

# Complexity of Pure Equilibria in Bayesian Games

**Georg Gottlob**

Computing Laboratory  
Oxford University  
OX1 3QD Oxford, UK  
georg.gottlob@comlab.ox.ac.uk

**Gianluigi Greco**

Dip. di Matematica  
Università della Calabria  
I-87030 Rende, Italy  
ggreco@mat.unical.it

**Toni Mancini**

Dip. di Informatica e Sistemistica  
Università di Roma “La Sapienza”  
I-00198 Roma, Italy  
tmancini@dis.uniroma1.it

## Abstract

In this paper we make a comprehensive study of the complexity of the problem of deciding the existence of equilibria in strategic games with incomplete information, in case of pure strategies. In particular, we show that this is NP-complete in general Bayesian Games in Standard Normal Form, and that it becomes PP-hard (and, in fixed-precision scenarios, PP-complete), when the game is represented succinctly in General Normal Form. Suitable restrictions in case of graphical games that make the problem tractable are also discussed.

## 1 Introduction

Game theory has a very general scope, encompassing questions that are basic to all of the social sciences, and offering insights into any economic, political, or social situation that involves multiple players that have different goals and preferences. In its generality, game theory may be defined as the study of mathematical models of conflict and cooperation between intelligent rational decision-makers, cf., e.g., [Myerson, 1997]. Not surprisingly, in Artificial Intelligence the framework of game theory has been immediately recognized to be very appealing, being the means for an artificial agent to act with rationality and intelligence when interacting with others. Additionally, game theory demonstrated its effectiveness in the recent results on algorithms for load balancing and routing in large networks and the Internet, where it has been extensively used for distributing the logic of the decision processes among many nodes (cf., e.g., [Feldmann *et al.*, 2003]).

The most well known form of games studied in the literature is that of strategic games. In this context, the main solution concept is that of *equilibrium* (called *Nash equilibrium* in games with complete information [Nash, 1951]). Equilibria model scenarios in which no player can improve his utility by unilaterally changing his strategy, or equivalently, such that the strategy chosen by any player is the *best response* to the strategies chosen by the others. Any strategy which is not a best response indicates that the corresponding player is not acting rationally, because he could improve his utility by choosing a different one. Although the existence of equilibria in strategic games is guaranteed, when the so called *randomized strategies* are allowed (cf. Section 2),

this is not true in the important case in which strategies are forced to be *pure* [Gottlob *et al.*, 2003]. In case of randomized strategies, the problem of computing an equilibrium was recently shown to be PPAD-complete, even for two-player games [Chen and Deng, 2006]; and, further results have been established for strategic games under various restrictions (cf., e.g., [Gilboa and Zemel, 1989; Megiddo and Papadimitriou, 1991; Papadimitriou, 1994b; Conitzer and Sandholm, 2003; Daskalakis and Papadimitriou, 2005]).

On the other hand, in this paper we consider games with *incomplete* information (aka *Bayesian Games*) that have been specifically introduced in the literature [Harsanyi, 1968] to account for uncertainties in the knowledge that the single players have about the games and their counterparts. Indeed, in these games players' payoff depends not only on the actions of the other players but also on their own *private type*, for which a probability distribution is a priori known.

The equilibrium concept relevant for this kind of games is that of *Bayesian Equilibrium*, in which each player plays a best response to the other players in expectation over their private types. The complexity of deciding the existence of a Bayesian Equilibrium in case of pure strategies (short:  $\exists$ PBE problem) has been recently investigated by Conitzer and Sandholm [2003], who proved its NP-hardness for symmetric two-players games, in the case where, for each player, the utility function and the probability distribution over other players' types are represented extensively as tables (called *Standard Normal Form (SNF)* in Section 2). Moreover, tractable classes of Bayesian Games have also been singled out in [Singh *et al.*, 2004], where an efficient algorithm is provided for finding approximate equilibria when the utility function of each player depends only on the actions and realized types of some *neighbor* players (cf. *Graphical Normal Form (GNF)*), and when the kind of interaction formed by the neighborhood relation is a tree.

However, a complete picture of the complexity of Bayesian Games is still missing in the literature, both for pure and randomized strategies. As an example, it is not known what happens if, unlike [Conitzer and Sandholm, 2003], the utility functions and the probability distributions are not explicitly represented as tables (cf. *General Normal Form (GenNF)*); and it has been left open in [Singh *et al.*, 2004] if the tractability result holds for larger classes of games in GNF, e.g., for games whose players interactions have tree-like structure.

The aim of this paper is precisely to shed lights on these aspects, by focusing on the case of pure strategies. Our contributions are as follows:

- ▷ In Section 3 we show that the result of Conitzer and Sandholm is tight, since  $\exists$ PBE is NP-complete, even if only one among the utility function and the probability distribution is represented as a table, for each player in the game.
- ▷ We study  $\exists$ PBE on the class of games in GenNF, and we show that the need of dealing with incomplete information has a strong impact on the intrinsic complexity of the games. Indeed, we show that the problem is PP-hard, which is quite surprisingly given that deciding the existence of a Pure Nash equilibrium is NP-complete [Gottlob *et al.*, 2003].
- ▷ We continue the analysis of  $\exists$ PBE on games in GenNF, under the assumption that the utility functions and the probability distributions range over real numbers with a *fixed-precision*. In this scenario, we prove that the problem is, in fact, complete for the class PP.
- ▷ Given that it is not generally known how efficient heuristics can be devised for problems hard for PP, it appears crucial to identify tractable classes of games. In Section 4, we face this problem and we show that  $\exists$ PBE is feasible in polynomial time not only on classes of acyclic games, but also on games of bounded hypertree width [Gottlob *et al.*, 2002] which is one of the broadest known generalizations of hypergraph acyclicity.

