ORIGINAL PAPER

Bilevel programming methods for computing single-leader-multi-follower equilibria in normal-form and polymatrix games

Nicola Basilico[1] · **Stefano Coniglio**[2] · **Nicola Gatti**[3] · **Alberto Marchesi**[3]

© The Author(s) 2019

Abstract

The concept of leader-follower (or Stackelberg) equilibrium plays a central role in a number of real-world applications bordering on mathematical optimization and game theory. While the single-follower case has been investigated since the inception of bilevel programming with the seminal work of von Stackelberg, results for the case with multiple followers are only sporadic and not many computationally affordable methods are available. In this work, we consider Stackelberg games with two or more followers who play a (pure or mixed) Nash equilibrium once the leader has committed to a (pure or mixed) strategy, focusing on normal-form and polymatrix games. As customary in bilevel programming, we address the two extreme cases where, if the leader's commitment originates more Nash equilibria in the followers' game, one which either maximizes (optimistic case) or minimizes (pessimistic case) the leader's utility is selected. First, we show that, in both cases and when assuming mixed strategies, the optimization problem associated with the search problem of finding a Stackelberg equilibrium is $\mathcal{NP}$-hard and not in Poly-$\mathcal{APX}$ unless $\mathcal{P} = \mathcal{NP}$. We then consider different situations based on whether the leader or the followers can play mixed strategies or are restricted to pure strategies only, proposing exact nonconvex mathematical programming formulations for the optimistic case for normal-form and polymatrix games. For the pessimistic problem, which cannot be tackled with a (single-level) mathematical programming formulation, we propose a heuristic black-box algorithm. All the methods and formulations that we propose are thoroughly evaluated computationally.

Extended author information available on the last page of the article

1 Introduction

Leader-follower (or *Stackelberg*) *games* model the interaction between rational agents (or players) when a hierarchical decision-making structure is in place. Considering, for simplicity, the two-player case, Stackelberg games model situations where an agent (the *leader*) plays first and a second agent (the *follower*) plays right after them, after observing the strategy the leader has chosen.

The number of real-world problems where a leader-follower (or Stackelberg) structure can be identified is extremely large. This is often the case in the security domain (An et al. 2011; Kiekintveld et al. 2009), where a defender, aiming to protect a set of valuable targets from the attackers, plays first, while the attackers, acting as followers, make their move only after observing the leader's defensive strategy. Other noteworthy cases are interdiction problems (Caprara et al. 2016; Matuschke et al. 2017), toll setting problems (Labbé and Violin 2016), network routing problems (Amaldi et al. 2013) and (singleton) congestion games (Castiglioni et al. 2018; Marchesi et al. 2018).

While, since the seminal work of von Stackelberg (2010), the case with a single leader and a single follower has been widely investigated, only a few results are known for the case with multiple followers and not many computationally affordable methods are available to solve the corresponding equilibrium-finding problem.

In this paper, we focus on the fundamental case of single-leader multi-follower games with a finite number of actions per player where the overall game can be represented as a normal-form or polymatrix game—the latter is of interest as it plays an important role in a number of applications such as in the security domain, where the defender may need to optimize against multiple uncoordinated attackers solely interested in damaging the leader. Throughout the paper, we assume the setting where the (two or more) followers play simultaneously in a noncooperative way, for which it is natural to assume that, after observing the leader's play (either as a strategy or as an action), the followers would reach a *Nash equilibrium* (NE) (see Shoham and Leyton-Brown (2008) for a thorough exposition of this equilibrium concept). We refer to an equilibrium in such games as *leader-follower Nash equilibrium* (LFNE).

As it is typical in bilevel programming, we study two extreme cases: the optimistic one where, if the leader's commitment originates more NE in the followers' game, one which maximizes the leader's utility is selected, and the pessimistic one where an equilibrium which minimizes the leader's utility is chosen.

In particular, the leader's utility at an optimistic equilibrium corresponds to the largest utility the leader may get assuming the best case in which the followers would (somehow) end up playing a Nash equilibrium which maximizes the leader's utility. Differently, the leader's utility at a pessimistic equilibrium corresponds to a utility value the leader could always get independently of the followers' behavior. From this perspective, a risk-taking leader would play according to an optimistic equilibrium, whereas a risk-averse leader would play according to a pessimistic equilibrium. For more types of solution concepts related to these two, we refer the reader to Alves and Antunes (2016).

The original contributions of our work are as follows.[1] First, we illustrate that the optimization problem associated with the search problem of computing an LFNE in mixed strategies when the followers play an NE which either maximizes or minimizes the leader's utility is $\mathcal{NP}$-hard and not in Poly-$\mathcal{APX}$ unless $\mathcal{P} = \mathcal{NP}$. After casting the general problem with mixed-strategy commitments in bilevel terms, we propose different nonlinear and nonconvex single-level mathematical programming formulations for the optimistic case, suitable for state-of-the-art spatial-branch-and-bound solvers. For the pessimistic case, which does not admit a single-level mathematical programming reformulation of polynomial size, we propose a heuristic method based on the combination of a spatial-branch-and-bound solver with a *black-box* algorithm. We also briefly investigate (easier) variants of the problem obtained when restricting either the leader or the followers to pure-strategy commitments. We conclude by providing a thorough experimental evaluation of our techniques on a (normal-form and polymatrix) test bed generated with GAMUT (Nudelman et al. 2004), also encompassing some structured games, employing different solvers: BARON, SCIP, CPLEX, SNOPT and RBFOpt. (The latter is used for black-box optimization).

2 Notation

Let $N = \{1, \ldots, n\}$ be the set of agents. For each $p \in N$, we denote by A_p the agent's set of actions, with $m_p = |A_p|$. For each agent $p \in N$, we denote by $x_p \in [0, 1]^{m_p}$, with $e^T x_p = 1$ (where e is the all-one vector), their *strategy vector* (or strategy, for short). Each component x_p^a of x_p represents the probability by which agent p plays action $a \in A_p$. We call x_p a vector of *pure strategies* if $x_p \in \{0, 1\}^{m_p}$ or of *mixed strategies* in the general case. We denote a *strategy profile*, i.e., the collection of the strategies each agent plays, by $x = (x_1, \ldots, x_n)$.

For each agent $p \in N$, we define their utility function as $u_p : [0, 1]^{m_1} \times \cdots \times [0, 1]^{m_n} \to \mathbb{R}$. A strategy profile $x = (x_1, \ldots, x_n)$ is an NE if and only if, for each agent $p \in N$, $u_p(x_1, \ldots, x_n) \geq u_p(x_1', \ldots, x_n')$ for every strategy profile x' where $x_q' = x_q$ for all $q \in N \setminus \{p\}$ and $x_p' \neq x_p$. (This corresponds to assuming that no unilateral deviations would take place.) We consider two game classes: *normal-form* (NF) and *polymatrix* (PM).

For NF games (see Shoham and Leyton-Brown (2008) for a reference), we let $U_p \in \mathbb{R}^{m_1 \times \cdots \times m_n}$ denote, for each agent $p \in N$, their (multi-dimensional) utility (or payoff) matrix where each component $U_p^{a_1, \ldots, a_n}$ denotes the utility of agent p when all the agents play actions $a_1, \ldots, a_n$. Given a strategy profile $(x_1, \ldots, x_n)$, the expected utility of agent $p \in N$ is equal to the multi-linear function $u_p(x_1, \ldots, x_n) = x_p^T(U_p \cdot \prod_{q \in N \setminus \{p\}} x_q)$.

For PM games (see Yanovskaya (1968) for a reference), we have a utility matrix $U_{pq} \in \mathbb{R}^{m_p \times m_q}$ per pair of agents $p, q \in A$. Given a strategy profile $(x_1, \ldots, x_n)$, the expected utility of agent p is equal to the bilinear function $u_p(x_1, \ldots, x_n) = \sum_{q \in N \setminus \{p\}} x_p^T U_{pq} x_q$.

---

[1] Some parts of the paper were presented at a preliminary stage in Basilico et al. (2016) and Basilico et al. (2017).

We remark that, while in the NF case the degree of the polynomial corresponding to an agent's expected utility is equal to the number of agents, it is always equal to 2 in the PM case, independently of the number of agents involved. The computational impact of this property will be discussed in the paper.

3 Previous works

Since the original work of Nash (1950), the problem of computing Nash equilibria in multi-player games (without a leader) has attracted a large interest—see the monograph (von Stengel 2010 and Chen and Deng 2006; Conitzer and Sandholm 2008) where the complexity of the problem is addressed. For more details on noncooperative game theory, we refer the interested reader to Shoham and Leyton-Brown (2008).

Most of the game-theoretic investigations on Stackelberg games have, to the best of our knowledge, mainly addressed the case of a single follower. In such setting, it is known that the single follower can play a pure strategy without loss of generality, i.e., that there always is a pure strategy by which they can maximize their utility and that the optimization problem associated with the search problem of computing an equilibrium is easy with complete information (von Stengel and Zamir 2010), while it becomes $\mathcal{NP}$-hard for Bayesian games (Conitzer and Sandholm 2006). Algorithms are proposed in Conitzer and Sandholm (2006).

For what concerns Stackelberg games with more than two players, some works have investigated the case with multiple leaders and a single follower; see Leyffer and Munson (2010). For the problem involving a single leader and multiple followers (the one on which we focus in this paper), only a few results are available. It is known, for instance, that an equilibrium can be found in polynomial time if the followers play a correlated equilibrium in the optimistic case (Conitzer and Korzhyk 2011) (see Shoham and Leyton-Brown 2008 for more detail on correlated equilibria), whereas the associated optimization problem is $\mathcal{NP}$-hard if they play sequentially one at a time (as in a classical Stackelberg game with many players) (Conitzer and Sandholm 2006).

4 Problem statements, bilevel perspective and computational complexity

In this section, we formalize the problem that we address in the paper, cast it in bilevel terms and investigate its computational complexity and approximability.

4.1 Problem statements

In formal terms, the two main versions of the problem of computing an LFNE that we tackle in this paper, optimistic (O-LFNE) and pessimistic (P-LFNE), are defined as follows:

O-LFNE: Given an n-agent game with $n \geq 3$, find a strategy vector δ for the leader such that, after committing, the largest leader's utility over all the NE in the followers' game parameterized by δ is as large as possible.

P-LFNE: Given an n-agent game with $n \geq 3$, find a strategy vector δ for the leader such that, after committing, the smallest leader's utility over all the NE in the followers' game parameterized by δ is as large as possible.

