Slim Belhaiza · Charles Audet · Pierre Hansen

A note on bimatrix game maximal Selten subsets

Abstract In this paper, we implement automatic procedures to enumerate all Nash maximal subsets of a bimatrix game and compute their dimensions. We propose a linear programming approach to identify extreme perfect Nash equilibria, enumerate all Selten maximal subsets and compute their dimensions. We present the $E\chi$-MIPerfect and the EEE-Perfect algorithms which enumerate all extreme perfect Nash equilibria. We finally report and comment computational experiments on randomly generated bimatrix games with different size and density.

Mathematics Subject Classification 90C90 · 91A05 · 91A10

<div dir="rtl">

الملخص

في هذه الورقة، نُنَفِّذُ إجراءات تلقائية لعدِّ جميع مجموعات ناش الأعْظميَّة الجزئية من لعبة مصفوفة مزدوجة وحساب أبعادها. نقترح مقاربة
برمجة خطية جديدة لتحديد توازنات ناش الكاملة القصوى، وعد جميع مجموعات سلتان الأعْظميَّة الجزئية، وحساب أبعادها. نقدم
خَوارزْميَّتَيْ MIP-E_χ-الكاملة وَ EEE-الكاملة لعدِّ جميع توازنات ناش الكاملة القصوى. في الختام، نقدم تقريراً ونعلق على تجارب حسابية
على ألعاب ذات مصفوفات مزدوجة يتم توليدها عشوائيّاً وذات أحجام وكثافات مختلفة.

</div>

1 Introduction

A *bimatrix game* is a strategic confrontation of two players I and II. Both players could be political, social or economic agents or institutions. Each player has a finite number of actions to choose from. The deterministic choice of an action is called *pure strategy*. Player I has to choose between n pure strategies, while player II

S. Belhaiza (✉)
King Fahd University of Petroleum and Minerals,
Dhahran, Saudi Arabia
E-mail: slim.belhaiza@gerad.ca; slimb@kfupm.edu.sa

C. Audet
GERAD, École Polytechnique de Montréal,
Montreal, Canada
E-mail: charles.audet@gerad.ca

P. Hansen
GERAD, École des Hautes Études Commerciales de Montréal,
Montreal, Canada
E-mail: pierre.hansen@gerad.ca

has to choose between m pure strategies. A bimatrix game $G(A, B)$ is described through a pair of $n \times m$ payoff matrices A and B. Elements a_{ij} and b_{ij} of matrices A and B are, respectively, the immediate payoffs of player I and player II when the first plays his ith strategy while the second simultaneously plays his jth strategy. Each player attempts to maximize his own payoff by selecting a probability vector over his set of pure strategies. These vectors are combinations of pure strategies, called *mixed strategies*, and represented by probability vectors $x_1 \in \mathbb{R}^n$ and $x_2 \in \mathbb{R}^m$. Hence, player I's payoff is $x_1^t A x_2$ and player II's payoff is $x_1^t B x_2$.

A Nash *equilibrium* is defined as a profile of strategies such that simultaneously player I maximizes his payoff given the strategic choice of player II and player II maximizes his payoff given the strategic choice of player I. An equilibrium point is a profile of strategies where neither player has an interest to unilaterally change his strategic choice. As shown by Nash [15], a bimatrix game has at least one equilibrium. Formally, a Nash equilibrium is a pair of strategies $(\hat{x}_1, \hat{x}_2)$ such that $\hat{x}_1 \in X_1(\hat{x}_2)$ and $\hat{x}_2 \in X_2(\hat{x}_1)$ where

$$
\begin{array}{ll}
X_1(\hat{x}_2) = \underset{x_1}{\operatorname{argmax}} \; x_1^t A \hat{x}_2 & X_2(\hat{x}_1) = \underset{x_2}{\operatorname{argmax}} \; \hat{x}_1^t B x_2 \\
\text{s.t.} \quad x_1^t e_n = 1, \quad \text{and} & \text{s.t.} \quad e_m^t x_2 = 1, \\
\qquad x_1 \geq 0, & \qquad x_2 \geq 0,
\end{array}
$$

and e_n and e_m are two $(n \times 1)$ and $(m \times 1)$ column vectors with all elements equal to 1. Clearly, $X_1(\hat{x}_2)$ for fixed $\hat{x}_2$, and $X_2(\hat{x}_1)$ for fixed $\hat{x}_1$ are polytopes.

Mills [13], and Mangasarian and Stone [11] studied the optimality conditions of the preceding system to establish necessary and sufficient conditions of equilibrium. Introducing real-valued variables α_1 and α_2, the duals of the above linear programs are

$$
\begin{array}{ll}
\min_{\alpha_1} \; \alpha_1 & \min_{\alpha_2} \; \alpha_2 \\
\text{s.t.} \; e_n \alpha_1 \geq A \hat{x}_2, & \text{s.t.} \; \alpha_2 e_m^t \geq \hat{x}_1^t B.
\end{array}
$$

Primal and dual feasibility conditions yield that a pair of strategies $(\hat{x}_1, \hat{x}_2)$ is a Nash equilibrium if there exist two scalars $\hat{\alpha}_1$ and $\hat{\alpha}_2$ satisfying

$$
(\hat{x}_1, \hat{\alpha}_2) \in X_1 \equiv \{(x_1, \alpha_2) \in \mathbb{R}^{n+1} : x_1^t B \leq \alpha_2 e_m^t, \; x_1^t e_n = 1, x_1 \geq 0\},
$$

and

$$
(\hat{x}_2, \hat{\alpha}_1) \in X_2 \equiv \{(x_2, \alpha_1) \in \mathbb{R}^{m+1} : A x_2 \leq e_n \alpha_1, \; e_m^t x_2 = 1, \; x_2 \geq 0\}.
$$

Moreover, from duality theory of linear programming, the optimal dual objective values $\hat{\alpha}_1$ and $\hat{\alpha}_2$ are, respectively, equal to the primal optimal payoffs of players I and II

$$
\hat{x}_1^t A \hat{x}_2 = \hat{\alpha}_1 \quad \text{and} \quad \hat{x}_1^t B \hat{x}_2 = \hat{\alpha}_2.
$$