## 2 Games of Incomplete Information

**Preliminaries.** Given sets  $S$  and  $R$ , a  $S$ -tuple with components in  $R$  is a total function from  $S$  to  $R$ . A  $S$ -tuple is denoted by a bold-face lowercase letter, e.g.,  $\mathbf{t}$ . Set  $R$  can be either a set of atomic values or a set of functions. Given a  $S$ -tuple  $\mathbf{t}$  and an element  $\bar{s} \in S$  (resp. a set  $\bar{S} \subseteq S$ ), we denote by  $\mathbf{t}[\bar{s}]$  (resp.  $\mathbf{t}[\bar{S}]$ ) the value of the  $\bar{s}$ -component of tuple  $\mathbf{t}$  (resp. the tuple obtained by taking only  $\mathbf{t}$ 's  $\bar{s}$ -components, with  $\bar{s} \in \bar{S}$ ). By  $\mathbf{t}_{-\bar{s}}$  (resp.  $\mathbf{t}_{-\bar{S}}$ ) instead, we denote the  $(S - \{\bar{s}\})$ -tuple (resp. the  $(S - \bar{S})$ -tuple) obtained by  $\mathbf{t}$  by removing  $\mathbf{t}$ 's  $\bar{s}$ -component (resp. all  $\mathbf{t}$ 's  $\bar{s}$ -components, with  $\bar{s} \in \bar{S}$ ). Furthermore, given a  $S$ -tuple  $\mathbf{t}$ , an element  $s \in S$ , and an element  $r \in R$ , we denote by  $\mathbf{t}_{s \rightarrow r}$  the tuple obtained from  $\mathbf{t}$  by replacing its  $s$ -component with  $r$  (i.e., the  $S$ -tuple  $\mathbf{t}'$  such that, for each  $i \in S - \{s\}$ ,  $\mathbf{t}'[i] = \mathbf{t}[i]$ , and  $\mathbf{t}'[s] = r$ ), and by  $\langle \mathbf{t}; t_x \rangle$ ,  $x \notin S$  the  $(S \cup \{x\})$ -tuple  $\mathbf{t}'$  such that,  $\mathbf{t}'[s] = \mathbf{t}[s]$ ,  $\forall s \in S$ , and  $\mathbf{t}'[x] = t_x$ .

Finally, for a set  $S$ , we denote by  $\Delta(S)$  the set of all probability distributions over elements in  $S$ , i.e., the set of real-valued functions  $\tau : S \rightarrow [0, 1]$  such that  $\sum_{s \in S} \tau(s) = 1$ .

**Bayesian Games.** A strategic game  $\mathcal{G}$  consists of a set of players  $N$  and, for each player  $i \in N$ , of a set of possible actions  $A_i$  and an utility function  $u_i : \times_{i \in N} A_i \rightarrow \mathbb{R}$ , that describes the payoff of player  $i$  as a function of his own and other players' actions. The behavior of a strategic game is as follows: each player  $i$  simultaneously chooses an action in his set  $A_i$ . The tuple of actions chosen by all players is called

the *action-profile*, and it is denoted by  $\hat{\mathbf{a}}$ . As a consequence of this choice, each player  $i$  receives a payoff of  $u_i(\hat{\mathbf{a}})$ .

These games are also called *games with complete information*, because players have all perfect information about the game. Bayesian games [Harsanyi, 1968] have been introduced to account for situations in which such assumption cannot be made, since there may be information which is uncertain, or unobservable by some players. This uncertainty is modelled by introducing unknown parameters, the *types* of the different players, into the utility functions.

**Definition 1 (Bayesian Game [Harsanyi, 1968]).** A Bayesian Game  $\mathcal{G}$  consists of a set of players  $N$  and, for each player  $i \in N$ :

- A set of possible actions  $A_i$ ;
- A set of possible types  $T_i$ ;
- A probability function  $p_i : T_i \rightarrow \Delta(\times_{j \in N - \{i\}} T_j)$ , assigning a probability distribution over  $\times_{j \in N - \{i\}} T_j$  to each possible type of player  $i$ . We write  $p_i(\cdot | t_i)$  to denote the belief that player  $i$  has over the types of the other players when he has type  $t_i \in T_i$ ;
- An utility function  $u_i : \times_{i \in N} A_i \times \times_{i \in N} T_i \rightarrow \mathbb{R}$ , that describes the payoff of player  $i$  as a function of the actions of all players and their types (including  $i$ 's).  $\square$

Although probability functions  $p_i(\cdot | t_i)$  can be given separately for each player  $i$ , a common assumption is that a prior joint distribution  $P(\times_{i \in N} T_i)$  over all the type profiles (i.e.,  $N$ -tuples  $\mathbf{t}$ ) exists, and is common knowledge among all players. Under this assumption, the single functions  $p_i(\cdot | t_i)$  can be built by the single players according to the Bayes' Rule. Indeed, as for the reasonability of this assumption (called the *consistent beliefs* assumption), it can be proven that every Bayesian Game with finite type set can be transformed into an equivalent game with consistent beliefs.