When notationally convenient, we will refer to an either optimistic or pessimistic LFNE as O/P-LFNE.

We will distinguish between the cases where either the leader or the followers are restricted or not to pure strategies, considering four cases: leader in mixed and followers in mixed (LMFM), leader in pure and followers in mixed (LPFM), leader in mixed and followers in pure (LMFP), and leader in pure and followers in pure (LPFP). In the general (mixed) case, we assume that the leader commits to a strategy, i.e., to a distribution of probability according to which they (the leader) select their action, and that, while the followers can observe the distribution chosen by the leader, they (the followers) cannot observe its realization (i.e., the action the leader plays). This is the case in, e.g., security games. The case in which the leader's strategy is pure is the converse one in which the leader's play is completely observable by the followers. The assumption behind the followers playing mixed strategies is the same as in games without a leader (e.g., one can consider repeated games in which the leader has to commit to a single strategy before the game starts, whereas the followers can, at each iteration, draw a different action profile from their distribution of choice, thus playing mixed strategies).

For the sake of presentation, in the remainder of the paper we assume $n = 3$ (one leader, two followers). We remark that our results can be adapted to any n. In Sect. 9, we will indeed report on computational experiments carried out for games with more than two followers.

In the remainder of the paper, we assume that the last agent (the third), whom we relabel as agent ℓ, takes the role of leader. All the other agents (the followers) are compactly denoted by the set $F = N \setminus \{\ell\}$. When $n = 3$, $F = \{1, 2\}$. For all $f \in F$, we define $f' := F \setminus \{f\}$. We also denote x_ℓ (the strategy vector of the leader) by δ and x_1, x_2 (the strategy vectors of the followers) by ρ_1, ρ_2. For each $p \in N$, we let Δ_p be the *simplex of strategies* of player p, i.e., the set of nonnegative vectors δ, ρ_1 or ρ_2 summing to 1.

4.2 Bilevel programming perspective

Computing an O/P-LFNE amounts to solving a *bilevel programming problem*.

In the optimistic case, we can compute an O-LFNE by solving the following problem:

$$\text{(O-LFNE)} \max_{\substack{(\rho_1, \rho_2, \delta) \in \\ \Delta_1 \times \Delta_2 \times \Delta_\ell}} \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_\ell^{ijk} \rho_1^i \rho_2^j \delta^k \tag{1a}$$

$$\text{s.t.} \quad \rho_1 \in \underset{\rho_1 \in \Delta_1}{\mathrm{argmax}} \left\{ \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_1^i \rho_2^j \delta^k \right\} \tag{1b}$$

$$\rho_2 \in \underset{\rho_2 \in \Delta_2}{\mathrm{argmax}} \left\{ \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \rho_2^j \delta^k \right\}. \tag{1c}$$

Due to Constraints (1b)–(1c), the second-level problems call for a pair (ρ_1, ρ_2) of followers' strategies forming an NE in the followers' game induced by the strategy $\delta \in \Delta_\ell$ chosen by the leader in the first level. Note that, due to the definition of NE, the pair (ρ_1, ρ_2) is an NE in the game induced by δ if and only if ρ_1 (resp., ρ_2) maximizes player 1's (resp., player 2's) utility when assuming that player 2 (resp., player 1) plays ρ_2 (resp., ρ_1). Subject to these constraints, the first level calls for a triple (ρ_1, ρ_2, δ) maximizing the leader's utility.

The problem is optimistic as, assuming that the second level admits many NE (ρ_1, ρ_2) for the chosen δ, it calls for a pair (ρ_1, ρ_2) which, together with δ, maximizes the leader's utility. Notice that, while any triple $(\rho_1, \rho_2, \delta) \in \Delta_1 \times \Delta_2 \times \Delta_\ell$ is a *feasible* solution to the problem as long as the pair (ρ_1, ρ_2) is an NE in the game induced by δ, Problem (1a)–(1c) calls for a triple (ρ_1, ρ_2, δ) which is *optimal*—as, if not, the leader would prefer to change their strategy and (ρ_1, ρ_2, δ) would not be a LFNE.

In the pessimistic case, computing a P-LFNE amounts to solving to the following problem:

$$\text{(P-LFNE)} \max_{\delta \in \Delta_\ell} \min_{\substack{(\rho_1, \rho_2) \in \\ \Delta_1 \times \Delta_2}} \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_\ell^{ijk} \rho_1^i \rho_2^j \delta^k \tag{2a}$$

$$\text{s.t.} \quad \text{Constraints (1b), (1c).} \tag{2b}$$

This problem differs from its optimistic counterpart as, due to the assumption of pessimism, the leader here maximizes the *minimum* value taken by their utility over all pairs (ρ_1, ρ_2) which are NE in the followers' game induced by δ—that is, for the chosen δ, ρ_1 and ρ_2 always correspond to a NE which minimizes the leader's utility.

4.3 Complexity results

As we will show, the optimization problem associated with the search problem of computing an LFNE is both $\mathcal{NP}$-hard and inapproximable in both versions (O-LFNE and P-LFNE) in the LMFM case even with a single leader action (which implies that the result also holds for the LPFM case). This follows from the $\mathcal{NP}$-hardness and inapproximability of the problem of computing, in a two-player game, a mixed-strategy NE which maximizes the sum of the players' utilities (the so-called *social welfare*) (Conitzer and Sandholm 2008):

Proposition 1 (Conitzer and Sandholm 2008) *The problem of computing a mixed-strategy NE which maximizes the total players' utility is $\mathcal{NP}$-hard and it is not in $\mathcal{APX}$ unless $\mathcal{P} = \mathcal{NP}$, even when the game is polymatrix.*

The result is based on the fact that, for any SAT instance, it is possible to build a symmetric two-player game (U_1, U_2), either NF or PM, such that:

(i) there is a (pure-strategy) NE in which both players play their last action and receive a utility equal to $\epsilon > 0$, where ϵ is an arbitrarily small constant;

(ii) the game admits a (mixed-strategy) NE providing each player with a utility of m, where m is the number of actions, if and only if the SAT instance is a YES instance.

This implies that, in any such game, finding an NE where the players achieve a utility strictly larger than ϵ would suffice to claim that the corresponding SAT instance is a YES instance. It follows that one cannot decide in polynomial time whether such games admit an NE providing the players with a utility strictly larger than ϵ unless $\mathcal{P} = \mathcal{NP}$ as, if that were the case, YES instances of SAT could be decided in polynomial time. This also shows that finding an NE which maximizes the social welfare (defined as the total players' utility) is not in $\mathcal{APX}$. This is because the existence of an NE providing the players with a total utility strictly greater than 2ϵ would suffice to conclude that the corresponding SAT instance admits answer YES.

We show that the result in Conitzer and Sandholm (2008) can be strengthened with a simple observation:

Proposition 2 *The problem of computing a mixed-strategy NE which maximizes the total players' utility is not in Poly-$\mathcal{APX}$ unless $\mathcal{P} = \mathcal{NP}$, even when the game is polymatrix.*

Proof Let $\epsilon = \frac{1}{2^m}$. On games corresponding to YES SAT instances (which admit an NE with total utility $2m$), an algorithm with approximation ratio $\frac{1}{\alpha}$ would yield an NE of total utility at least $\frac{1}{\alpha} 2m$. Note that, if $\frac{1}{\alpha} 2m > 2\epsilon$ (i.e., $\frac{1}{\alpha} > \frac{\epsilon}{m}$), the SAT instance is proved to be a YES instance. Therefore, there cannot be a polynomial time approximation algorithm with a factor better than $\frac{\epsilon}{m} = \frac{1}{2^m m}$ unless $\mathcal{P} = \mathcal{NP}$. Since the reciprocal of this factor is superpolynomial, the problem is not in Poly-$\mathcal{APX}$. □

For the problem of computing an O/P-LFNE, we show the following result:

Proposition 3 *The optimization problem associated with the search problem of computing an O/P-LFNE in the LMFM and LPFM cases is $\mathcal{NP}$-hard and it is not in Poly-$\mathcal{APX}$ unless $\mathcal{P} = \mathcal{NP}$, even when the game is polymatrix.*

Proof Let us consider the O-LFNE case first. Given a game with utilities (U_1, U_2) and m actions per player as defined in Conitzer and Sandholm (2008), we construct a 3-player leader-follower polymatrix game where:

– the leader only has one action and utility matrices $U_{\ell f_1} = U_{\ell f_2} = \left[1, \ldots, 1, \frac{1}{2^m}\right]$;

– player f_1's utility matrices are $U_{f_1 \ell} = \mathbf{0}$ and $U_{f_1 f_2} = U_1$;

– player f_2's utility matrices are $U_{f_2 \ell} = \mathbf{0}$ and $U_{f_2 f_1} = U_2$.

Due to having a single action, the presence of the leader is immaterial. (Note that, therefore, the LMFM and LPFM cases coincide.) Therefore, the set of followers' equilibria in the leader-follower game is the same as that of the original two-player game. It follows that SAT has answer YES if and only if the leader-follower game admits an equilibrium with leader's utility strictly larger than $\frac{1}{2^m}$, as that corresponds

to an NE in the followers' game with utility strictly larger than ϵ for each player. Along the lines of the previous proof, an algorithm with approximation factor $\frac{1}{\alpha}$ would yield, for a YES instance, a leader utility of at least $\frac{1}{\alpha}$, allowing us to conclude that the instance is a YES instance if $\frac{1}{\alpha} > \frac{1}{2^m}$. This shows that the problem of computing an O-LFNE is not in Poly-$\mathcal{APX}$ unless $\mathcal{P} = \mathcal{NP}$ (even in polymatrix games).

For the computation of a P-LFNE, the reasoning is the same except for defining $U_{\ell f_1} = U_{\ell f_2} = \left[\frac{1}{2^m}, \frac{1}{2^m}, \ldots, \frac{1}{2^m}, 1\right]$. □

We conclude the section by showing that deciding whether one of the leader's actions can be safely discarded is a hard problem, which implies that dominance-like techniques often used in game theory to reduce the search space of an equilibrium-computing algorithm are inapplicable.