The set *NE* of all equilibrium points is the union of a finite number of polytopes called *maximal Nash subsets* (Millham [12]). The set of vertices of the maximal Nash Subsets defines the set of extreme Nash equilibria. Each Nash equilibrium can be obtained by a convex combination of some extreme Nash equilibria. We have shown in [2] that complete enumeration of all bimatrix game extreme equilibria can be done through complete enumeration of all extreme feasible solutions of a mixed $0 - 1$ linear program with the following conditions (1.1) and any linear objective function

$$
\begin{aligned}
& \hat{x}_1^t e_n = 1, \\
& e_m^t \hat{x}_2 = 1, \\
& \hat{x}_1^t B - \alpha_2 e_m^t \leq 0, \\
& A \hat{x}_2 - \alpha_1 e_n \leq 0, \\
& \hat{x}_1 + u_1 \leq e_n, \\
& \hat{x}_2 + u_2 \leq e_m, \\
& (\alpha_1 e_n - A \hat{x}_2) - L_1 u_1 \leq 0, \\
& (\alpha_2 e_m^t - \hat{x}_1^t B) - L_2 u_2 \leq 0, \\
& \hat{x}_1 \geq 0, \; \hat{x}_2 \geq 0, \\
& u_1 \text{ and } u_2 \text{ binary vectors},
\end{aligned}
\tag{1.1}
$$

with L_1 and L_2 defined as large positive scalars adequately chosen. As the number of extreme solutions to this program could be very large, Game theorists introduced many refinements of the concept of Nash equilibria.

The Perfectness concept is a refinement based on the idea that a reasonable equilibrium should be stable against some slight perturbations in the equilibrium strategies [17].

In Sect. 2, we present automatic procedures to enumerate all Nash maximal subsets of a bimatrix game and compute their dimensions. In Sect. 3, we show that the *Perfectness* certificate of a Nash equilibrium is obtained through the optimization of a pair of linear programs. We propose a linear programming approach to identify extreme perfect Nash equilibria, enumerate all Selten maximal subsets and compute their dimensions. In Sect. 4, we present the $E\chi$-MIPerfect and the EEE-Perfect algorithms which enumerate all extreme perfect Nash equilibria. In Sect. 5, finally, we report and comment computational experiments on randomly generated bimatrix games with different size and density.

2 Enumeration of all maximal Nash subsets

The set *NE* of all equilibrium points is the union of a finite number of maximal Nash subsets. A subset $T \subset NE$ is a Nash subset if and only if every pair of elements in T is interchangeable, i.e. $(x_1, x_2) \in T$ and $(y_1, y_2) \in T$ imply that $(x_1, y_2) \in T$ and $(y_1, x_2) \in T$. A Nash subset T is called maximal if it is not properly contained in another Nash subset (Jansen [6]). Any extreme equilibrium is an extreme point of one of these maximal Nash subsets.

Enumeration of all maximal Nash subsets corresponding to a bimatrix game can be achieved using an algorithm for the enumeration of all maximal cliques of a graph G. We define $G = (V, E)$ as the graph obtained from the analysis of the extreme Nash equilibria in *NE*, which was first defined as the set of all Nash equilibria of the bimatrix game. The extreme points of *NE* define the set of nodes V of G while E is the set of edges of G. Any edge in E is a connection between two nodes (extreme Nash equilibria) of G, $X = (x_1, x_2) \in V$ and $Y = (y_1, y_2) \in V$, if and only if X and Y are interchangeable.

Every clique of the graph G is a *complete* subgraph of G. Each clique of G corresponds then to a set of extreme Nash equilibria, in which each extreme Nash equilibria is interchangeable with all others. Thus, a maximal Nash subset T is a maximal clique of G. Several papers study the maximal cliques enumeration problem. We have coded a C++ version of the widely used algorithm of Bron and Kerbosch [7]. Our implementation of this algorithm is included in the *XGame1.0* package available in [5].

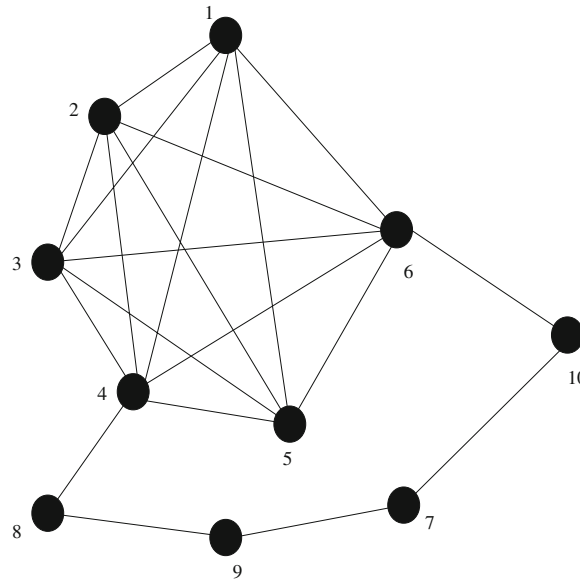*Example 2.1* Let A and B be the payoff matrices of a bimatrix game taken from Borm et al. [6]

$$A = \begin{pmatrix} -3 & -3 & -3 & -3 \\ 3 & 3 & 3 & -6 \\ -3 & -3 & -3 & -3 \end{pmatrix} \qquad B = \begin{pmatrix} 3 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 2 \end{pmatrix}.$$

Using the $E\chi MIP$ [2] algorithm, ten extreme Nash equilibria are enumerated (Table 1). This game has six maximal Nash subsets $T_1 = \{1, 2, 3, 4, 5, 6\}$, $T_2 = \{4, 8\}$, $T_3 = \{6, 10\}$, $T_4 = \{7, 9\}$, $T_5 = \{7, 10\}$ and $T_6 = \{8, 9\}$.

The interchangeability graph G obtained from *NE* is represented by Fig. 1.