A Bayesian Game behaves as follows:

1. At the beginning, each player  $i$  "gets revealed" his *realized type*  $\hat{t}_i \in T_i$  by an external entity (usually called "Nature"). Under the consistent beliefs assumption, we can think that the realized types for all players are drawn from the joint prior distribution  $P(\cdot)$ .
- Each player  $i$  knows only his realized type  $\hat{t}_i$ , and does not know the realized types of the other players (this realizes the incompleteness of information discussed so far). The  $N$ -tuple of realized types of all players is called the *realized type-profile*, and it is denoted by  $\hat{\mathbf{t}}$ .
2. According to their realized types, each player  $i$  simultaneously chooses an action, in his set  $A_i$ . The tuple of actions chosen by all players is called the *action-profile*, and it is denoted by  $\hat{\mathbf{a}}$ .
3. As a consequence of this choice, and depending on the realized type-profile, each player  $i$  receives a payoff of  $u_i(\hat{\mathbf{a}}, \hat{\mathbf{t}})$ . We observe that, for the players to be able to evaluate their actual payoffs, information about the realized types of the other players must be observable after the games terminates. Indeed, since this information is

hidden when actions have to be decided, what the players can evaluate is their *expected* payoff, according to the probability distribution  $P(\cdot)$  on types.

Given a Bayesian Game  $\mathcal{G}$  and a player  $i \in N$ , a *randomized strategy* for  $i$  is a probability function  $\sigma_i : T_i \rightarrow \Delta(A_i)$ , i.e., a total function assigning, to each type in  $T_i$ , a probability distribution over the set of actions available to player  $i$ . We write  $\sigma_i(\cdot|t_i)$  to denote the probability distribution over  $A_i$  when the type of player  $i$  is  $t_i \in T_i$ . Hence,  $\sigma_i(a_i|t_i)$  is the probability with which player  $i$  will choose action  $a_i \in A_i$  in case his realized is  $t_i$ . A *randomized strategy profile* is a  $N$ -tuple  $\sigma$  of randomized strategies, one for each player. In this paper we are interested in *pure strategy profiles*, i.e., strategy profiles  $\sigma$  such that, for each player  $i \in N$ , the strategy for player  $i$ ,  $\sigma[i] = \sigma_i(\cdot)$  assigns, to each type  $t_i \in T_i$ , a *single* action in  $A_i$  (and not a probability distribution over the actions). We write  $\sigma_i(t_i)$  to denote the action  $a_i \in A_i$  chosen by player  $i$  when his type is  $t_i$ . A pure strategy can be equivalently regarded as a randomized strategy where all probability distributions (one for each type  $t_i \in T_i$ ) assign 1 to action  $\sigma_i(t_i)$ , and 0 to the others.

**Definition 2.** A *Pure Bayesian Equilibrium* (PBE) is a strategy profile  $\sigma$ , with each  $\sigma_i(\cdot) = \sigma[i]$  being a pure strategy for player  $i \in N$ , such that, for every type  $t_i \in T_i$ :

$$\sigma_i(t_i) \in \operatorname{argmax}_{a_i \in A_i} \sum_{t^{-i} \in \prod_{j \in N - \{i\}} T_j} p_i(t^{-i}|t_i) u_i(\mathbf{a}, \langle t^{-i}; t_i \rangle) \quad (1)$$

where, to ease notation, we denoted by  $\mathbf{a}$  the  $N$ -tuple such that  $\mathbf{a}[k] = \sigma_k(t^{-i}[k])$  for  $k \neq i$  (i.e., the action expected to be played by player  $k$  according to strategy profile  $\sigma$  when he is of type  $t^{-i}[k]$ ), and  $\mathbf{a}[i] = a_i$ .  $\square$

It is well known that, although every Bayesian Game has a BE, it does not necessarily have a PBE.

It can be observed that games with complete information are a particular case of Bayesian Games in which all players may assume only one type. All the definitions made for Bayesian Games (strategies, strategy profiles and equilibria) are generalizations of analogous notions for games with complete information. In particular, equilibria in games with complete information are called (Pure) Nash Equilibria [Nash, 1951].

**Representing games.** Games can be represented in different ways. Below we briefly review the most used ones:

**Standard Normal Form (SNF):** In games of this form (cf., e.g. [Osborne and Rubinstein, 1994]), sets  $N$  and  $A_i, T_i, i \in N$  are represented explicitly,  $P(\mathbf{t})$  and  $u_i(\mathbf{a}, \mathbf{t}), \forall i \in N$  are represented by tables of  $|N|+1$  and  $2|N|+1$  columns, having  $\prod_{i \in N} |T_i|$  and  $\prod_{i \in N} (|T_i| \times |A_i|)$  tuples, respectively. This form is also known as *Strategic Form* (cf., e.g., [Myerson, 1997]).

**Graphical Normal Form (GNF):** This form is suitable to represent games in which the utility function of each player depends only on the actions chosen and realized types of a *subset* of the other players (cf., e.g., [Kearns et al., 2001; Vickrey and Koller, 2002]). For each player  $i$ , let  $X_i$  be the set of players  $j$  such that  $u_i(\mathbf{t}, \mathbf{a})$  does

not depend on  $\mathbf{t}[j]$  and  $\mathbf{a}[j]$ . Players in  $N - X_i - \{i\}$  are called the *neighbors* of player  $i$ , and denoted by  $Neigh(i)$ . The representation is similar to SNF, but tables for utility functions  $u_i(\cdot)$  may have a reduced number of columns. In particular, for each player  $i$ , column relative to actions and types of players in set  $X_i$  can be omitted.