Proposition 4 *In the LMFM case, deciding whether an action of the leader is played with strictly positive probability at an O/P-LFNE is $\mathcal{NP}$-hard.*

Proof Given a symmetric two-player game (U_1, U_2) with m actions as defined in Conitzer and Sandholm (2008), we build a three-player game $(U_\ell, U_{f_1}, U_{f_2})$ in which:

- the leader has two actions, while f_1 and f_2 have m actions each;
- when the leader plays their first action, the payoffs of all the players are 1/4;
- when the leader plays their second action, the payoffs of f_1 and f_2 are those in (U_1, U_2) and the leader's payoffs are 1 for all the actions of f_1 and f_2, except for the combination composed of the last action of f_1 and the last action of f_2, in which the leader's payoff is 0.

We show that the first action of the leader can be safely discarded from the game $(U_\ell, U_{f_1}, U_{f_2})$ if and only if the game (U_1, U_2) admits a mixed-strategy NE providing the players with a utility of m, which implies that deciding whether the first action of the leader can be discarded is $\mathcal{NP}$-hard. If the leader plays their first action, they receive a utility of 1/4. If the leader plays their second action, the followers play the best NE for the leader, which can be either (i) the pure-strategy NE in which both play their last action providing the leader with a utility of 0 or (ii) if it exists, the mixed-strategy NE providing the leader with a utility of 1. For any mixed strategy of the leader, the behavior of the followers does not change w.r.t. the case in which the leader plays their second action as a pure strategy. This is because, when the leader randomizes between their two actions, the utility of the followers f_1 and f_2 is an affine transformation (with positive coefficients) of U_1 and U_2, making them play exactly as in the case where the leader plays their second action as a pure strategy. Thus, at an optimistic LFNE the leader plays a pure strategy, playing their first action if (U_1, U_2) does not admit a mixed-strategy NE and their second action if it does. The first action of the leader can therefore be safely discarded if and only if (U_1, U_2) admits a mixed-strategy NE providing the players with a utility of m.

The proof is analogous in the pessimistic case after interchanging the leader payoffs of values 0 and 1. □

5 Optimistic case with leader in mixed and followers in mixed (O-LMFM)

In this section, we focus on the optimistic setting in the general case where each player is allowed to play mixed strategies. We propose three different exact mathematical programming formulations for NF games and then illustrate how they can be simplified for PM games.

5.1 Exact formulations for NF games

We report the three formulations illustrating how to derive each of them in sequence.

5.1.1 O-NF-LMFM-I

To obtain a single-level formulation for the problem, we proceed by applying a standard reformulation (Shoham and Leyton-Brown 2008) involving complementarity constraints.

Let, for all $i \in A_1$ and $j \in A_2$, $\tilde{U}_1^{ij} := \sum_{k \in A_\ell} U_1^{ijk} \delta^k$ and $\tilde{U}_2^{ij} = \sum_{k \in A_\ell} U_2^{ijk} \delta^k$ be the matrices of the followers' game, parameterized by δ. According to Constraint (1b), for (ρ_1, ρ_2) to be a NE ρ_1 must be an optimal solution to the Linear Program (LP):

$$\max_{\rho_1 \in \Delta_1} \left\{ \sum_{i \in A_1} \sum_{j \in A_2} \tilde{U}_1^{ij} \rho_1^i \rho_2^j \right\},$$

where $\tilde{U}_1^{ij} \rho_1^i \rho_2^j$ is a linear function of ρ_1 if ρ_2 is fixed. Since the LP is feasible and bounded for any $\rho_2 \in \Delta_2$, by complementary slackness we have that $\rho_1 \in \Delta_1$ is optimal if and only if there is a scalar v_1 such that the following holds for all $i \in A_1$:

$$\left(v_1 - \sum_{j \in A_2} \tilde{U}_1^{ij} \rho_2^j \right) \rho_1^i = 0$$

$$v_1 \geq \sum_{j \in A_2} \tilde{U}_1^{ij} \rho_2^j.$$

v_1 can be interpreted as the *best-response value* of follower 1, equal to the largest utility the follower can achieve at an equilibrium. Applying a similar reasoning to ρ_2, we obtain that $\rho_2 \in \Delta_2$ is optimal if and only if there is a scalar v_2 such that the following holds for all $j \in A_2$:

$$\left(v_2 - \sum_{i \in A_1} \tilde{U}_2^{ij} \rho_1^i \right) \rho_2^j = 0$$

$$v_2 \geq \sum_{i \in A_1} \tilde{U}_2^{ij} \rho_1^i.$$

We conclude that (ρ_1, ρ_2) is an NE if and only if there are $v_1, v_2 \geq 0$ such that ρ_1 and ρ_2 simultaneously satisfy these four conditions.

After substituting for $\tilde{U}_1$ and $\tilde{U}_2$ their linear expressions in δ, we obtain the following constraints for player 1 and for all $i \in A_1$:

$$\left(v_1 - \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \delta^k \right) \rho_1^i = 0$$

$$v_1 \geq \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \delta^k.$$

For player 2 and for all $j \in A_2$, we obtain:

$$\left(v_2 - \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \delta^k \right) \rho_2^j = 0$$

$$v_2 \geq \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \delta^k.$$

By imposing such constraints *in lieu* of the two second-level argmax constraints of Problem (1) (Constraints (1b)–(1c)), we obtain a continuous single-level formulation with nonconvex trilinear terms.[2] Overall, the formulation reads:

$$\max_{\rho_1, \rho_2, \delta, v} \quad \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_\ell^{ijk} \rho_1^i \rho_2^j \delta^k \qquad (3)$$

$$\text{s.t.} \left(v_1 - \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \delta^k \right) \rho_1^i = 0 \qquad \forall i \in A_1 \qquad (4)$$

$$v_1 \geq \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \delta^k \qquad \forall i \in A_1 \qquad (5)$$

$$\left(v_2 - \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \delta^k \right) \rho_2^j = 0 \qquad \forall j \in A_2 \qquad (6)$$

$$v_2 \geq \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \delta^k \qquad \forall j \in A_2 \qquad (7)$$

$$\sum_{k \in A_\ell} \delta^k = 1, \delta \geq 0 \qquad (8)$$

$$\sum_{i \in A_f} \rho_f^i = 1, \rho_f \geq 0 \qquad \forall f \in F \qquad (9)$$

$$v_f \geq 0 \qquad f \in F. \qquad (10)$$

The problem contains $m_1 + m_2$ cubic constraints, $m_1 + m_2$ quadratic constraints and a cubic objective function.

---

[2] Note that strong duality can be employed in place of complementary slackness. Preliminary experiments suggest that the second option is computationally preferable.

5.1.2 O-NF-LMFM-II

What we propose now is aimed at achieving a formulation which can be solved more efficiently. Since each term of the complementarity constraints we introduced is bounded from above and below, we can apply a simple reformulation along the lines of Sandholm et al. (2005). Let $s_1 \in \{0, 1\}^{m_1}$ and $s_2 \in \{0, 1\}^{m_2}$ be the *antisupport vectors* of ρ_1 and ρ_2, (i.e., two binary vectors with m_1 and, respectively, m_2 components each of which has value 0 if and only if ρ_1 and, respectively, ρ_2 is strictly positive in that component). It suffices to impose the following constraints for all $i \in A_1$:

$$\rho_1^i \leq 1 - s_1^i$$

$$v_1 - \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \delta^k \leq M s_1^i$$

and the following ones for all $j \in A_2$:

$$\rho_2^j \leq 1 - s_2^j$$

$$v_2 - \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \delta^k \leq M s_2^j.$$

M is an upper bound on the entries of U_1, U_2. This way, while still retaining the original trilinear objective function only bilinear constraints are needed.

We obtain the following reformulation:

$$\max_{\rho_1, \rho_2, \delta, v, s} \quad \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_\ell^{ijk} \rho_1^i \rho_2^j \delta^k \tag{11}$$

$$\text{s.t.} \quad v_1 - \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \delta^k \leq M s_1^i \quad \forall i \in A_1 \tag{12}$$

$$v_2 - \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \delta^k \leq M s_2^j \quad \forall j \in A_2 \tag{13}$$

$$\rho_f^i \leq 1 - s_f^i \quad \forall f \in F, i \in A_f \tag{14}$$

$$s_f^j \in \{0, 1\} \quad \forall f \in F, j \in A_f \tag{15}$$

$$\text{Constraints (5), (7), (8)–(10).} \tag{16}$$

At the cost of introducing binary variables, with this formulation we achieve fewer nonlinearities: only $2m_1 + 2m_2$ quadratic constraints and a cubic objective function.

5.1.3 O-NF-LMFM-III

Ultimately, we aim to solve the problem with spatial-branch-and-bound techniques, such as those implemented in BARON and SCIP. The main strategy of such methods

to handle nonlinearities is to isolate "simple" nonlinear terms (bilinear or trilinear in our case) by shifting them into a new (so-called *defining*) constraint to which a convex envelope is applied.

We propose to anticipate this reformulation, so to be able to derive some valid constraints. First, we introduce:

(i) variable y_2^{jk} and constraint $y_2^{jk} = \rho_2^j \delta^k$ for all $j \in A_2, k \in A_\ell$,
(ii) variable y_1^{ik} and constraint $y_1^{ik} = \rho_1^i \delta^k$ for all $i \in A_1, k \in A_\ell$,
(iii) variable z^{ijk} and constraint $z^{ijk} = \rho_1^i y_2^{jk}$ for all $i \in A_1, j \in A_2, k \in A_\ell$.

By substituting each bilinear and trilinear term with the newly introduced variables, we then obtain a formulation which is linear everywhere, except for the defining constraints themselves.

The advantage of carrying out this reformulation step *a priori* is that we can now observe that, after introducing the new variables, the matrix $\{y_2^{jk}\}_{jk \in A_2 \times A_\ell}$ is, by definition, the outer product of the stochastic vectors ρ_2 and δ and, as such, is a stochastic matrix itself. The same holds for the tensor $\{z^{ijk}\}_{ijk \in A_1 \times A_2 \times A_\ell}$, which is the outer product of the vectors ρ_1, ρ_2, δ and, as such, is a stochastic tensor. This implies the validity of the following three constraints:

$$\sum_{i \in A_1} \sum_{k \in A_\ell} y_1^{ik} = 1$$

$$\sum_{j \in A_2} \sum_{k \in A_\ell} y_2^{jk} = 1$$

$$\sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} z^{ijk} = 1.$$

We remark that these inequalities are a subset of those that are obtained by applying a relaxation-linearization technique *à la* Sherali and Adams (1990) to Constraints (8) and (9).