Table 1 Extreme Nash equilibria from Borm et al. [6]

Eq.	x_1				x_2			α_1	α_2
1	0	1	0	0	0	1	0	3	0
2	0	1	0	0	1	0	0	3	0
3	0	1	0	1	0	0	0	3	0
4	0	1	0	0	0	1/3	2/3	-3	0
5	0	1	0	0	1/3	0	2/3	-3	0
6	0	1	0	1/3	0	0	2/3	-3	0
7	2/3	0	1/3	0	0	0	1	-3	2
8	1/3	0	2/3	0	0	1/3	2/3	-3	2
9	1/3	0	2/3	0	0	0	1	-3	2
10	2/3	0	1/3	1/3	0	0	2/3	-3	2

Fig. 1 Interchangeability graph $G = (V, E)$ from Borm et al. [6]

2.1 Maximal Nash subset geometry and dimension

To study the geometry and the dimension of maximal Nash subsets, we need to introduce some notation. Let us define:

$$S_n = \left\{ x_1 \in \mathbb{R}^n, \quad x_1 \geq 0, \sum_{i=1}^n x_{1i} = 1 \right\} \quad \text{and} \quad S_m = \left\{ x_2 \in \mathbb{R}^m, \quad x_2 \geq 0, \sum_{j=1}^m x_{2j} = 1 \right\}.$$

Heuer and Millham [8] showed that a maximal Nash subset for an $n \times m$-bimatrix game is a closed and convex subset of $S_n \times S_m$. This implied for Jansen [9] that a maximal Nash subset T is the Cartesian product of two convex polytopes $K(\dot{x}_2)$ and $L(\dot{x}_1)$, such that $(\dot{x}_1, \dot{x}_2) \in relint(T)$ (relative interior of T).

Then, $T = K(\dot{x}_2) \times L(\dot{x}_1)$, where

$$K(\dot{x}_2) = \{x_1 \in S_n, \ (x_1, \dot{x}_2) \in NE\} \quad \text{and} \quad L(\dot{x}_1) = \{x_2 \in S_m, \ (\dot{x}_1, x_2) \in NE\}.$$

For an equilibrium (x_1, x_2) of a bimatrix game $G(A, B)$, $M(A, x_2)$ is defined as the set of pure best replies of player I against x_2:

$$M(A, x_2) = \{i \in N = \{1, \ldots, n\}; A_i x_2 = \max_{k \in N} A_k x_2\}, \tag{2.1}$$

and similarly,

$$M(x_1, B) = \{j \in M = \{1, \ldots, m\}; x_1^t B^j = \max_{r \in M} x_1^t B^r\}, \tag{2.2}$$

is the set of pure best replies of player II against x_1.

We also define $C(x_1) = \{i \in N; x_{1i} > 0\}$ as the *carrier* of x_1 and $C(x_2) = \{j \in M; x_{2j} > 0\}$ as the carrier of x_2.

Let $G(A, B)$ be a bimatrix game and let T be a maximal Nash subset for the game $G(A, B)$, then the matrices $A(T) = [a_{ij}]_{i \in M(A, x_2), j \in C(x_2)}$ and $B(T) = [b_{ij}]_{i \in C(x_1), j \in M(x_1, B)}$ do not depend on the choice of the point $(\dot{x}_1, \dot{x}_2) \in relint(T)$ (Jansen [9]).

Jansen proved that if $(x_1, x_2) \in relint(T)$, then $dim[L(x_1)] = |C(x_2)| - rank[A(T)]$ and $dim[K(x_2)] = |C(x_1)| - rank[B(T)]$. Since T is the Cartesian product of the two convex polytopes $L(x_1)$ and $K(x_2)$ and $L(x_1) \cap K(x_2) = \emptyset$, then $dim(T) = dim[L(x_1)] + dim[K(x_2)]$.

We use these results to implement an automatic procedure that determines the dimension of each maximal Nash subset. To do so, we generate an equilibrium point (x_1, x_2) in the relative interior of the maximal Nash subset T using a convex combination of the extreme equilibrium points of T, since T is convex and closed. We then use exact arithmetics as in [3] to determine the rank of the corresponding matrices $A(T)$ and $B(T)$.

Example 2.2 For the bimatrix game taken from Borm et al. [6], we have determined the dimensions of the six maximal Nash subsets found (Table 2).

3 Extreme perfect equilibria

When decision-makers face a large set of Nash equilibria to choose from, additional rational concepts can be used to refine the set. As described by Borm et al. [6], a *maximal Selten subset* is a set of interchangeable perfect equilibria. Each maximal Selten subset is a subset of a maximal Nash subset and each extreme point of a maximal Selten subset corresponds to an extreme perfect equilibrium. Borm et al. [6] mention also that each maximal Selten subset is a *face* of a maximal Nash subset. To the best of our knowledge, the implementation of computational methods to enumerate all maximal Selten subsets of bimatrix games was not achieved before. However, the computation of normal form perfect equilibria was already achieved by von Stengel et al. [19] for extensive two-person games. In their paper, von Stengel et al. use the sequence form of extensive two-person games. Starting from an arbitrary vector, von Stengel et al. present an algorithm which traces an equilibrium on a piecewise linear path in the sequence form strategy space. They show that if the starting vector represents a pair of completely mixed strategies ($x_{1i} > 0$ and $x_{2j} > 0$, $\forall\, i,\ j$), then the equilibrium is normal form perfect.

The properties of maximal Nash subsets allow us to define a complete method to systematically enumerate all the extreme equilibria using the $E\chi MIP$ [2] and the EEE [1] algorithms and determining which of them correspond to a maximal Selten subset extreme point, i.e. an extreme perfect equilibrium. Using this approach, automatic identification of all perfect extreme Nash equilibria is made possible. Let us first define $CM_1(A, x_2) = conv(M(A, x_2))$ and $CM_2(x_1, B) = conv(M(x_1, B))$. The following definition summarizes the definition of a perfect equilibrium given by Selten [17] .