**General Normal Form (GenNF):** This is the most compact representation for a game, since  $P(\mathbf{t})$  and  $u_i(\mathbf{a}, \mathbf{t}), i \in N$  can be given by (succinctly encoded) functions that can be computed in polynomial-time in the size of their input (such form is a generalization of that defined in [Gottlob et al., 2003] for games with complete information). We also assume that a polynomial-time computable function  $Neigh(\cdot) : N \rightarrow 2^N$  that, for each player, returns his neighbors, is part of the representation. We will see that such function is very useful in case the interaction among players is restricted (cf. Section 4). It can be observed that games in SNF and GNF are trivially in GenNF (as for the former, the function  $Neigh(\cdot)$  would return the set  $N - \{i\}$  for any player  $i \in N$ ).

### 3 Intractability results

In this section, we face the complexity of deciding whether a Bayesian Game has a PBE (the  $\exists$ PBE problem). We start with the class of games in SNF.

**Theorem 1.**  $\exists$ PBE for games in SNF is NP-complete. Hardness holds even for symmetric, two-players games, while membership holds even if only one among  $P(\cdot)$  and the vari-ous  $u_i(\cdot)$  is given explicitly as a table.

*Proof. (Hardness)* Hardness follows from [Conitzer and Sandholm, 2003, Theorem 2], where the NP-complete Set Covering problem is reduced to  $\exists$ PBE of a game in SNF with only 2 players. Actually, the definition of a Bayesian Game given there is more restricted than ours, since the utility function of player  $i$  does not depend on the realized types of the other players.

*(Membership)* We prove the membership in NP for the more general case of  $N$  unbounded and only one of the functions among  $P(\cdot)$  and  $u_i(\cdot)$  represented as a table. In this case, the size of the input is exponential in the number of players  $|N|$ . Problem  $\exists$ PBE can be solved by guessing a strategy-profile  $\sigma$  in polynomial time (it consists in guessing an action for every pair player/type), and checking whether  $\sigma$  is a PBE. The latter task can be performed in polynomial time in the size of the input, since, for each player and for each type, formula (1) in Definition 2 can be evaluated in time polynomial in the size of the table encoding  $P(\cdot)$  (which, in turn, is smaller than that encoding  $u_i(\cdot)$ ).  $\square$

By using the same line of reasoning as in the proof above, we get a similar result for games in GNF.

**Theorem 2.**  $\exists$ PBE for games in GNF is NP-complete.

Let us now turn to games in GenNF. We next show that  $\exists$ PBE is hard for the class PP (aka majority-P) of languages that can be decided by a nondeterministic Turing machine

in polynomial time where the acceptance condition is that a majority (more than half) of computation paths accept (cf., e.g., [Papadimitriou, 1994a]). This class contains NP and is contained in PSPACE.

**Theorem 3.**  $\exists$ PBE for games in GenNF with unbounded number of players is PP-hard. The result holds also for games in which players can assume just two types.

*Proof.* We reduce the canonical PP-complete MAJ-SAT problem (cf., e.g., [Papadimitriou, 1994a]) to  $\exists$ PBE. Given a propositional formula  $\varphi$  over  $n$  variables as input, MAJ-SAT( $\varphi$ ) is true iff more than half of its possible assignments satisfies it.

Let  $V$  be the set of variables of formula  $\varphi$ . We build a Bayesian Game  $\mathcal{G}$  that has a PBE iff MAJ-SAT( $\varphi$ ) is true. The game has a set of  $n+3$  players  $N = V \cup \{eval\} \cup \{R, S\}$ , with the following characteristics:

- Players in  $V$  (corresponding to the variables of formula  $\varphi$ ) may assume one of two types  $\{true, false\}$ , have only one available action *nop*, and utility functions which evaluate always to 0.
- *eval* (“evaluator”) may assume only one type, but has two available actions: *count* and *pass*. His utility function  $u_{eval}(\cdot)$  is defined as follows:

$$u_{eval}(\mathbf{a}, \mathbf{t}) = \begin{cases} \text{if } \mathbf{a}[eval] = \text{count}: & \begin{cases} \text{if } \mathbf{t}_{-\{eval, R, S\}} \models \varphi : 1 \\ \text{otherwise: } 0 \end{cases} \\ \text{otherwise: } 1/2 \end{cases}$$

- Players  $R$  and  $S$  may assume only one type, and have two available actions  $a_1$  and  $a_2$ . Their utility functions  $u_R(\cdot)$  and  $u_S(\cdot)$  are such that they evaluate to 0 if  $a_{eval} = \text{count}$ , while, in the other cases (where  $a_{eval} = \text{pass}$ ) they are defined as follows (independently on the types and actions of players in  $V$ ):

$$(u_R(a_R, a_S), u_S(a_R, a_S)) = \begin{cases} (0, 1) & \text{if } (a_R, a_S) = (a_1, a_1) \\ (1, 0) & \text{if } (a_R, a_S) = (a_1, a_2) \\ (1, 0) & \text{if } (a_R, a_S) = (a_2, a_1) \\ (0, 1) & \text{if } (a_R, a_S) = (a_2, a_2) \end{cases}$$

The idea behind such choice for  $u_R$  and  $u_S$  is that, if  $a_{eval} = \text{count}$  both  $a_1$  or  $a_2$  are always best-responses for players  $R$  and  $S$ ; on the other hand, if  $a_{eval} = \text{pass}$ , their utility functions prevent the existence of a PBE.