The formulation that we obtain is the following one:

$$\max_{\rho_1, \rho_2, \delta, v, s, y, z} \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_\ell^{ijk} z^{ijk} \tag{17}$$

$$\text{s.t.} \quad v_1 \geq \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} y_2^{jk} \quad \forall i \in A_1 \tag{18}$$

$$v_2 \geq \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} y_1^{ij} \quad \forall j \in A_2 \tag{19}$$

$$v_1 - \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} y_2^{jk} \leq M s_1^i \quad \forall i \in A_1 \tag{20}$$

$$v_2 - \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} y_1^{ik} \leq M s_2^j \quad \forall j \in A_2 \tag{21}$$

$$y_f^{ik} = \rho_f^i \delta^k \quad \forall k \in A_\ell, f \in F, i \in A_f \tag{22}$$

$$z^{ijk} = \rho_1^i y_2^{jk} \quad k \in A_\ell, i \in A_1, j \in A_2 \tag{23}$$

$$\sum_{i \in A_f} \sum_{k \in A_\ell} y_f^{ik} = 1 \quad \forall f \in F \tag{24}$$

$$\sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} z^{ijk} = 1 \tag{25}$$

$$y_f^{ij} \geq 0 \quad \forall f \in F, i \in A_f, k \in A_\ell \tag{26}$$

$$z^{ijk} \geq 0 \quad \forall i \in A_1, j \in A_2, k \in A_\ell \tag{27}$$

$$\text{Constraints}(8)\text{–}(10), (14)\text{–}(15). \tag{28}$$

Overall, we obtain $m_\ell(m_1 + m_2) + m_\ell m_1 m_2$ quadratic constraints and a linear objective function, yielding a tighter formulation than O-NF-LMFM-II, as we will show computationally.

5.2 Exact formulations for PM games

We illustrate how the three formulations we proposed can be substantially simplified for PM games.

5.2.1 O-PM-LMFM-I

In PM games, the expected utility for follower 1 corresponding to an action $i \in A_1$ (which is a trilinear function for NF games with $n = 3$ and of order n in general) is defined as the following linear function (which is linear for any n and, in particular, for $n = 3$):

$$\sum_{k \in A_\ell} U_{1\ell}^{ik} \delta^k + \sum_{j \in A_2} U_{12}^{ij} \rho_2^j.$$

The leader's utility is the following function, bilinear for any n:

$$\sum_{k \in A_\ell} \sum_{i \in A_1} U_{\ell 1}^{ik} \rho_1^i \delta^k + \sum_{k \in A_\ell} \sum_{j \in A_2} U_{\ell 2}^{jk} \rho_2^j \delta^k.$$

As a consequence, the PM counterpart to formulation O-NF-LMFM-I reads:

$$\max_{\rho_1, \rho_2, \delta, v} \sum_{k \in A_\ell} \sum_{i \in A_1} U_{\ell 1}^{ik} \rho_1^i \delta^k + \sum_{k \in A_\ell} \sum_{j \in A_2} U_{\ell 2}^{jk} \rho_2^j \delta^k \tag{29}$$

$$\text{s.t.} \quad v_1 \geq \sum_{k \in A_\ell} U_{1\ell}^{ik} \delta^k + \sum_{j \in A_2} U_{12}^{ij} \rho_2^j \quad \forall i \in A_1 \tag{30}$$

$$v_2 \geq \sum_{k \in A_\ell} U_{2\ell}^{jk} \delta^k + \sum_{i \in A_1} U_{21}^{ij} \rho_1^i \quad \forall j \in A_2 \tag{31}$$

$$\left(v_1 - \sum_{k \in A_\ell} U_{1\ell}^{ik} \delta^k + \sum_{j \in A_2} U_{12}^{ij} \rho_2^j \right) \rho_1^i = 0 \quad \forall i \in A_1 \tag{32}$$

$$\left(v_2 - \sum_{k \in A_\ell} U_{2\ell}^{jk} \delta^k + \sum_{i \in A_1} U_{21}^{ij} \rho_1^i \right) \rho_2^j = 0 \quad \forall j \in A_2 \tag{33}$$

$$\text{Constraints (8)–(10).} \tag{34}$$

Differently from the NF case, this formulation only contains $m_1 + m_2$ quadratic constraints and a quadratic objective (as Constraints (5) and (7) become linear here, while Constraints (4) and (6) and Objective (3) become quadratic).

5.2.2 O-PM-LMFM-II

Applying for the PM case the same reformulation we carried out in O-NF-LMFM-II, we obtain:

$$\max_{\rho_1, \rho_2, \delta, v, s} \sum_{k \in A_\ell} \sum_{i \in A_1} U_{\ell 1}^{ik} \rho_1^i \delta^k + \sum_{k \in A_\ell} \sum_{j \in A_2} U_{\ell 2}^{jk} \rho_2^j \delta^k \tag{35}$$

$$\text{s.t.} \quad v_1 - \sum_{k \in A_\ell} U_{1\ell}^{ik} \delta^k + \sum_{j \in A_2} U_{12}^{ij} \rho_2^j \leq M s_1^i \quad \forall i \in A_1 \tag{36}$$

$$v_2 - \sum_{k \in A_\ell} U_{2\ell}^{jk} \delta^k + \sum_{i \in A_1} U_{21}^{ij} \rho_1^i \leq M s_2^j \quad \forall j \in A_2 \tag{37}$$

$$\text{Constraints (8)–(10), (14)–(15), (30)–(31).} \tag{38}$$

Besides the binary variables, this formulation contains only linear constraints and a quadratic objective.

5.2.3 O-PM-LMFM-III

Similarly to O-NF-LMFM-III, this formulation is derived by reformulating each multilinear term in O-PM-LMFM-II. In the latter, the only nonlinearity is in the objective function. Therefore, O-PM-LMFM-III is obtained by just reformulating the products $\delta^i \rho_f^j$ it contains for all $f \in F$ and $j \in A_f$, adding valid constraints identical to those we added to O-NF-LMFM-III. We obtain:

$$\max_{\rho_1, \rho_2, \delta, v, s, y} \sum_{k \in A_\ell} \sum_{i \in A_1} U_{\ell 1}^{ik} y_1^{ik} + \sum_{k \in A_\ell} \sum_{j \in A_2} U_{\ell 2}^{jk} y_2^{jk} \tag{39}$$

$$\text{s.t.} \quad \text{Constraints (8)–(10), (14)–(15), (22), (24), (26), (30)–(31), (36)–(37).} \tag{40}$$

Similarly to O-NF-LMFM-III, O-PM-LMFM-III is completely linear except for the $m_\ell(m_1 + m_2)$ defining quadratic Constraints (22).

6 Pessimistic case with leader in mixed and followers in mixed (P-LMFM)

Unless $\mathcal{P} = \mathcal{NP}$, it is clear that there is no single-level formulation of polynomial size (in terms of variables and constraints) for the problem of computing a pessimistic LFNE. This is because, given a triple δ, ρ_1, ρ_2, a single-level reformulation of polynomial size for the problem would allow for checking whether, for the given δ, the (ρ_1, ρ_2) pair yields not just an NE (this can be checked in polynomial time by inspecting polynomially many constraints) but an *optimal one*. That is, it would allow us to verify *in polynomial time* whether a given solution to an $\mathcal{NP}$-hard problem is optimal, which cannot be done in general unless $\mathcal{P} = \mathcal{NP}$.

For this reason, we adopt a different approach here, designing a heuristic method to tackle the pessimistic case based on a black-box solver coupled with an exact oracle. While the method is conceived to tackle the pessimistic case, it can also be used for the optimistic one (as we show in Computational results section).

The method is based on a radial basis function (RBF) estimation which relies on the solver RBFOpt (Costa et al. 2015). The idea is of exploring the leader's strategy space (variables δ) with a direct search which iteratively builds an RBF approximation of the objective function relying on the solution of an *oracle formulation* which is responsible for carrying out the objective function evaluation.

Given any incumbent value $\hat{\delta}$, the oracle solves the (NF or PM) second-level problem exactly after imposing $\delta = \hat{\delta}$. For NF games, the oracle formulation we use is similar to O-NF-LMFM-III, employing a different reformulation with auxiliary variables $y^{jk} = \rho_1^j \rho_2^k$, which yields a tighter reformulation than the original one in O-NF-LMFM-III when δ is given (as in this case). Crucially, in this formulation the sign of the objective function has to be changed so to produce a pair (ρ_1, ρ_2) which minimizes the leader's objective function (rather than maximizing it) for the given $\delta = \hat{\delta}$.

The oracle formulation for the optimistic and pessimistic cases reads as follows ($\pm$ indicates that the sign of the objective function has to be flipped from $+$ to $-$ in the pessimistic case):

$$\max_{\rho_1, \rho_2, v, s, y} \quad \pm \sum_{i \in A_1} \sum_{j \in A_2} \sum_{k \in A_\ell} U_\ell^{ijk} y^{ij} \hat{\delta}^k \tag{41}$$

$$\text{s.t.} \quad v_1 \geq \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \hat{\delta}^k \quad \forall i \in A_1 \tag{42}$$

$$v_2 \geq \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \hat{\delta}^k \quad \forall j \in A_2 \tag{43}$$

$$v_1 - \sum_{j \in A_2} \sum_{k \in A_\ell} U_1^{ijk} \rho_2^j \hat{\delta}^k \leq M s_1^i \quad \forall i \in A_1 \tag{44}$$

$$v_2 - \sum_{i \in A_1} \sum_{k \in A_\ell} U_2^{ijk} \rho_1^i \hat{\delta}^k \leq M s_2^j \quad \forall j \in A_2 \tag{45}$$

$$y^{ij} = \rho_1^i \rho_2^j \quad \forall i \in A_1, j \in A_2 \tag{46}$$

$$\sum_{i \in A_1} \sum_{j \in A_2} y^{ij} = 1 \tag{47}$$

$$y_f^{ij} \geq 0 \quad \forall f \in F, i \in A_f, k \in A_\ell \tag{48}$$

$$\text{Constraints (8)–(10), (14)–(15).} \tag{49}$$

Besides the defining constraints for y^{ij}, the other parts of the formulation are all linear.

For PM games, we can directly use formulation O-PM-LMFM-II: Since each of the nonlinear terms in O-PM-LMFM-II is bilinear and it involves δ, when δ is fixed to $\hat{\delta}$ the formulation corresponds to a mixed-integer linear program (MILP).