Definition 3.1 Let $G(A, B)$ be an $n \times m$ bimatrix game. An equilibrium $(\hat{x}_1, \hat{x}_2) \in S_n \times S_m$ is called *perfect* if there exists a sequence $\{(x_1^k, x_2^k)\}_{k \in \mathbb{N}}$ of completely mixed strategy pairs converging to $(\hat{x}_1, \hat{x}_2)$, such that, for all $k \in \mathbb{N}$, $\hat{x}_1 \in CM_1(A, x_2^k)$ and $\hat{x}_2 \in CM_2(x_1^k, B)$.

Selten [17] and Myerson [14] show that there is always at least one perfect equilibrium for any strategic form game. To define an automatic procedure to identify perfect equilibria, we will use the following characterization by Van Damme [18].

Definition 3.2 Let $(\hat{x}_1, \hat{x}_2)$ be a Nash equilibrium of the bimatrix game. If there is a unit vector x_1 such that $x_1^t A \geq \hat{x}_1^t A$ and $x_1^t A \neq \hat{x}_1^t A$, or if there is a unit vector x_2 such that $Bx_2 \geq B\hat{x}_2$ and $Bx_2 \neq B\hat{x}_2$, then $(\hat{x}_1, \hat{x}_2)$ is not perfect. Otherwise, $(\hat{x}_1, \hat{x}_2)$ is said to be perfect.

This characterization of perfect equilibrium strategies can be used to show that it is possible to verify if an equilibrium $(\hat{x}_1, \hat{x}_2)$ is perfect or not. In the following Definition 3.3, we introduce two linear programs to state an alternative definition of a perfect equilibrium.

Definition 3.3 The equilibrium $(\hat{x}_1, \hat{x}_2)$ is perfect if and only if both of the optimal objective functions values of the following linear programs are equal to zero

$$
\begin{array}{ll}
\max\limits_{(x_1,\epsilon_1)\in\Re^n\times\Re^m} e_m^t \epsilon_1 & \max\limits_{(x_2,\epsilon_2)\in\Re^m\times\Re^n} e_n^t \epsilon_2 \\
\text{s.t} \quad e_n^t x_1 = 1, & \text{s.t} \quad e_m^t x_2 = 1, \\
\quad\quad x_1^t A \geq \hat{x}_1^t A + \epsilon_1, & \quad\quad Bx_2 \geq B\hat{x}_2 + \epsilon_2, \\
\quad\quad x_1, \epsilon_1 \geq 0, & \quad\quad x_2, \epsilon_2 \geq 0.
\end{array} \tag{3.1}
$$

Table 2 Maximal Nash subsets dimensions from Borm et al. [6]

T_i	Dimension
$i = 1$	4
$i = 2$	3
$i = 3$	3
$i = 4$	2
$i = 5$	2
$i = 6$	2

To show the equivalence between Definitions 3.2 and 3.3, let (x_1^*, ϵ_1^*) and (x_2^*, ϵ_2^*) be optimal solutions for the two programs (3.1). If one of the two optimal objective function values is strictly positive, then at least one of the ϵ variables is strictly positive. Hence, there is at least one $\epsilon_1^i > 0, i \in \{1, 2, \ldots, n\}$, or $\epsilon_2^j > 0, j \in \{1, 2, \ldots, m\}$, such that $x_1^{t*} A_i \geq \hat{x}_1^t A_i + \epsilon_1^i$, or $B^j x_2^* \geq B^j \hat{x}_2 + \epsilon_2^j$, with A_i the ith row of A and B^j the jth column of B. Thus, we either have $x_1^{t*} A_i > \hat{x}_1^t A_i$, or $B^j x_2^* > B^j \hat{x}_2$. Therefore, we have $x_1^t A \neq \hat{x}_1^t A$, or $Bx_2 \neq B\hat{x}_2$, while $x_1^{t*} A \geq \hat{x}_1^t A + \epsilon_1$ and $Bx_2^* \geq B\hat{x}_2 + \epsilon_2$ are both satisfied. Hence, the equilibrium $(\hat{x}_1, \hat{x}_2)$ is not perfect.

If the two optimal objective functions are equal to zero, then all of ϵ_1^* and ϵ_2^* vectors elements are equal to zero. These ϵ_1^* and ϵ_2^* vectors correspond to the maximum slack vectors between $x_1^{t*} A$ and $\hat{x}_1^t A$, and between Bx_2^* and $B\hat{x}_2$. Thus, $x_1^{t*} A = \hat{x}_1^t A$ and $Bx_2^* = B\hat{x}_2$. Hence, if all the ϵ variables are equal to zero the equilibrium $(\hat{x}_1, \hat{x}_2)$ is perfect. The two linear programs (3.1) are always feasible. For example, setting $\epsilon_1 = 0$ and $x_1 = \hat{x}_1$ or $\epsilon_2 = 0$ and $x_2 = \hat{x}_2$ always provides a feasible solution to each of the two linear programs.

Example 3.4 For the second extreme Nash equilibrium found for the bimatrix game taken from Borm et al. [6], the corresponding pair of linear programs is expressed as follows:

$$
\begin{aligned}
&\max_{x_1, \epsilon_1} \quad \epsilon_{11} + \epsilon_{12} + \epsilon_{13} + \epsilon_{14} && \max_{x_2, \epsilon_2} \quad \epsilon_{21} + \epsilon_{23} \\
&\text{s.t.} && \text{s.t.} \\
&x_{11} + x_{12} + x_{13} = 1, && x_{21} + x_{22} + x_{23} + x_{24} = 1, \\
&-3x_{11} + 3x_{12} - 3x_{13} \geq 3 + \epsilon_{11}, && 3x_{21} + 2x_{24} \quad\quad \geq 0 + \epsilon_{21}, \\
&-3x_{11} + 3x_{12} - 3x_{13} \geq 3 + \epsilon_{12}, && \\
&-3x_{11} + 3x_{12} - 3x_{13} \geq 3 + \epsilon_{13}, && 3x_{23} + 2x_{24} \quad\quad \geq 0 + \epsilon_{23}, \\
&-3x_{11} - 6x_{12} - 3x_{13} \geq -6 + \epsilon_{14}, && \\
&x_1, \epsilon_1 \geq 0, && x_2, \epsilon_2 \geq 0.
\end{aligned}
$$

One should note that the exact arithmetics libraries used in [3] were used to obtain exact solutions for these two linear programs. For this extreme Nash equilibrium, none of the ϵ_{1j} variables can exceed zero. However, $\hat{x}_2^t = (0, 1, 0, 0)$ is dominated by $x_2^t = (0, 0, 0, 1)$, as $\epsilon_{21} = 2$ and $\epsilon_{23} = 2$. Thus, the second extreme Nash equilibrium is not perfect.