- $P(\cdot) \in \Delta(\times_{i \in V \cup \{eval\} \cup \{R, S\}} T_i)$  is uniform: since  $|T_{eval}| = |T_R| = |T_S| = 1$  and  $|T_v| = 2$  for each  $v \in V$ , this means that, for each type-profile  $\mathbf{t}$ ,  $P(\mathbf{t}) = 1/2^n$ .

We now show that MAJ-SAT( $\varphi$ ) is true iff  $\mathcal{G}$  has a PBE.

*(only if)-part.* If MAJ-SAT( $\varphi$ ) is true, we observe that any strategy-profile  $\sigma$  such that  $\sigma[eval] = \text{count}$  is a PBE. In fact, that strategy is an equilibrium for players in  $V \cup \{R, S\}$  since their expected payoff cannot be different from 0. Moreover, action *count* is a best-response for player *eval*, that achieves an expected payoff of  $1/2^n \cdot \#\varphi > 1/2$  (with  $\#\varphi > 2^{n-1}$  being the number of models of formula  $\varphi$  and  $1/2$  being the

payoff that he would obtain by unilaterally changing his action into *pass*).

*(if)-part.* As for the other direction, if MAJ-SAT( $\varphi$ ) is false, *eval* would play *pass*, which guarantees him higher utility, but in this case, it can be observed that the structure of the utility functions for players  $R$  and  $S$  prevents them from reaching an equilibrium.  $\square$

The reader may now wonder whether the problem is, in fact, complete for PP. While this might appears at a first sight very natural, the formal proof has to take care of the fact that both  $P(\cdot)$  and  $u_i(\cdot)$  (for each  $i \in N$ ) are real-valued functions. Hence, to prove the membership result, we next restrict ourselves to real values with *fixed-precision*, i.e., we assume that all the values of interest for the game comes as multiples of a value  $\epsilon = 10^{-c}$ , for some fixed natural number  $c$ .

**Theorem 4.** In the fixed-precision setting,  $\exists$ PBE for games in GenNF with unbounded number of players is PP-complete.

*Proof. (Sketch.)* Assume, wlog, that all such functions range in the domain of reals in  $[0, 1]$ . Then, we can regard all the functions in the game as ranging in the integer domain  $[0, 1/\epsilon]$ . We can now build a NP Turing Machine  $M_{ep}$  (“ep” stands for “expected payoff”) that, given as input  $i \in N, t_i \in T_i, a_i \in A_i, \sigma_{-i}$ , and a boolean *accept*, behaves as follows: guesses a tuple  $\mathbf{t}^{-i}$  and then performs  $p_i(\mathbf{t}^{-i}|t_i) \cdot u_i(\mathbf{a}, \mathbf{t}) \leq (1/\epsilon)^2$  additional branches (with  $\mathbf{a}$  and  $\mathbf{t}$  defined as in formula (1)), accepting (resp. rejecting) on all of them if *accept* is *true* (resp. *false*). Such machine will have

$$\sum_{\substack{\mathbf{t}^{-i} \in \\ j \in N - \{i\}}} p_i(\mathbf{t}^{-i}|t_i) \cdot u_i(\mathbf{a}, \mathbf{t})$$

computation paths, all accepting or all rejecting, depending on the value for *accept*. Build now a second Turing Machine  $M$  that, given  $\sigma$ , behaves as follows: for each triple  $(i, t_i, a_i)$ , ( $i \in N, t_i \in T_i, a_i \in A_i - \{\sigma[i]\}$ ) makes two branches, on one invoking  $M_{ep}(i, t_i, \sigma[i], \sigma_{-i}, \text{true})$  and on the other  $M_{ep}(i, t_i, a_i, \sigma_{-i}, \text{false})$ . Such machine will have a number of accepting paths lower than that of rejecting ones if and only if  $\sigma$  is not a PBE. Machine  $M$  clearly operates in NP.  $\square$

Finally, we conclude this section by observing that the number of players plays a crucial role in the PP-hardness above.

**Theorem 5.**  $\exists$ PBE for games in GenNF is NP-complete, if the number of players  $|N|$  is bounded by a constant.

*Proof. (Hardness)* The hardness follows from Theorem 1.

*(Membership)* Problem  $\exists$ PBE can be solved by guessing a strategy-profile  $\sigma$  in polynomial time (it consists in guessing an action for every pair player/type), and checking whether  $\sigma$  is a PBE. In case of  $|N|$  bounded by a constant, formula (1) clearly shows that such task can be performed in polynomial time.  $\square$

## 4 Tractability results

In order to look for tractable cases of the  $\exists$ PBE problem, we might think of issuing some suitable restrictions on the games. For instance, one may think about restrictions on the maximum number of neighbors of each player, on the maximum number of types of each player, and on the kinds of interaction among players.

