._4$	7	9	13	6
$.T._1$	$.T._2$	$.F._3$	$.T._4$	-5	4	13	15
$.T._1$	$.T._2$	$.T._3$	$.F._4$	7	9	-3	6
$.T._1$	$.T._2$	$.T._3$	$.T._4$	-5	4	-3	15

Figure 3: A "4-agents game" to illustrate the separate algorithm

Definition 10. Agent dependence. Let $\langle A, C, u\rangle$ be a strategic game, let $(i, j) \in A^2, i \neq j$ be two agents. We denote by $i\delta_u j$ the dependance relation:

$$i\delta_u j \Leftrightarrow \exists (c_i, c_i') \in C_i^2, \exists c_{-i} \in \Pi_{k \in A - i} C_k \qquad u_j(c_{-i}, c_i) \neq u_j(c_{-i}, c_i')$$

Thus, with the dependence relation, we can determine all the agents which influence a given agent:

Definition 11. Set of influent agents. Let $G = \langle A, C, u\rangle$ be a strategic game, and $j \in A$ an agent. We denote by $\delta_u^-(j)$ the set of all agents which influence j:

$$\forall j \in A, \delta_u^-(j) = \{i \in A | i\delta_u j\}$$

Example 2. Dependences in fig. 3 game. From the payoffs we can deduce the following dependences:

$$a_1(.F._1, .F._2, .F._3, .F._4) \neq a_1(.F._1, .F._2, .F._3, .T._4) \Rightarrow a_4\delta_u a_1$$

$$a_2(.F._1, .F._2, .F._3, .F._4) \neq a_2(.F._1, .F._2, .F._3, .T._4) \Rightarrow a_4\delta_u a_2$$

$$a_3(.F._1, .F._2, .F._3, .F._4) \neq a_2(.F._1, .T._2, .F._3, .F._4) \Rightarrow a_2\delta_u a_3$$

$$a_4(.F._1, .F._2, .F._3, .F._4) \neq a_2(.F._1, .T._2, .F._3, .F._4) \Rightarrow a_2\delta_u a_4$$

6.2 Separate a game node

Figure 4 presents the *separate* algorithm which computes a normal form from a game.

Let $G = \langle A, C, u\rangle$ the starting game and ω a function of separation with e_ω as neutral element. First, the separate function research how many game nodes have to be created. The dependence graph is used to emphasize the interactions between agents

```
function separate(⟨A,u⟩ : game)
    U' := ∅;  g := 0;

    /* Compute game nodes to be created */
    FORALL i ∈ A DO
        g := g + 1;
        agent(g) := i ∪ δ_u^-(i) ;
    ENDFORALL
    U = [1 : g];
    FORALL g' ∈ [1 : g] DO
        U := U − { g'' ∈ U | (agent(g'') ⊂ agent(g')) ∨
                   (agent(g') = agent(g'') ∧ g'' < g') };
    ENDFORALL

    /* Attribution of payoffs */
    FORALL g ∈ U DO
        FORALL j ∈ agent(g) DO
            IF δ_u^-(j) ∩ agent(g) = δ_u^-(j) THEN
                FORALL c ∈ C_agent(g) DO
                    u_j^g(c) := pick(c,u);
                ENDFORALL
            ELSE
                FORALL c ∈ C_agent(g) DO
                    u_j^g(c) := e;
                ENDFORALL
            ENDIF
        ENDFORALL
            U' = U' ∪ {⟨agent(g), u^g⟩};
    ENDFORALL

    RETURN U';
```

Figure 4: Normal Form Algorithm for a Game

and thus determine which agents participate to a same game node. The game nodes are defined by the agents which are involved in. For each agent, a game node which contains all its predecessors exists and, given two game nodes $g_1 = \langle A_1, u_1\rangle$ and $g_2 = \langle A_2, u_2\rangle$, we cannot have $A_1 \subseteq A_2$ or $A_2 \subseteq A_1$.

Once we have the game nodes, we can compute the payoffs. Let $a \in A$ be an agent and $g = \langle A_g, u^g\rangle$ be a game node to be created.

- If all the predecessors of a are in g, we can easily compute the payoffs for a, because none of the missing agents in g have any influence on a's payoffs. In fact, for any game $\langle A^*, C^*, u^*\rangle$, we have:

$$\forall \sigma, \sigma' \in \Delta(C^*)^2, \forall j \in A^*$$
$$p(\sigma, j \cup \delta^-_{u^*(j)}) = p(\sigma', j \cup \delta^-_{u^*(j)}) \Rightarrow u^*_j(\sigma) = u^*_j(\sigma')$$

  Thus, given a game configuration c_g of g, each pure profile c_G of G, the starting game, such that the restriction of $c_G \Downarrow_{A_g} = c_g$ gives the same payoffs for a. The *pick* function in fig.4 chooses one of these c_G configuration.