Eight of the ten extreme Nash equilibria enumerated are perfect. These extreme perfect equilibria are 1, 3, 4, 6, 7, 8, 9 and 10. Given the Nash maximal subsets identified earlier, and by eliminating the second and the fifth non-perfect extreme equilibria, the maximal Selten subsets of this game are: $Sl_1 = \{1, 3, 4, 6\}$, $Sl_2 = \{4, 8\}$, $Sl_3 = \{6, 10\}$, $Sl_4 = \{7, 9\}$, $Sl_5 = \{7, 10\}$ and $Sl_6 = \{8, 9\}$.

3.1 Maximal Selten subset geometry and dimension

Let *NP* be the set of extreme perfect Nash equilibria of a bimatrix game. As a maximal Selten subset is a face of a maximal Nash subset, Borm et al. [6] define a dimension relation for maximal Selten subsets.

Let *Sl* be a maximal Selten subset for a bimatrix game $G(A, B)$. One can write:

$$ Sl = U(\dot{x}_2) \times V(\dot{x}_1), $$

where $U(\dot{x}_2) = \{x_1 \in S_n, (x_1, \dot{x}_2) \in NP\}$ and $V(\dot{x}_1) = \{x_2 \in S_m, (\dot{x}_1, x_2) \in NP\}$.

Let us also define the matrices

$$ A(Sl) = \big[a_{ij}\big]_{i \in M(A, x_2), j \in C(x_2)} \quad \text{and} \quad B(Sl) = \big[b_{ij}\big]_{i \in C(x_1), j \in M(x_1, B)}. $$

Hence, if $(x_1, x_2) \in relint(Sl)$ then

$$ dim\,[U(x_1)] = |C(x_2)| - rank\,[A(Sl)] \quad \text{and} \quad dim\,[V(x_2)] = |C(x_1)| - rank\,[B(Sl)]. $$

While this definition applies only for bimatrix games $G(A, B)$ such that $A, B > 0$, it is well known that every bimatrix game can be modified to make $A, B > 0$, and without changing the set of maximal Nash subsets. For example, this can easily be done by adding $1 + |a_{min}|$, with $a_{min} = min\ a_{ij}$, to each element of A and $1 + |b_{min}|$, with $b_{min} = min\ b_{ij}$, to each element of B.

We can use this result to automatically compute the dimension of all maximal Selten subsets of a given bimatrix game $G(A, B)$. We extended the automatic procedure implemented in Sect. 2.1 to determine the dimension of each maximal Selten subset obtained by identification of all extreme perfect Nash equilibria using the linear programs presented in Definition 3.3.

Example 3.5 We have determined the dimensions of the six maximal Selten subsets found for the bimatrix game taken from Borm et al. [6] (Table 3).

3.2 Detection of extreme perfect Nash equilibria

We next combine the pair of linear programs proposed earlier (Definition 3.3) in a single linear program. The optimal objective function value of this linear program would indicate if the Nash equilibrium is perfect or not.

Proposition 3.6 *The equilibrium $(\hat{x}_1, \hat{x}_2)$ is perfect if and only if the optimal objective function value of the following linear program $Q(\hat{x}_1, \hat{x}_2)$ is equal to zero.*

$$
\begin{aligned}
\max_{x_1, x_2, \epsilon_1, \epsilon_2} \quad & e_m^t \epsilon_1 + e_n^t \epsilon_2 \\
s.t \quad & e_n^t x_1 = 1, \\
& e_m^t x_2 = 1, \\
& x_1^t A \geq \hat{x}_1^t A + \epsilon_1 \\
& B x_2 \geq B \hat{x}_2 + \epsilon_2, \\
& x_1 \geq 0, \ \epsilon_1 \geq 0, \\
& x_2 \geq 0, \ \epsilon_2 \geq 0.
\end{aligned}
\tag{3.2}
$$

Proof If the optimal objective function is equal to zero, then all of ϵ_1^* and ϵ_2^* vectors elements are equal to zero. These ϵ_1^* and ϵ_2^* vectors correspond to the maximum slack vectors between $x_1^{t*} A$ and $\hat{x}_1^t A$, and $B x_2^*$ and $B \hat{x}_2$. Thus, $x_1^{t*} A = \hat{x}_1^t A$ and between $B x_2^* = B \hat{x}_2$. At optimality, if all the ϵ variables are equal to zero the equilibrium $(\hat{x}_1, \hat{x}_2)$ is perfect. □

4 Enumeration of extreme perfect Nash equilibria

To detect all extreme perfect Nash equilibria of bimatrix games directly, the two state of the art algorithms $E\chi$-MIP [2,3] and the EEE [1] were modified to incorporate the linear program (3.2) as a subproblem during the enumeration of all extreme Nash equilibria. In this section, we show how the $E\chi$-MIP algorithm was improved and extended order to enumerate all extreme perfect Nash equilibria. We also show how the EEE was implemented in exact arithmetic and extended to achieve the main goal of this work.

4.1 The $E\chi$-MIPerfect algorithm

The algorithm $E\chi$-MIPerfect is an improved and extended version of the $E\chi$-MIP algorithm for the enumeration of all extreme equilibria for bimatrix and polymatrix games. $E\chi$-MIPerfect generates a main branching tree and many auxiliary trees. At each node of the main branching tree, a dichotomous branching over one of the binary variables is performed.