Results in Section 3 show that, exploiting only one of the restrictions above is not enough: requiring the size of the neighborhood, or the maximum number of types for each player to be bounded by a constant does not lead to tractability (cf., respectively, Theorem 1 and [Gottlob *et al.*, 2003] for the more restricted case of games with complete information, i.e., games in which players can assume only one type). On the other hand, it is possible to show that, by combining orthogonal, but weaker restrictions, we can obtain somewhat appealing tractable cases. To this end, we will exploit the concepts of *intricacy*, *hyper-graph* and *hypertree-width* of a game [Gottlob *et al.*, 2003] below, and we show that the problem  $\exists$ PBE becomes polynomial in case of games  $\mathcal{G}$  having *bounded intricacy* (or, equivalently, the *small neighborhood* property), bounded number of types, and whose associated hypergraph  $\mathcal{H}(\mathcal{G})$  has *bounded hypertree-width*.

Definitions of intricacy and hyper-graph have been proposed for games with complete information only, but we show that they can be used, without modification, also in the context of Bayesian Games. Formally, given a game  $\mathcal{G}$ , the intricacy of  $\mathcal{G}$ ,  $i(\mathcal{G})$ , is defined as

$$i(\mathcal{G}) = \max_{j \in N} \left( \frac{|Neigh(j)| \times \log |A_j|}{\log \|\mathcal{G}\|} \right) \quad (2)$$

where  $\|\mathcal{G}\|$  is the size of the representation of  $\mathcal{G}$ .

A game  $\mathcal{G}$  that has *bounded intricacy*, i.e., for which  $i(\mathcal{G}) \leq k$  for a fixed constant  $k$ , also enjoys the so-called *small neighborhood* property: for each player  $j \in N$ ,  $|Neigh(j)| \in \mathcal{O}(\log \|\mathcal{G}\| / \log |A_j|)$ . It can be observed that while bounded neighborhood implies bounded intricacy, the reverse does in general not hold. Hence, bounding intricacy could be a suitable restriction for those games in which bounding the number of neighbors is too severe.

Moreover, given a game  $\mathcal{G}$ , the hyper-graph  $\mathcal{H}(\mathcal{G}) = (V, E)$  of  $\mathcal{G}$  has players as vertices, i.e.,  $V = N$ , and set of hyper-edges defined as  $E = \{j \cup Neigh(j) \mid j \in N\}$ . For a game  $\mathcal{G}$ , its associated hypergraph encodes the structural interactions among the players. In many cases, these interaction are not too intricate, and the notion of hypertree-width helps in singling out these scenarios.

**Definition 3** ([Gottlob *et al.*, 2002]). Given a hypergraph  $H = (V_H, E_H)$ , a hypertree decomposition of  $H$  is a hypertree  $HD = \langle T, \chi, \lambda \rangle$  for  $H$ , where  $T = (V_T, E_T)$  is a tree, and  $\chi$  and  $\lambda$  are labelling functions that associate to each vertex  $p \in V_T$  two sets  $\chi(p) \subseteq V_H$  and  $\lambda(p) \subseteq E_H$ . Moreover,  $HD$  satisfies all the following conditions:

1. For each hyper-edge  $h \in E_H$  of  $H$ , there exists  $p \in V_T$  s.t.  $h \subseteq \chi(p)$ ;
2. For each vertex  $v \in V_H$  of  $H$ , the set  $\{p \in V_T \mid v \in \chi(p)\}$  induces a (connected) subtree of  $T$ ;
3. For each  $p \in V_T$ ,  $\chi(p) \subseteq \bigcup_{e \in \lambda(p)} e$ ;

4. For each  $p \in V_T$ ,  $(\bigcup_{e \in \lambda(p)} e) \cap (\bigcup_{p' \in T_p} \chi(p')) \subseteq \chi(p)$ , where  $T_p$  is the set of nodes in the subtree rooted at  $p$ .

The width of  $HD$  is  $\max_{p \in V_T} |\lambda(p)|$ . The hypertree-width  $hw(H)$  of an hyper-graph  $H$  is the minimum width over all its hypertree decompositions.  $\square$

To prove our main tractability result, we will exploit a transformation from a Bayesian Game into a game with complete information, and show that it preserves important structural properties of the game. The transformation is called the *type-agent representation* of a Bayesian Game, as it was originally discussed in [Harsanyi, 1968]. In the type-agent representation there is a player (called, to avoid confusion, *type-agent*) for every possible pair  $(i, t_i)$ , with  $i \in N$  and  $t_i \in T_i$ . The set of actions available to each type-agent  $(i, t_i)$  is exactly the set  $A_i$  of actions available to player  $i$ . The idea is that type-agent  $(i, t_i)$  is responsible for selecting the action that player  $i$  would use in the original Bayesian Game if  $t_i$  is his realized type. Hence, an action profile for the new game is a tuple  $\mathbf{a}$  having a component  $\mathbf{a}[(i, t_i)]$  for each type-agent  $(i, t_i)$ . Also the utility functions of each type-agent  $(i, t_i)$  are derived from that of player  $i$ , being defined as the conditionally expected utility to player  $i$  given that  $t_i$  is his realized type. Formally, the utility function for type-agent  $(i, t_i)$ ,  $v_{(i, t_i)}(\mathbf{a})$ , is as follows:

$$v_{(i, t_i)}(\mathbf{a}) = \sum_{\mathbf{t}^{-i} \in \prod_{j \in N - \{i\}} T_j} p_i(\mathbf{t}^{-i} | t_i) u_i(\mathbf{a}', \langle \mathbf{t}^{-i}; t_i \rangle) \quad (3)$$

where, to ease notation, we denoted by  $\mathbf{a}'$  the  $N$ -tuple having  $\mathbf{a}[(j, \mathbf{t}[j])]$  as its  $j$ -component  $\mathbf{a}'[j]$ , for all  $j \in N$ .