7 Optimistic case with leader in pure and followers in mixed (O-LPFM)

We focus now on the case in which the leader is restricted to pure strategies.

7.1 Exact formulations for NF and PF games

As it is clear, in the LPFM case the problem can be solved by using one of the formulations we proposed after imposing $\delta \in \{0, 1\}^{m_\ell}$. With a binary δ, though, we can obtain different formulations which contain fewer nonlinearities. We present them here for the NF and PM cases. We only consider the formulations denoted by III since they turn out to be easier to solve in practice (as we will see in Computational results section).

7.1.1 O-NF-LPFM-III

For $\delta \in \{0, 1\}^{m_\ell}$, the quadratic defining Constraints (22) in O-NF-LMFM-III can be dropped in favor of the following three linear constraints:

$$y_f^{ik} \leq \delta^k \quad \forall k \in A_\ell, f \in F, i \in A_f \tag{50}$$

$$y_f^{ik} \leq \rho_f^i \quad \forall k \in A_\ell, f \in F, i \in A_f \tag{51}$$

$$y_f^{ik} \geq \delta^k + \rho_f^i - 1 \quad \forall k \in A_\ell, f \in F, i \in A_f. \tag{52}$$

Together with $y_f^{ik} \geq 0$, these constraints constitute the so-called McCormick envelope (McCormick 1976) of the set $\{(y_f^{ik}, \delta^k, \rho_f^i) \in [0, 1]^3 : y_f^{ik} = \delta^k \rho_f^i\}$. When either $\delta^k \in \{0, 1\}$ or $\rho_f^i \in \{0, 1\}$, the envelope yields an exact reformulation (Al-Khayyal and Falk 1983). The resulting formulation is obtained from O-NF-LMFM-III by dropping the quadratic (defining) Constraints (22) and substituting for them the linear Constraints (50)–(52). The only nonlinear constraints still present in the formulation are Constraints (23).

7.1.2 O-PM-LPFM-III

In O-PM-LMFM-III, the only nonlinearities are due to the quadratic (defining) Constraints (22). Due to $\delta \in \{0, 1\}^{m_\ell}$, by applying the McCormick envelope via Constraints (50)–(52) we can remove all the nonlinearities from the problem, obtaining an MILP.

7.2 O-NF/PM-LPFM-implicit-enumeration

When $\delta \in \{0, 1\}^{m_\ell}$, an LFNE can also be found by solving m_ℓ times one of our formulations. It suffices to change the sign of the objective function in the pessimistic case, iteratively fixing $\delta = e_k$ (where e_k is the all zero vector with a single 1 in position k) and selecting the best outcome over all the iterations as the solution to the problem. While this method is correct for both variants (optimistic and pessimistic), in the optimistic case we can design a better algorithm, which we now introduce.

The main idea of the algorithm is pruning the search space A_ℓ so to solve fewer subproblems thanks to a bounding technique. For each of the leader's actions, we compute the utility they would obtain if the followers played a *correlated equilibrium* (CE) (which can be computed in polynomial time via linear programming; see Shoham and Leyton-Brown (2008)). Since the set of correlated strategies is a (strict) superset of that of mixed strategies, its computation yields an upper bound (UB). We can thus iterate over $i \in A_\ell$ and solve one of our formulations with $\delta = e_k$ (where e_k is the unit vector with a single 1 in position k) only if the UB with $\delta = e_k$ is better than the best solution found thus far.

The algorithm reads:

```
1: for k ∈ A_ℓ do
2:     UB(k) = BestCorrelatedEquilibrium(k)
3: end for
4: A_ℓ = DescendingSort(A_ℓ, UB)
5: LB = −∞
6: for k ∈ A_ℓ and UB(k) > LB do
7:     LB = max{LB, Utility(e_k)}
8: end for
```

BestCorrelatedEquilibrium(k) computes a UB with $\delta = e_k$ by computing a CE in polynomial time via linear programming, along the lines of Shoham and Leyton-Brown (2008). After sorting the leader's actions in decreasing order of UB via *DescendingSort*(A_ℓ, UB), the algorithm iterates over A_ℓ, computing with *Utility*(e_k) the exact leader's utility corresponding to playing the pure action $\delta = e_k$ only if $UB(k)$ is sufficiently promising. In our implementation, *Utility*(e_k) solves the same oracle formulations adopted in the black-box method.

8 A note on solution approaches for the remaining cases

For completeness, in this section we address the remaining cases that are obtained by restricting either the leader or the followers to pure strategies. Since all these cases can be solved fairly easily with only one exception, we will not consider them in Computational results section.

8.1 O/P-LFNE with leader in pure and followers in pure (O/P-LPFP)

The case where both the leader and the followers can only play pure strategies is trivial in both the optimistic and pessimistic versions. For its solution, one can, first, construct each of the m^3 possible outcomes of the three players and, then, discard all the outcomes where the pair of followers' strategies do not induce an NE for the pure leader strategy they contain. For the optimistic case, it then suffices to compare the leader's utility corresponding to all the outcomes which have not been discarded, identifying one where the leader's utility is maximized. For the pessimistic case, an extra step is needed as one has to, first, group all the outcomes by leader strategy and then identify, in each group, an outcome corresponding to the smallest leader utility. An equilibrium is found by selecting, among all the remaining outcomes (at most one per leader's pure strategy) one which maximizes the leader's utility.

8.2 O/P-LFNE with leader in mixed and followers in pure (O/P-LMFP)

In the optimistic setting, the case in which only the followers are restricted to pure strategies can be solved by solving m^2 linear programming problems, one per followers' outcome. In each problem, we only have to impose best-response constraints on the followers' utilities guaranteeing that there is a leader's strategy δ for which the chosen outcome is an NE, maximizing the leader's utility at that outcome for δ. The follower's outcome and the corresponding δ yielding the largest leader utility is then an O-LFNE.

It is not difficult to see that the previous algorithm (which, overall, runs in polynomial time) is not correct in the pessimistic case. This is not surprising since, as shown in Coniglio et al. (2017, 2018), the optimization problem corresponding to the equilibrium-finding problem is $\mathcal{NP}$-hard in the pessimistic case even with followers restricted to pure strategies. For its solution, we can resort to the same methods proposed in this paper for the LMFM case, simply requiring ρ_1 and ρ_2 to be binary.

9 Computational results

For our computational experiments, we adopt a test bed composed of instances mainly taken from two GAMUT (Nudelman et al. 2004) classes: *Uniform RandomGames* (NF games) and *PolymatrixGames* (PM games), generated with payoffs in [0, 100].

For simplicity, we assume that all the players have the same number of actions m, i.e., that $m_p = m$ for all $p \in N$.

This is w.l.o.g., as one can always add extra actions to a player with a payoff small enough to guarantee that such actions will never be played at an equilibrium.

We experiment on games of increasing size of m and n, with $m \in \{2, 3, \ldots, 10\} \cup \{15, \ldots, 25\}$ when $n = 3$ (2 followers) and $m \in \{2, 3, \ldots, 10\}$ when $n \geq 4$ (3 or more followers). We generate 10 instances per value of m, n and game class.

For the experiments on NF games in the LMFM case, we also consider eight GAMUT classes of structured normal-form games, *BertrandOligopoly*, *Bidirectional-LEGs*, *MinimumEffortGames*, *RandomGraphicalGames*, *DispersionGames*, *CovariantGames*, *TravelersDilemma* and *UniformLEGs*, generating 10 instances with 2 followers and $m = 8$ actions per player for each of them.

Throughout the section, the results of our experiments are compared w.r.t. *computing time* (in seconds) and (multiplicative) *optimality gap*.[3] For both values, we report the arithmetic average for each game class and value of m and n over the 10 corresponding instances. In all the boxplots that we report, the red dash indicates the median, the box extends from the 25th to the 75th percentile, and dotted lines denote the whole sample distribution. Outliers are highlighted with a red mark.

We adopt five solvers: BARON and SCIP (for globally optimal solutions to every formulation, apart from O-PM-LPFM-III which is an MILP), CPLEX (for globally optimal solutions to O-PM-LPFM-III, as well as to the oracle formulation for PM games in the implicit-enumeration and black-box methods), SNOPT (for locally optimal solutions to the formulations with purely continuous variables) and RBFOpt as the backbone of our black-box heuristic for pessimistic cases of LFNE. (We will, nevertheless, also experiment with it for some optimistic variants.) The O-NF-LPFM-implicit-enumeration algorithm is implemented in C. The experiments are run on a UNIX computer with a dual quad-core CPU at 2.33 GHz, equipped with 8 GB of RAM. Each algorithm is run using a single thread within a time limit of 3600 seconds. For the exact methods, we halt the execution whenever the optimality gap reaches $10^{-12}\%$.[4]

9.1 O-NF-LMFM-I, II, and III ($n = 3$)

We compare the different NF formulations when solved with BARON and SCIP. For *RandomGames* instances, the average computing time and optimality gap for each combination of formulation and solver is reported in Fig. 1 as a function of m.

The results obtained with the two solvers are quite different. BARON better performs on O-NF-LMFM-I (the formulation with purely continuous variables), while SCIP better performs on O-NF-LMFM-III (the "reformulated" formulation which contains binary variables introduced to remove nonquadratic terms from O-NF-LMFM-II,

---

[3] The optimality gap is defined as $\min\left\{\frac{\text{UB}-\text{LB}}{\text{LB}} 100, 10^5\right\}$ %, where LB and UB are, resp., the largest lower bound (corresponding to the best feasible solution) and the smallest upper bound found by the solver within the time limit. The min operator prevents an unbounded value for LB = 0. An optimality gap of 10^5 highlights that the method fails to produce a useful solution as, due to the payoffs being in [0, 100], *any* strategy of the leader can achieve, at least, a utility of 0.