   The main tree seeks binary variables combinations involved in one or more extreme equilibria. An auxiliary tree is generated from every main tree node offering an extreme feasible solution, i.e. an extreme Nash equilibrium. Each binary variable combination involved in an extreme Nash equilibrium is then explored to find all extreme perfect Nash equilibria resulting from this combination. To avoid redundant exploration of binary variables combinations, a cutting constraint is inserted once auxiliary exploration is completed. Moreover,

Table 3 Maximal Selten subsets dimensions from Borm et al. [6]

Sl_i	Dimension
$i = 1$	3
$i = 2$	3
$i = 3$	3
$i = 4$	2
$i = 5$	2
$i = 6$	2

these cutting constraints are often redundant and can be eliminated to reduce the size of the program.

The master problem (P) corresponds to a $0 - 1$ mixed linear program resulting from conditions (1.1)

$$
\begin{aligned}
&\max_{\hat{x}_1, \hat{x}_2, u_1, u_2, \alpha_1, \alpha_2} \quad \hat{x}_1^t e_n + e_m^t \hat{x}_2 \\
&\text{s.t.} \\
&(1) \quad \hat{x}_1^t e_n = 1, \\
&(2) \quad e_m^t \hat{x}_2 = 1, \\
&(3) \quad \hat{x}_1^t B - \alpha_2 e_m^t \;\leq\; 0, \\
&(4) \quad A\hat{x}_2 - \alpha_1 e_n \;\leq\; 0, \\
&(5) \quad \hat{x}_1 + u_1 \leq e_n, \\
&(6) \quad \hat{x}_2 + u_2 \leq e_m, \\
&(7) \quad (\alpha_1 e_n - A\hat{x}_2) \;-\; L_1 u_1 \;\leq\; 0, \\
&(8) \quad (\alpha_2 e_m^t - \hat{x}_1^t B) \;-\; L_2 u_2 \;\leq\; 0, \\
&(9) \quad \hat{x}_1 \geq 0, \quad \hat{x}_2 \geq 0, \\
&(10) \quad u_1 \in \{0, 1\}^n \text{ and } u_2 \in \{0, 1\}^m.
\end{aligned}
\tag{4.1}
$$

At each node of the main tree, a master problem (P) with at most two constraints fixing one or more binary variables, and one or more combination of binary variables cutting constraints, is solved. This represents a reduction in the number of binary variables fixing constraints compared to the original version of the $E\chi$-MIP algorithm. Each time a new extreme Nash equilibrium is obtained, the corresponding subproblem (Q) defined by Proposition 3.6 is solved. In the case where the optimal objective of the subproblem (Q) is equal to zero, the extreme Nash equilibrium is perfect. At each node of an auxiliary tree node, a linear program is solved. This linear program results from the $0 - 1$ mixed linear program (4.1), i.e. master problem (P), with one combination of binary variables fixing constraints and at most two constraints fixing one or more continuous variables. This represents again a reduction in the number of continuous variables fixing constraints compared to the original version of the $E\chi$-MIP algorithm. If a new extreme Nash equilibrium is obtained, the corresponding subproblem (Q) defined by Proposition 3.6 is solved. In the case where the optimal objective of the subproblem (Q) is equal to zero, the extreme Nash equilibrium is perfect.

4.1.1 Improvement of the variable fixing constraints

With the original version $E\chi$-MIP algorithm, at each node of the main tree one or more binary variables are fixed to 0 or 1. This has been performed by adding one constraint for each variable set to a fixed value, i.e. $u_k^q = 0$ or $u_k^q = 1$, where k $(k = 1, 2)$ is the player's index and q is the strategy's index of player k. These constraints are represented by quasi-empty rows with respect to the columns entries since one single variable is considered at each new branching. One single entry of the inserted row is non-zero and the right hand side is either equal to 1 or 0. Considering a bimatrix game where player I has n pure strategies and player II has m pure strategies, $n + m$ quasi-empty rows are added once the maximal depth in the main tree is reached. One can think also that the same number of slack variables is added. Moreover, these equality constraints would increase the degeneracy of the linear programs to be solved. This can be improved using the following constraint formulation. Only two rows are added to the original $0 - 1$ mixed linear program (P) to represent the main tree branching.

For each branch where a binary variable $u_k^q = 0$, the column entry of u_k^q in the following condition is set to 1;

$$
\text{For each } u_k^q = 0 \quad \Rightarrow \quad 0 + \cdots + u_k^q + \cdots + 0 \leq 0.
\tag{4.2}
$$

The same way, for each branch where a binary variable $u_k^q = 1$, the column entry of u_k^q in the following condition is set to 1;

$$
\text{For each } u_k^q = 1 \quad \Rightarrow \quad 0 + \cdots + u_k^q + \cdots + 0 \geq 0 + \cdots + 1 + \cdots + 0.
\tag{4.3}
$$

At each node of an auxiliary tree, one or more real variables are fixed to 0. With the original version $E\chi$-MIP algorithm, this has been performed by adding one constraint for each variable set to zero, i.e. $x_k^q = 0$. Therefore, we can use the same formulation as for the binary variables branching.

For each branch where a real variable $x_k^q = 0$, the column entry of x_k^q in the following condition is set to 1;

$$
\text{For each } x_k^q = 0 \Rightarrow 0 + \cdots + x_k^q + \cdots + 0 \leq 0.
\tag{4.4}
$$

4.1.2 Eχ-MIPerfect pseudo-code

The $E\chi$-MIPerfect algorithm can be formally stated as follows.

Step a. **Initialization.**

Let

– P; Master problem defined by the mixed 0-1 linear program (4.1).
– Q; Subproblem defined by the linear program (3.2).
– $\hat{X}$; Set of P's real variables $\hat{X}$.
– X; Set of Q's real variables X.
– U; Set of P's binary variables.
– $E = \emptyset$; Set of extreme equilibria.
– $NP = \emptyset$; Set of extreme perfect Nash equilibria.
– $N = 0$; Depth level in the main tree.
– CR; main tree root node.
– C; Current node.
– $\hat{x}_k^q$; Real variable associated with player k ($k = 1, 2$) $q^t h$ strategy.
– u_k^q; Binary variable associated with player k ($k = 1, 2$) $q^t h$ strategy.

Set $C = CR$ and **go to step** b.