- If at least one of the agents which influences a is not in g, we give e_ω, the neutral element of ω, to a as payoff.

If we consider a n-agents game, each player having p strategies, the complexity of algorithm from fig.4 is n^2p^n. However, the complexity is n^2 if each agent depends on all the others, i.e. if the game is elementary. The n^2p^n is obtained with games where agents are not very dependent, thus in that case, the game is highly separable and the complexity of computation of mixed Nash equilibria is highly decreased.

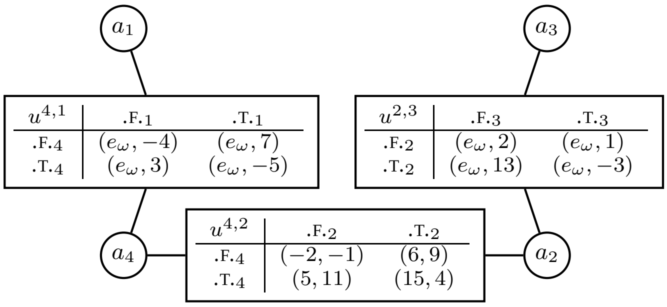


Figure 5: Normal Form of fig. 3 game

Example 3. Normal Form of fig. 3 game. According to the algorithm we can deduce that the game node from example 2 is separated into three game nodes, each one having 2 agents. Figure 5 describes the resulting games network.

7. CONCLUSION

In this paper we have proposed an extension of game theory, named games networks theory, which provides a framework to model complex systems in terms of sets of interacting agents. By contrast of game theory where all agents are interacting together, games networks allow definition of local interactions which help to understand the structure of complex systems. Each game which composes a games network represents a set of localy interacting agents. The issue of the games network dynamics is defined by games networks equilibria (**Gne**) which correspond to steady states.

Different games networks structures — how agents are connected to games — can provide the same global dynamics. Thus, we focus on the determination of a normal form games network composed of games as small as possible. The separate algorithm computes a normal form. It is based on the notion of dependence, which allows the study of interactions occuring in a network. With dependence, highly interacting agents can be determined and gathered in a same game node.

The normal form game network shows clusters of agents tightly coupled by interactions. Considering applications in Biology, this decomposition is useful to show specific interacting components, and to reveal the association between these components and the biological function they may support.

8. REFERENCES

[1] R. Alur, T. Henzinger, and O. Kupferman. Alternating-time temporal logic. In *Proceedings of the 38th IEEE Symposium on Foundations of Computer Science*, Florida, October 1997.

[2] C. Chettaoui, F. Delaplace, M. Manceny, and M. Malo. Games Network & Application to PAs system. In *Information Processing in Cells and Tissues (IPCAT)*, 2005.

[3] F. Delaplace and M. Manceny. Games network. Technical Report 101-2004, Laboratoire de Méthodes Informatiques (LaMI), CNRS-UMR 8042, Université d'Évry-Val d'Essonne, 2004.

[4] M. Kearns, M. L. Littman, and S. Singh. Graphical models for game theory. In *Proceedings of the 17th Conference on Uncertaintly in Artificial Intelligence (UAI)*, pages 253–260, 2001.

[5] D. Koller and B. Milch. Multi-agent influence diagrams for representing and solving games. In *Proceedings of the 17th*

International Joint Conference on Artificial Intelligence (IJCAI), pages 1027–1034, 2001.

[6] D. M. Kreps. *A Course in Microeconomic Theory*. Princeton University Press, 1990.

[7] P. La Mura. Game networks. In *Proceedings of the 16th Conference on Uncertaintly in Artificial Intelligence (UAI)*, pages 335–342, 2000.

[8] J. Maynard Smith. The games the lizards play. *Nature*, 380:198–199, 1996.

[9] R. D. McKelvey and A. McLennan. Computation of equilibria in finite games. In *Handbook of Computational Economics*, volume 1, pages 87–142. Elsevier, 1996. http://econweb.tamu.edu/gambit/.

[10] R. D. McKelvey and A. McLennan. Computation of equilibria in finite games. In *Handbook of Computational Economics*, volume 1, pages 87–142. Elsevier, 1996. http://econweb.tamu.edu/gambit/.

[11] R. B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, 1991.

[12] M. A. Nowak and K. Sigmund. Games on grids. In *The Geometry of Ecological Interactions: Simplifying Spatial Complexity*, pages 134–150, 2000.

[13] M. A. Nowak and K. Sigmund. Evolutionary dynamics of biological games. *Sciences*, 303(6):793–799, februar 2004.

[14] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*, volume 380. MIT Press, 1994.

[15] C. H. Papadimitriou. Game theory and mathematical economics: a theoretical computer scientist's introduction. In *42nd IEEE Symposium on Foundations of Computer Science: Proceedings*, pages 4–8, 2001.

[16] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, 34:166–176, June 2003.