BEs of the original game are in one-to-one correspondence with Nash Equilibria of the new game with complete information.

**Theorem 6.**  $\exists$ PBE is feasible in polynomial time for the class of games having bounded hypertree-width, bounded intricacy, and bounded number of types, even with unbounded number of players.

*Proof.* We exploit the transformation of the Bayesian Game  $\mathcal{G}$  into a game  $\mathcal{G}'$  with complete information described above. We remind the PBEs of  $\mathcal{G}$  are in one-to-one correspondence with Pure Nash Equilibria of  $\mathcal{G}'$ .  $\mathcal{G}'$  has a set  $N'$  of  $\sum_{i \in N} |T_i|$  type-agents  $(i, t_i)$  (with  $t_i \in T_i$ ), each one having utility function  $v_{(i, t_i)}(\cdot)$  defined as in (3).

Since, in the Bayesian Game  $\mathcal{G}$ , player  $i$  has only set  $Neigh(i)$  as neighbors, functions  $v_{(i, t_i)}(\cdot)$  and  $u_i(\cdot)$  do not depend on actions and types of players in set  $X_i = N - Neigh(i) - \{i\}$ . Hence,  $\mathbf{a}$  can be regarded as a tuple in  $A_i \times \prod_{j \in Neigh(i)} A_j^{T_j}$ . Moreover, by splitting tuple  $\mathbf{t}^{-i}$  into  $\langle \mathbf{t}_{Neigh(i)}^{-i}; \mathbf{t}_{X_i}^{-i} \rangle$ ,  $p_i(\mathbf{t}^{-i} | t_i) = p_i(\mathbf{t}_{Neigh(i)}^{-i}, \mathbf{t}_{X_i}^{-i} | t_i)$  can be rewritten as  $p_i(\mathbf{t}_{Neigh(i)}^{-i} | t_i) \cdot p_i(\mathbf{t}_{X_i}^{-i} | \mathbf{t}_{Neigh(i)}^{-i}, t_i)$ . Hence,  $v_{(i, t_i)}(\cdot)$  becomes:

$$v_{(i, t_i)}(\mathbf{a}) = \sum_{\mathbf{t}_{Neigh(i)}^{-i} \in \prod_{j \in Neigh(i)} T_j} p_i(\mathbf{t}_{Neigh(i)}^{-i} | t_i) \left( \sum_{\mathbf{t}_{X_i}^{-i} \in \prod_{j \in X_i} T_j} p_i(\mathbf{t}_{X_i}^{-i} | \mathbf{t}_{Neigh(i)}^{-i}, t_i) u_i(\mathbf{a}', \mathbf{t}) \right)$$

Since, the subexpression in parentheses evaluates to 1, the formula can be rewritten as:

$$v_{(i,t_i)}(\mathbf{a}) = \sum_{\mathbf{t}_{\text{Neigh}(i)}^{-i} \in \prod_{j \in \text{Neigh}(i)} T_j} p_i(\mathbf{t}_{\text{Neigh}(i)}^{-i} | t_i) u_i(\mathbf{a}', \mathbf{t})$$

On the other hand, from formula (2) we obtain, for every player  $j \in N$ :  $i(\mathcal{G}) \times \log \|\mathcal{G}\| \geq |\text{Neigh}(j)| \times \log |A_j|$ . Wlog, we can assume that, for each  $j \in N$ ,  $|A_j| \geq 2$ , hence  $\log |A_j| \geq 1$ . Thus, we have  $|\text{Neigh}(j)| \leq i(\mathcal{G}) \cdot \log \|\mathcal{G}\|$ .

We now show that, for every type-agent  $(i, t_i)$ , function  $v_{(i,t_i)}(\cdot)$  can be computed in polynomial time in  $\|\mathcal{G}\|$ . This is because, given an action-profile  $\mathbf{a}$ , computing  $v_{(i,t_i)}(\mathbf{a})$  accounts to sum out  $\prod_{j \in \text{Neigh}(i)} |T_j|$  factors (each one evaluable in polynomial time), and, as we observed above, we have that, for each  $j \in N$ , such value is polynomial in  $\|\mathcal{G}\|$  if the game has bounded intricacy and the number of types available to each player is bounded by a constant  $T$ .

Hence, we have shown how  $\mathcal{G}$  can be transformed in polynomial time into the game  $\mathcal{G}'$ . In order to conclude the proof, it remains to show that deciding whether Nash equilibria exist in  $\mathcal{G}'$  can also be made in polynomial time.

Theorem 5.3 of [Gottlob *et al.*, 2003] claims that this problem is polynomial for games with complete information which have bounded hypertree-width and bounded intricacy. What we show next is that the reduction above is such that  $\mathcal{G}'$  meets all these conditions.

**Bounded Intricacy:** Any type-agent  $(i, t_i)$  will have at most  $|\text{Neigh}(i)| \cdot T$  neighbors in  $\mathcal{G}'$ . Hence,  $i(\mathcal{G}') \leq i(\mathcal{G}) \cdot T$ .