Step b. **Solving master problem.**

If $N \leq |\hat{X}|$, **solve** current node's program P.

If the program is infeasible **go to step** d.

Else

– **Let** $S = (\hat{X}, U)$ be the solution obtained ; If $\hat{X} \notin E$, add S to E and **go to step** c.
– **Go to step** d.

Step c. **Solving subproblem.**

Solve $Q(\hat{X})$ to optimality. If the optimal objective of $Q(\hat{X})$ is equal to 0, **add** $\hat{X}$ to NP.

Step d. **Auxiliary Branching.**

If the current node C belongs to the main tree, **set** binary variables vector U at its value in S.

For each $\hat{x}_k^q > 0 \in \hat{X}$: **Set** the entry of x_k^q to 1 in condition (4.4) and **go to step** b.

Step e. **Main Branching.**

If the current node belongs to the main tree, no extreme perfect equilibria can be found from this node or its sons.

Else, **return** to the father node in the main tree and **add** a cutting constraint to eliminate the combination of binary variables U in S.

Select a binary variable $u_k^q \in U$ on which no branching was performed in the preceding nodes and such that its real variable x_k^q is the closest to $\frac{1}{2}$.

Let $p = N + 1$.

If $p \leq |\hat{X}|$ **set** $N = p$ and

– **Set** the entry of u_k^q to 1 in condition (4.2). If $u_k^q = 1$ in U: **Remove** the cutting constraint on the combination of binary variables U and **go to step** b.
– **Set** the entry of u_k^q to 1 in condition (4.3). If $u_k^q = 0$ in U: **Remove** the cutting constraint on the combination of binary variables U and **go to step** b.

Else ($p > |\hat{X}|$), **go to step** f.

Step f. **End.**

The sets E and NP contain, respectively, all extreme equilibria and all extreme perfect equilibria of the bimatrix game.

Theorem 4.1 *The algorithm $E\chi$-MIPerfect enumerates in finite time all extreme perfect Nash equilibria points of any bimatrix game.*

Proof The algorithm seeks all possible ways to satisfy the complementary conditions through the main branching tree and the auxiliary trees. If $\hat{X}$ is an extreme perfect Nash equilibrium, there exists at least a path in the branching tree generated by $E\chi$-MIPerfect leading to it. $\square$

4.2 The EEE-Perfect algorithm

This section illustrates the new exact arithmetics implementation of the *EEE* algorithm and its extension to enumerate all extreme perfect Nash equilibria. The *EEE* algorithm determines all extreme Nash equilibria by enumerating all vertices of X_1 and X_2 that satisfy the complementarity conditions $x_1^t(\alpha_1 e_n - Ax_2) = 0$ and $(\alpha_2 e_m - x_1^t B)x_2 = 0$, where

$$(\hat{x}_1, \hat{\alpha}_2) \in X_1 \equiv \left\{ (x_1, \alpha_2) \in \mathbb{R}^{n+1} : x_1^t B \leq \alpha_2 e_m^t, \ x_1^t e_n = 1, x_1 \geq 0 \right\},$$
$$(\hat{x}_2, \hat{\alpha}_1) \in X_2 \equiv \left\{ (x_2, \alpha_1) \in \mathbb{R}^{m+1} : Ax_2 \leq e_n \alpha_1, \ e_m^t x_2 = 1, \ x_2 \geq 0 \right\}.$$

The *EEE* algorithm considers the parametrized linear programs:

$$R_1(x_2) \equiv \max_{(x_1, \alpha_2) \in X_1} x_1^t Ax_2 - \alpha_2, \qquad \text{and} \qquad R_2(x_1) \equiv \max_{(x_2, \alpha_1) \in X_2} x_1^t Bx_2 - \alpha_1.$$

Indeed, the constraints appearing in the definition of X_1 and X_2 are those of the primal and dual problems. The objective functions $x_1^t Ax_2 - \alpha_2$ and $x_1^t Bx_2 - \alpha_1$ are gaps between the primal and dual objective functions. The structure of the algorithm presented in [1] almost directly applies. Given a pair of current subproblems $\tilde{R}_1$ and $\tilde{R}_2$ at a given node of the search tree, we define the following subproblems:

R_1^i to be $\tilde{R}_1$ in which the constraint $x_{1i} \geq 0$ is replaced by $x_{1i} = 0$,

R_1^j to be $\tilde{R}_1$ in which the constraint $x_1^t B^j \leq \alpha_2$ is replaced by $x_1^t B^j = \alpha_2$,

R_2^j to be $\tilde{R}_2$ in which the constraint $x_{2j} \geq 0$ is replaced by $x_{2j} = 0$,

R_2^i to be $\tilde{R}_2$ in which the constraint $A_i x_2 \leq \alpha_1$ is replaced by $A_i x_2 = \alpha_1$.

We here state the pseudo code of the EEE-Perfect algorithm for enumeration of bimatrix games extreme perfect Nash equilibria.

a. Initialization. Initialize the set of nodes to be explored to $ND = \{(x_1, R_1, R_2)\}$ where x_1 is set to an arbitrary initial value (for example $x_1 = (\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})$). Go to step b.

b. Selection of a Node. If the set ND of nodes to be explored is empty, then stop. Otherwise, choose and remove a node C from ND. If $C = (x_1, \tilde{R}_1, \tilde{R}_2)$ then go to step c, otherwise $C = (x_2, \tilde{R}_1, \tilde{R}_2)$ and go to step d.

c. Direct Feasibility Test (R_2). If $\tilde{R}_2(x_1)$ is infeasible then go to step b, otherwise choose (x_2, α_1) that optimizes $\tilde{R}_2(x_1)$, update (x_1, α_2) that optimizes $\tilde{R}_1(x_2)$ and then go to step e.

d. Direct Feasibility Test (R_1). If $\tilde{R}_1(x_2)$ is infeasible then go to step b, otherwise choose (x_1, α_2) that optimizes $\tilde{R}_1(x_2)$, update (x_2, α_1) that optimizes $\tilde{R}_2(x_1)$ and then go to step e.