[4] Preliminary experiments with four tolerance values, namely $10^{-12}\%$, $10^{-9}\%$, $10^{-6}\%$ and $10^{-3}\%$, showed, for a larger tolerance, a negligible reduction in computing time by, at most and only in few instances, 2.5% with SCIP and 7.0% with BARON. The stricter tolerance was thus preferred.
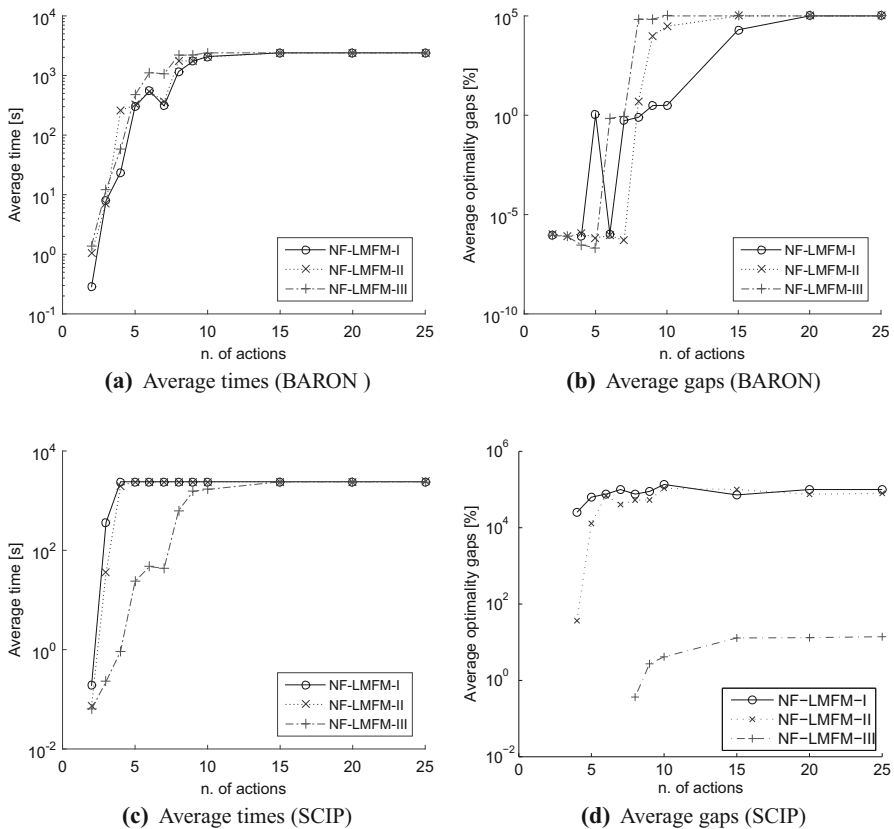
(a) Average times (BARON)

(b) Average gaps (BARON)

(c) Average times (SCIP)

(d) Average gaps (SCIP)

Fig. 1 Computing times and optimality gaps obtained with the NF-LMFM formulations

as well as extra valid constraints). These results suggest that the formulation which is solved more efficiently with each solver is O-NF-LMFM-I with BARON and O-NF-LMFM-III with SCIP. These results are in line with the general computational behavior of BARON and SCIP, as the former tends to exhibiting a better performance on highly nonlinear and mostly continuous problems, whereas the latter becomes more efficient as the number of integer/binary variables of the problem increases.

Further inspecting Fig. 1, we notice that, with SCIP, O-NF-LMFM-III always outperforms O-NF-LMFM-II. This shows that SCIP is incapable of automatically constructing the reformulation obtained with O-NF-LMFM-III.

As to the computing times, the largest m for which at least a game is solved to optimality by BARON within the time limit is $m = 8$ for O-NF-LMFM-I and $m = 7$ for the other formulations. With SCIP, we reach $m = 9$ with O-NF-LMFM-III and $m = 3$ with the other ones. In particular, SCIP with O-NF-LMFM-III always requires a shorter computing time than BARON with O-NF-LMFM-I for every number of actions.

In terms of optimality gaps, SCIP remarkably outperforms BARON. As one can see in Fig. 1b, d, the gap achieved by BARON with O-NF-LMFM-I reaches $10^5\%$
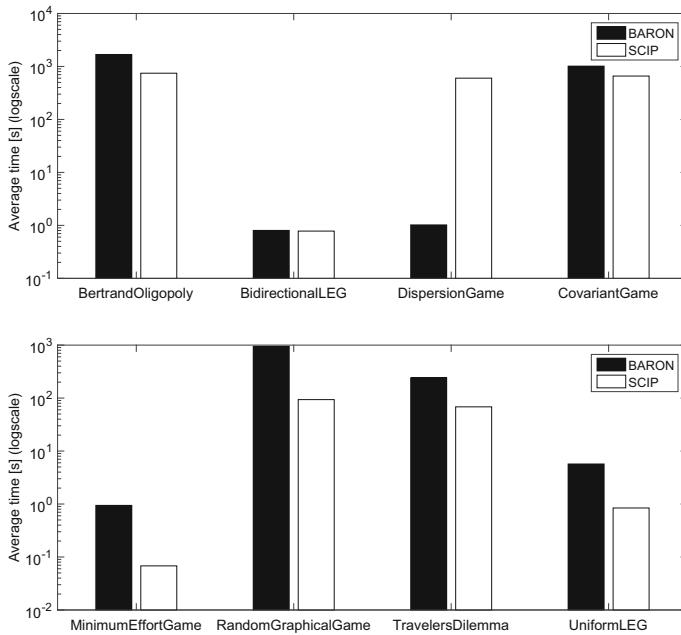
Fig. 2 Computing times obtained when solving formulation O-NF-LMFM-I with BARON and formulation O-NF-LMFM-III with SCIP for different GAMUT classes of structured games

when $m \geq 20$. This is due to the solver returning an LB of 0 due to failing find a feasible solution in the time limit. Differently, the gap achieved by SCIP with O-NF-LMFM-III is below 15% for m up to $m = 25$. Such results suggest that, for games of this size, one can always achieve an almost constant gap, contrarily to what the intrinsic difficulty of the problem would suggest, namely an exponential quality degradation as the number of actions grows. Moreover, these results show that SCIP with O-NF-LMFM-III always finds a feasible solution (an NE) for the followers' game and for some leader's strategy, differently from the other pairs of solver and formulation.

These observations are substantially confirmed when experimenting with the same solver/formulation pair on the eight structured classes of NF games. The average computing times reported in Fig. 2 are indeed in line with the trends we observed for *RandomGames*, with SCIP outperforming BARON most of the times (on average). This trend becomes different when considering *DispersionGames*, where SCIP performs less efficiently than for the other classes of games, achieving computing times which are considerably larger than those obtained with BARON. This is due to the solver failing to solve two game instances within the time limit. This can be better observed in Figure 3 which reports the computing times only for the instances that are solved to optimality with the two solvers, as well as the percentage of such instances. In particular, we observe that SCIP solves 91.875% of the instances on average, whereas BARON only solves 81.25%.
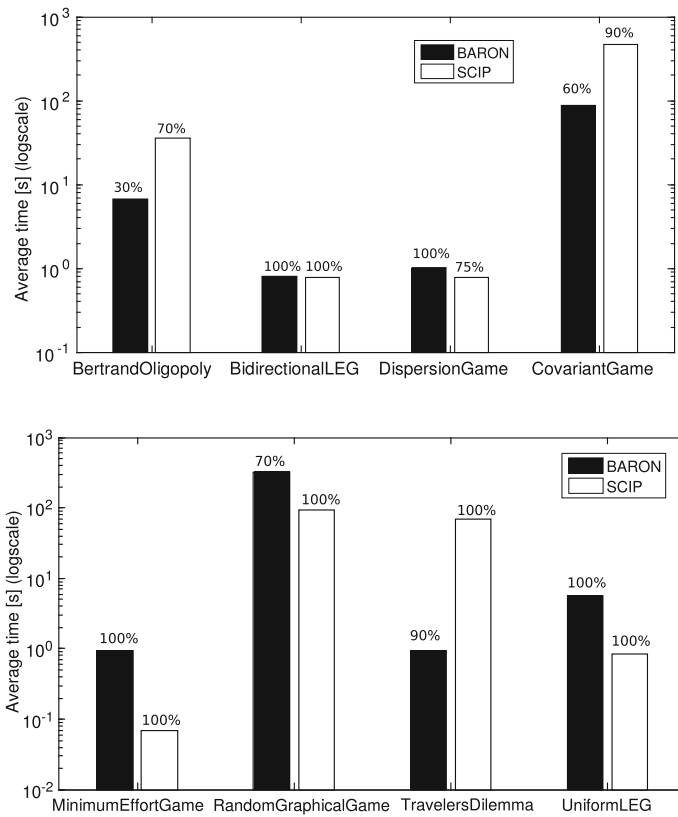
Fig. 3 Computing times only considering games for which the computations terminated; the percentage of instances solved to optimality is reported on top of each bar

9.2 O-PM-LMFM-I, II and III (*n* = 3)

In Fig. 4, we report the computing times and the optimality gaps obtained with SCIP for games of the GAMUT class *PolymatrixGames*. Since the results obtained with BARON are similar to those we illustrated for NF games, we omit them for the sake of brevity.

Within the time limit, the largest m for which at least an instance is solved to optimality is $m = 15$. For $m \leq 10$, all instances are solved to within a gap of 0 (within the numerical tolerance we set). In particular, the optimality gap is always below 15% for instances with up to $m = 25$, showing a trend which is substantially less steep than that for NF games. This suggests that PM games are, as expected, easier to solve.