**Bounded hypertree-width:** Given the hypergraph  $\mathcal{H}(\mathcal{G})$  associated to  $\mathcal{G}$ , with bounded hypertree-width  $hw(\mathcal{G})$ , we can build the hypergraph for  $\mathcal{G}'$  as follows: the nodes are the type-agents  $(i, t_i)$ , and for each type-agent  $(i, t_i)$ , an hyper-edge  $e_{(i,t_i)}^{\mathcal{G}'}$  exists connecting  $(i, t_i)$  to all the variables occurring in  $v_{(i,t_i)}(\cdot)$ , i.e., all type-agents  $(j, t_j)$  with  $j \in \text{Neigh}(i)$  and  $t_j \in T_j$ . Such a hypergraph has hypertree-width bounded by  $hw(\mathcal{G}) \cdot T$ . To show this, we present a procedure that computes an hyper-tree decomposition  $\langle T', \chi', \lambda' \rangle$  of  $\mathcal{G}'$  from an hyper-tree decomposition  $\langle T, \chi, \lambda \rangle$  of  $\mathcal{G}$  of minimal width. The transformation is as follows: (1)  $T' = T$ ; (2) For every node  $v \in T$ ,  $\lambda'(v) = \{e_{(i,t_i)}^{\mathcal{G}'} | e_i^{\mathcal{G}} \in \lambda(v) \wedge t_i \in T_i\}$ , with  $e_i^{\mathcal{G}}$  being the hyper-edge in  $\mathcal{H}(\mathcal{G})$  defined by  $\{i\} \cup \text{Neigh}(i)$ ; (3) For every node  $v \in T$ ,  $\chi'(v) = \{(i, t_i) | i \in \chi(v) \wedge t_i \in T_i\}$ ;

It can be seen that  $\langle T', \chi', \lambda' \rangle$  satisfies all the conditions to be an hypertree-decomposition of  $\mathcal{H}(\mathcal{G}')$ , and has width bounded by  $hw(\mathcal{G}) \cdot T$ .  $\square$

## Acknowledgments

This work was supported by a Royal Society Wolfson Research Merit Award, allowing G. Gottlob to invite G. Greco and T. Mancini to Oxford.

In addition, G. Greco's work was supported by M.I.U.R. under project "tocai.it: Tecnologie Orientate alla Conoscenza per Aggregazioni di Imprese in Internet".

## References

- [Chen and Deng, 2006] Xi Chen and Xiaotie Deng. Settling the complexity of 2-player Nash-Equilibrium. In *Proc. of FOCS 2006*. IEEE Comp. Society Press. To appear.
- [Conitzer and Sandholm, 2003] V. Conitzer and T. Sandholm. Complexity results about Nash equilibria. In *Proc. of IJCAI 2003*, pages 765–771. Morgan Kaufmann.
- [Daskalakis and Papadimitriou, 2005] K. Daskalakis and C. H. Papadimitriou. Three-player games are hard. *Electronic Colloquium on Comp. Complexity*, 12, 2005.
- [Feldmann *et al.*, 2003] R. Feldmann, M. Gairing, T. Lücking, B. Monien, and M. Rode. Selfish routing in non-cooperative networks: A survey. In *Proc. of MFCS 2003*, pages 21–45. Springer.
- [Gilboa and Zemel, 1989] I. Gilboa and E. Zemel. Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior*, 1:80–93, 1989.
- [Gottlob *et al.*, 2002] G. Gottlob, N. Leone, and F. Scarcello. Hypertree decompositions and tractable queries. *J. of Comp. and System Sciences*, 64(3):579–627, 2002.
- [Gottlob *et al.*, 2003] G. Gottlob, G. Greco, and F. Scarcello. Pure Nash equilibria: hard and easy games. In *Proc. of TARK 2003*, pages 215–230. ACM Press.
- [Harsanyi, 1968] J. Harsanyi. Games with incomplete information played by 'bayesian' players, parts i-iii. *Management Sci.*, 14:159–182, 320–334, 486–502, 1968.
- [Kearns *et al.*, 2001] M. Kearns, M. L. Littman, and S. Singh. Graphical models for Game Theory. In *Proc. of UAI 2001*, pages 253–260. Morgan Kaufmann.
- [Megiddo and Papadimitriou, 1991] N. Megiddo and C. H. Papadimitriou. On total functions, existence theorems and computational complexity. *Theoretical Comp. Science*, 81(2):317–324, 1991.
- [Myerson, 1997] R. B. Myerson. *Game Theory: Analysis of Conflict*. Harvard Univ. Press, 1997.
- [Nash, 1951] J. F. Nash. Non-cooperative games. *Annals of Math.*, 54(2):286–295, 1951.
- [Osborne and Rubinstein, 1994] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. The MIT Press, 1994.
- [Papadimitriou, 1994a] C. H. Papadimitriou. *Computational Complexity*. Addison Wesley Publ. Co., 1994.
- [Papadimitriou, 1994b] C. H. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *J. of Comp. and System Sciences*, 48(3):498–532, 1994.
- [Singh *et al.*, 2004] S. Singh, V. Soni, and M. P. Wellman. Computing approximate Bayes-Nash Equilibria in tree-games of incomplete information. In *Proc. of ACM-EC 2004*, pages 81–90. ACM Press.
- [Vickrey and Koller, 2002] D. Vickrey and D. Koller. Multi-agent algorithms for solving graphical games. In *Proc. of AAAI 2002*, pages 345–351. AAAI Press/The MIT Press.