9.3 O-NF-LMFM-I, local optimization (*n* = 3)

In Fig. 5, we report the experimental results obtained with SNOPT for *RandomGames* using formulation O-NF-LMFM-I. Due to the local optimization nature of the solver
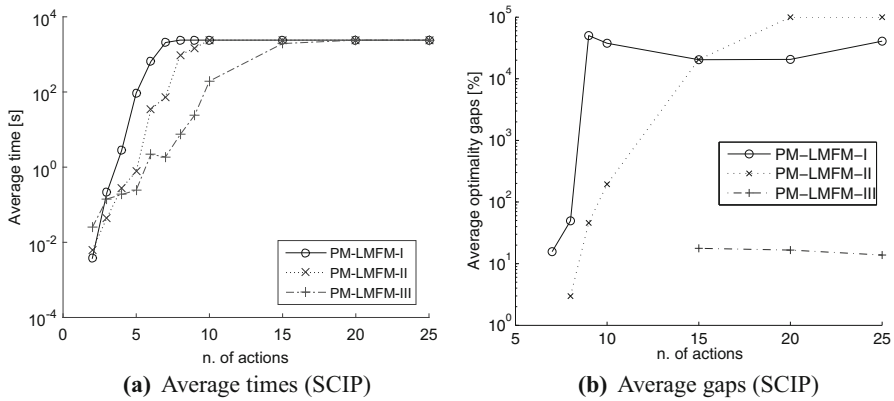
(a) Average times (SCIP)   **(b)** Average gaps (SCIP)

Fig. 4 Computing times and optimality gaps with SCIP with O-PM-LMFM formulations

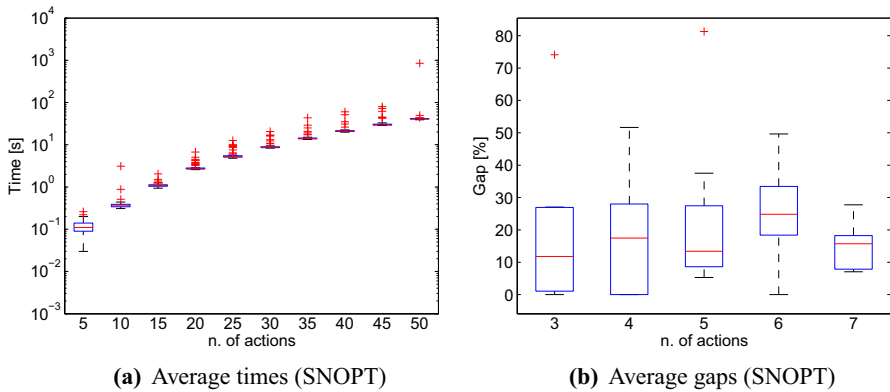**(a)** Average times (SNOPT)   **(b)** Average gaps (SNOPT)

Fig. 5 Computing times and $\frac{LB}{OPT}$ ratios obtained with SNOPT with O-NF-LMFM-I within 30 random restarts

for nonconvex problems, to obtain statistically more relevant results we run 30 restarts with different initial starting solutions, sampled uniformly at random from the simplices of the strategies of the three agents, and return the best solution found.

Figure 5a shows that the computing times with SNOPT (cumulated over the 30 random restarts) are much shorter than the computing times required by BARON and SCIP, allowing for solving (to a local optimum) almost all the instances with $m = 20$ within the time limit. Differently, as shown in Fig. 5b the quality of the solutions returned by SNOPT (measured as their ratio over the value of an optimal solution found by SCIP or BARON) is rather poor even with very few actions. Indeed, the median of the ratios is between 10 and 20% for games with up to $m = 7$. This emphasizes the effectiveness of our approach based on spatial-branch-and-bound methods.

9.4 O-NF/PM-LMFM-III ($n \geq 4$)

In Table 1, we report the average computing times obtained with SCIP when employing formulations O-NF-LMFM-III and O-PM-LMFM-III for games with 4 players

Table 1 Computing times (in seconds) with SCIP and O-NF/PM-LMFM-III, within a time limit of 3600 s

n/m	2	3	4	5
Normal-form games				
3	0.06	0.20	0.92	23.79
4	0.19	8.274	142.66	1304.45
5	278.06	409.78	2016.97	—
6	172.90	2350.95	2212.95	—

n/m	5	6	7	8	9	10
Polymatrix games						
3	0.24	2.17	1.87	7.31	24.45	194.71
4	4.84	10.85	121.57	247.84	622.72	1947.54
5	7.51	90.83	332.04	1982.77	2396.01	2175.29
6	10.31	1169.50	2062.75	—	—	—

or more. In the time limit, we can solve NF games with up to $m = 5$ for $n \leq 4$ (corresponding to up to $m^n = 625$ different outcomes and $nm^n = 2500$ different payoffs) and up to $m = 4$ for $n \leq 6$ (corresponding to up to $m^n = 4096$ outcomes and $nm^n = 24,576$ payoffs). Quite interestingly, with our methods we can tackle instances of a size comparable to that of the largest instances used in Porter et al. (2008) (such instances are generated with GAMUT (Nudelman et al. 2004) and are comparable to the ones in our test bed) to evaluate a set of algorithms proposed to find an NE (in a single-level problem), in spite of our problem being clearly harder (as it admits the former as a subproblem). With PM games, our algorithms scale much better, allowing for finding exact solutions to PM games with up to $m = 10$ for $n \leq 5$ and up to $m = 7$ for $n \leq 6$.

9.5 O/P-NF/PM-LMFM-blackBox

When experimenting with the black-box method, we first consider the optimistic case for NF games as, for it, we can compare the quality of the solutions we find to either the optimal solution value or its tightest upper bound. Namely, we compare O-NF-LMFM-blackBox to O-NF-LMFM-III, the latter solved with SCIP within the time limit. The results are reported in Fig. 6.

In Fig. 6a, we observe, on average and for $m \leq 10$, that the black-box method yields solutions to within 90% of the optimal ones found with SCIP. This suggests that the method might be sufficiently accurate. As shown in Fig. 6b, for $m \geq 10$ the burden of calling SCIP to solve the oracle formulation becomes too large, making the black-box algorithm impractical.

An interesting result, see Fig. 6a, concerns the gap between the utility of the leader at an optimistic LFNE or at a pessimistic LFNE. On the instances solved to optimality ($m \leq 5$), where we can verify the quality of the heuristic solutions, we see that the gap is rather small, suggesting that, in *RandomGames* instances generated with GAMUT,
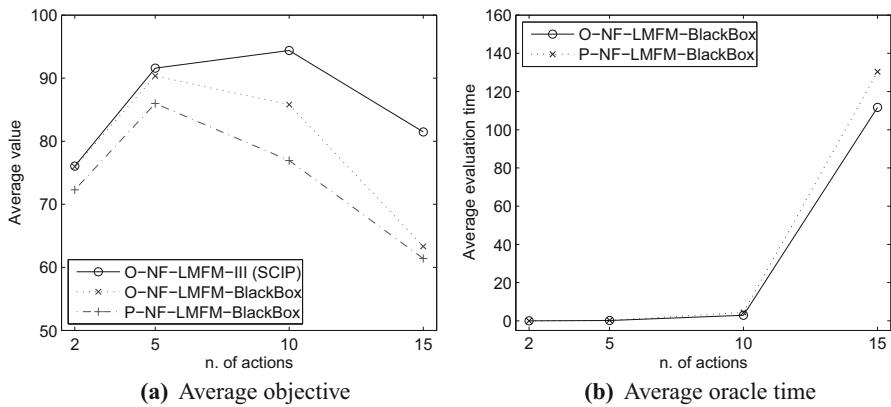
(a) Average objective   **(b)** Average oracle time

Fig. 6 Performance of the black-box approach for O/P-NF-LMFM compared to O-NF-LMFM-III

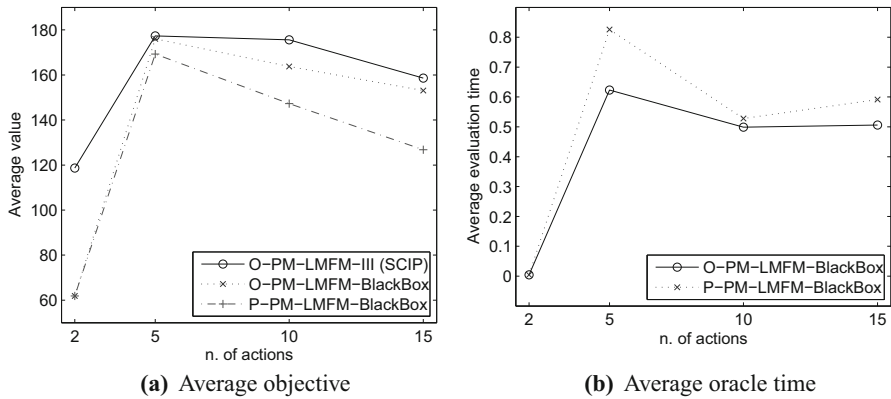**(a)** Average objective   **(b)** Average oracle time

Fig. 7 Performance of the black-box approach for O/P-PM-LMFM compared to O-PM-LMFM-III

the leader can manage to force the followers to play a strategy which provides the leader with a utility not dramatically smaller than that which they would obtain in an optimistic LFNE.

In Fig. 7, we report analogous results obtained with polymatrix games. In the time limit, we compare O-PM-LMFM-III solved with SCIP to O-PM-LMFM-blackBox. Differently from the NF case, Fig. 7b shows that, for PM games, the computing time needed to solve the oracle formulation (which is an MILP in this case) is much smaller and scales much better with m. Except for the case of $m = 2$, Fig. 7a allows us to draw comparable conclusions to those that we have drawn for the NF case, with the leader achieving, in the pessimistic case, solutions that are not too far away from the corresponding optimistic ones w.r.t. their utility.

9.6 O-NF/PM-LPFM and O-NF/PM-implicit-enumeration ($n = 3$)

Lastly, we focus on the case where the leader is restricted to pure strategies. We report the results in terms of computing times obtained by imposing $\delta \in \{0, 1\}^m$ in
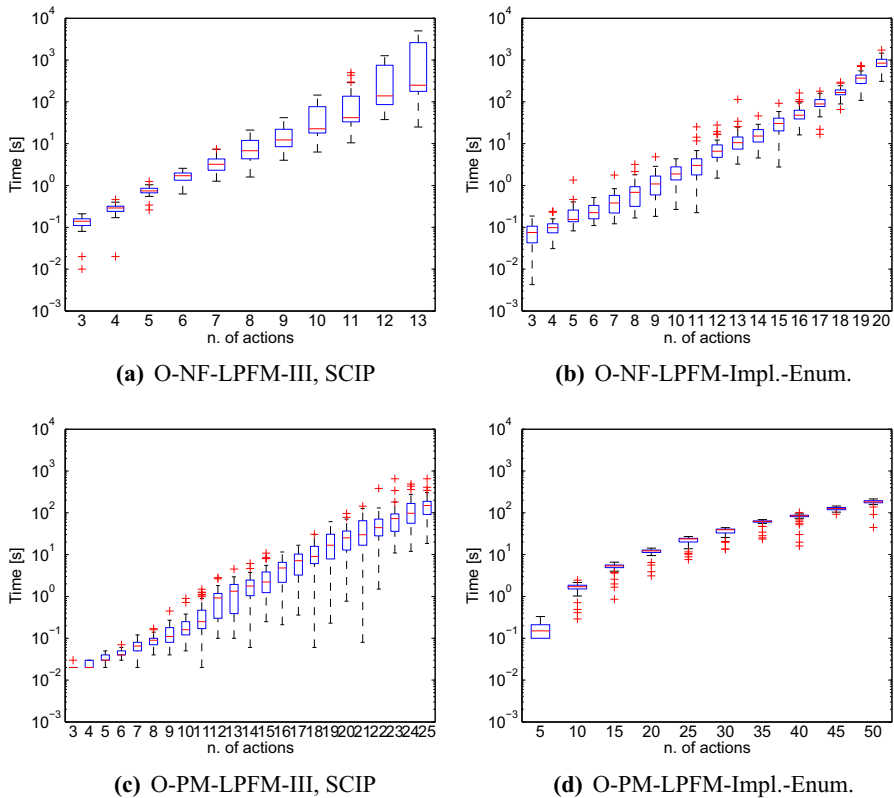
(a) O-NF-LPFM-III, SCIP

(b) O-NF-LPFM-Impl.-Enum.

(c) O-PM-LPFM-III, SCIP

(d) O-PM-LPFM-Impl.-Enum.

Fig. 8 Computing times on NF/PM-LPFM instances with O-NF/PM-LPFM-III (**a**/**c**) and O-NF/PM-LPFM-implicit-enumeration (**b**/**d**), using SCIP/CPLEX

O-NF/PM-LPFM-III with SCIP for *RandomGames* in Fig. 8a, b and with CPLEX for *PolymatrixGames* (for which the formulation becomes an MILP) in Fig. 8c, d. Interestingly, by imposing a binary δ to tackle the LPFM case the size of the largest instances solvable within the time limit increases from $m = 9$ to $m = 13$ in *RandomGames* and from $m = 15$ to $m = 25$ for *PolymatrixGames* when compared to the LMFM case.

For both *RandomGames* and *PolymatrixGames*, a dramatic performance improvement is obtained with O-NF/PM-LPFM-implicit-enumeration: with it, the size of the largest instance that we can solve increases from $m = 13$ to $m = 20$ for *RandomGames* and from $m = 25$ to $m = 50$ for *PolymatrixGames*. As expected, the computing times for *PolymatrixGames* are much smaller (due to only requiring the solution of an MILP at each step), allowing us to solve to optimality much larger instances.

10 Conclusions and future work

We have studied game-theoretic leader-follower (Stackelberg) situations with a bilevel structure where multiple followers play a Nash equilibrium once the leader has com-

mitted to a strategy. After analyzing the complexity of the problem, we have provided different algorithms and mathematical programming formulations to find an equilibrium for the optimistic case as well as a heuristic black-box method for the pessimistic case. We have conducted a thorough experimental evaluation of the different methods we have proposed, using various optimization solvers. Our experiments suggest that spatial-branch-and-bound solvers can be used as effective solution methods when coupled with our formulations, providing a reasonably good optimality gap even for large games.

Future works include the study of structured games, with focus on understanding how the specific structure of a game could be exploited to obtain easier to solve formulations (as we did for polymatrix games in this work).

Moreover, it would be of interest to study the adaptation of our techniques to succinct games (whose normal-form representation has exponential size) relying on cutting plane methods to cope with the presence of exponentially many best-response constraints, possibly using notions of diversity and bound improvement within the separation problem, see Amaldi et al. (2010, 2014), Coniglio and Tieves (2015), to achieve a faster convergence.

It would also be of interest to combine state-of-the-art equilibrium-finding algorithms for such games with methods similar to the black-box one we have proposed, which would directly benefit from the existence of an efficient equilibrium-finding algorithm for reoptimizing the followers' problem after changing the leader's strategy.

Future works also include the study of equilibrium-finding methods based on support enumeration, understanding, in particular, whether games which admit Nash equilibria of small support in the case without a leader would still admit small support equilibria in the Stackelberg case.

Among the challenging problems that we are interested to address in the future, we mention the design of algorithms to find an equilibrium when the followers play either a strong Nash equilibrium, a strong correlated equilibrium or a solution concept defined in cooperative game theory.

References

Al-Khayyal FA, Falk JE (1983) Jointly constrained biconvex programming. Math Oper Res 8(2):273–286

Alves MJ, Antunes CH (2016) An illustration of different concepts of solutions in semivectorial bilevel programming. In: 2016 IEEE symposium series on computational intelligence (SSCI). IEEE, pp 1–7

Amaldi E, Capone A, Coniglio S, Gianoli LG (2013) Network optimization problems subject to max–min fair flow allocation. IEEE Commun Lett 17(7):1463–1466

Amaldi E, Coniglio S, Gualandi S (2014) Coordinated cutting plane generation via multi-objective separation. Math Program 143(1–2):87–110

Amaldi E, Coniglio S, Gualandi S (2010) Improving cutting plane generation with 0-1 inequalities by bi-criteria separation. In: Festa P (ed) Experimental algorithms. SEA 2010. Lecture notes in computer science, vol 6049. Springer, Berlin, Heidelberg, pp 266–275

An B, Pita J, Shieh E, Tambe M, Kiekintveld C, Marecki J (2011) Guards and Protect: next generation applications of security games. ACM SIGecom Exch 10(1):31–34

Basilico N, Coniglio S, Gatti N, Marchesi A (2017) Bilevel programming approaches to the computation of optimistic and pessimistic single-leader–multi-follower equilibria. In: Iliopoulos CS, Pissism SP, Puglisi SJ, and Raman R (eds) 16th international symposium on experimental algorithms. SEA 2017. Leibniz international proceedings in informatics (LIPIcs), vol 75. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, pp 31:1–31:14

Basilico N, Coniglio S, Gatti N (2016) Methods for finding leader-follower equilibria with multiple followers: (extended abstract). In: Proceedings of the 15th international joint conference on autonomous agents and multiagent systems, AAMAS 2016, pp 1363–1364

Caprara A, Carvalho M, Lodi A, Woeginger G (2016) Bilevel knapsack with interdiction constraints. INFORMS J Comput 28(2):319–333

Castiglioni M, Marchesi A, Gatti N, Coniglio S (2018) Leadership in singleton congestion games: what is hard and what is easy. CoRR. arXiv:1808.10209

Chen X, Deng X (2006) Settling the complexity of two-player nash equilibrium. In: 2006 47th annual IEEE symposium on foundations of computer science (FOCS'06). IEEE, pp 261–272

Coniglio S, Tieves M (2015) On the generation of cutting planes which maximize the bound improvement. In: Bampis E (ed) Experimental algorithms. SEA 2015. Lecture notes in computer science, vol 9125. Springer, Cham, pp 97–109

Coniglio S, Gatti N, Marchesi A (2017) Pessimistic leader-follower equilibria with multiple followers. In: Proceedings of the 26th international joint conference on artificial intelligence, IJCAI 2017, pp 171–177

Coniglio S, Gatti N, Marchesi A (2018) Computing a pessimistic leader-follower equilibrium with multiple followers: the mixed-pure case. CoRR. arXiv:1808.01438

Conitzer V, Sandholm T (2008) New complexity results about Nash equilibria. Games Econ Behav 63(2):621–641

Conitzer V, Korzhyk D (2011) Commitment to correlated strategies. In: Proceedings of the 25th AAAI conference on artificial intelligence, AAAI 2011, pp 632–637

Conitzer V, Sandholm T (2006) Computing the optimal strategy to commit to. In: Proceedings of the 7th ACM conference on electronic commerce, EC 2006, pp 82–90

Costa A, Nannicini G, Schroepfer T, Wortmann T (2015) Black-box optimization of lighting simulation in architectural design. In: Cardin MA, Krob D, Lui P, Tan Y, Wood K (eds) Complex systems design & management Asia. Springer, Cham, pp 27–39

Kiekintveld C, Jain M, Tsai J, Pita J, Ordóñez F, Tambe M (2009) Computing optimal randomized resource allocations for massive security games. In: Proceedings of 8th international joint conference on autonomous agents and multiagent systems, AAMAS 2009, pp 689–696

Labbé M, Violin A (2016) Bilevel programming and price setting problems. Ann Oper Res 240(1):141–169

Leyffer S, Munson T (2010) Solving multi-leader-common-follower games. Optim Methods Softw 25(4):601–623

Marchesi A, Coniglio S, Gatti N (2018) Leadership in singleton congestion games. In: Proceedings of the 27th international joint conference on artificial intelligence, IJCAI 2018, pp 447–453

Matuschke J, McCormick ST, Oriolo G, Peis B, Skutella M (2017) Protection of flows under targeted attacks. Oper Res Lett 45(1):53–59

McCormick G (1976) Computability of global solutions to factorable nonconvex programs: Part I—convex underestimating problems. Math Program 10(1):147–175

Nash JF (1950) Equilibrium points in n-person games. Proc Natl Acad Sci 36:48–49

Nudelman E, Wortman J, Leyton-Brown K, Shoham Y (2004) Run the GAMUT: a comprehensive approach to evaluating game-theoretic algorithms. In: Proceedings of the 3rd international joint conference on autonomous agents and multiagent systems, AAMAS 2004, pp 880–887

Porter R, Nudelman E, Shoham Y (2008) Simple search methods for finding a nash equilibrium. Games Econ Behav 63(2):642–662

Sandholm T, Gilpin A, Conitzer V (2005) Mixed-integer programming methods for finding Nash equilibria. In: Proceedings of the 20th AAAI conference on artificial intelligence, AAAI 2005, pp 495–501

Sherali HD, Adams WP (1990) A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. SIAM J Discrete Math 3(3):411–430

Shoham Y, Leyton-Brown K (2008) Multiagent Systems: algorithmic, game theoretic and logical foundations. Cambridge University Press, Cambridge

von Stackelberg H (2010) Market structure and equilibrium. Springer, Berlin

von Stengel B (2010) Computation of nash equilibria in finite games: introduction to the symposium. Econ Theory 42(1):1–7

von Stengel B, Zamir S (2010) Leadership games with convex strategy sets. Games Econ Behavior 69:446–457

Yanovskaya EB (1968) Equilibrium points in polymatrix games. Lith Math J 8:381–384

Affiliations

Nicola Basilico[1] · **Stefano Coniglio[2]** · **Nicola Gatti[3]** · **Alberto Marchesi[3]**

✉ Stefano Coniglio
  s.coniglio@soton.ac.uk

  Nicola Basilico
  nicola.basilico@unimi.it

  Nicola Gatti
  nicola.gatti@polimi.it

  Alberto Marchesi
  alberto.marchesi@polimi.it

[1] Universitá degli Studi Di Milano, Via Comelico 39/41, 20135 Milan, Italy

[2] University of Southampton, University Road, Southampton SO17 1BJ, UK

[3] Politecnico di Milano, Via Ponzio, 34/5, 20133 Milan, Italy