e. Branching. Branch on the variable associated with the largest complementarity product $x_{1i}(\alpha_1 - A_i x_2)$ or $(\alpha_2 - x_1^t B^j)x_{2j}$:

$$\text{either} \quad ND = ND \cup \{(x_2, R_1^i, \tilde{R}_2), \ (x_1, \tilde{R}_1, R_2^i)\}$$
$$\text{or} \quad ND = ND \cup \{(x_1, \tilde{R}_1, R_2^j), \ (x_2, R_1^j, \tilde{R}_2)\}.$$

In the case where all complementarity products are already forced to be null, $\hat{X} = (\hat{x}_1, \hat{x}_2)$ is an extreme Nash equilibrium, go to step f, then:

$$\text{let } ND = ND \cup \{(x_2, R_1^i, \tilde{R}_2) : x_{1i} > 0\} \cup \{(x_1, \tilde{R}_1, R_2^i) : A_i x_2 < \alpha_1\}$$
$$\cup \{(x_1, \tilde{R}_1, R_2^j) : x_{2j} > 0\} \cup \{(x_2, R_1^j, \tilde{R}_2) : x_1^t B^j < \alpha_2\}.$$

Go to step b.

f. Solving subproblem Q. Solve $Q(\hat{X})$ to optimality. If the optimal objective of $Q(\hat{X})$ is equal to 0, **add** $\hat{X}$ to NP.

The Steps c and d are considered as the backtracking steps. In each of them, only one problem $\tilde{R}_1$ or $\tilde{R}_2$ is checked for feasibility as only one of them differs from its father. When this problem is feasible, these steps end with feasible strategies x_1 and x_2 which are vertices of X_1 and X_2. Step e creates new nodes in the tree. When the depth of the current node is less than or equal to $n + m$, (i.e. when at least one of the complementarity conditions is not forced to be satisfied), two newsons are created. Otherwise, the corresponding solution defines

equilibrium strategies since all the complementarity conditions are satisfied, and feasibility is insured by Steps c and d; a new son is created for each strictly positive variable or slack variable.

Theorem 4.2 *The algorithm EEE-Perfect enumerates in finite time all extreme perfect Nash equilibria of a bimatrix game.*

Proof The EEE-perfect algorithm seeks all possible ways to satisfy the complementary conditions through its branching rules as shown in [1]. Step f verifies the perfectness of any extreme Nash equilibrium found. Hence, if $\hat{X}$ is an extreme perfect Nash equilibrium of a given bimatrix game, there exists at least one path in the branching tree generated by EEE-Perfect leading to it.   □

5 Computational results

Our computational experiments on randomly generated bimatrix games with different size and density are presented in Table 4. Using C++ implementations of the $E\chi$-MIPerfect and the EEE-Perfect algorithms, these experimental results were obtained under WindowsXP on workstations with 3.3GHz Intel Core i5 vPro processors, and 3.2 GB RAM.

For the entries in Table 4, the column "Size" indicates the original size of the generated set of bimatrix games. The column "d" indicates the density of the generated bimatrix games. For each given size and density, we have randomly generated 30 different bimatrix games. The columns "n" and "m" indicate, respectively, the average number of undominated strategies of the first and second players. The column "ne" indicates the average number of extreme Nash equilibria enumerated. The column "np" indicates the average number of extreme perfect Nash equilibria enumerated. The column "%" indicates the percentage of extreme perfect equilibria with respect to the total number of extreme Nash equilibria. The column "nt" indicates the average number of maximal Selten subsets enumerated. The column "dim" indicates the average dimension of each maximal Selten subset. Finally, the column "time" indicates, on the 30 bimatrix games, the average execution time (in seconds) per game, respectively, for the $E\chi$-MIPerfect algorithm ("Eχ-MIP") and the EEE algorithm ("EEE").

5.1 Discussion

The experimental results show that the average dimension of the maximal Selten subsets increases with the size of the bimatrix game. The results also show that the percentage of extreme perfect equilibria increases with an increase in the bimatrix games density. The bimatrix games density has also an impact on the computing time required by each algorithm to enumerate all extreme perfect Nash equilibria. For both algorithms, $E\chi$-MIPerfect and EEE-Perfect, the computing times increase if the bimatrix games density decreases. Hence, for the $E\chi$-MIPerfect algorithm, the computing time jumps from 269.11 seconds to 839.20 seconds when the 5×5 bimatrix games density drops from 50 to 25 %. Similarly, for the EEE-Perfect algorithm, the computing time jumps from 57.44 to 180.29 s when the 5×5 bimatrix games density drops from 50 to 25 %. The sparsest case where the bimatrix games density is 25 % appears to be the most time consuming for both algorithms for every given bimatrix game' size. This could be explained by the fact that the high number of zeros in the payoff matrices reduces the number of dominated strategies and increases the number of extreme Nash equilibria by offering a number of strategies with the same zero payoff for each player. This typical behavior was already observed for the floating arithmetics implementations of the $E\chi$-MIP and the EEE algorithms by Audet et al. [2], and for the exact arithmetics implementation of the original EEE algorithm by Avis et al. [4]. Finally, the experimental results show that the EEE algorithm is on average much faster than the $E\chi$-MIP algorithm. This can be explained by the fact that our exact arithmetics implementation of the $E\chi$-MIP algorithm uses both of the Branch and Bound and the Simplex algorithms to optimize the $0-1$ mixed linear programs while the EEE algorithm just uses the Simplex algorithm implementation. However, the $E\chi$-MIP algorithm is still, to the best of our knowledge, the only algorithm able to enumerate polymatrix games (more than 2 players) extreme Nash equilibria. A future work would be to explore the possibility of extending the EEE algorithm to polymatrix games and improve our current implementation.

Table 4 Computational results on random bimatrix games ($d = 25, 50, 75, 100\%$)

Size	n	m	ne	np	%	nt	Dim	Time Exχ-MIP	EEE	n	m	ne	np	%	nt	Dim	Time Exχ-MIP	EEE
	d = 0.25									*d* = 0.50								
2 × 2	1.80	1.83	2.57	1.97	76.62	1.1	2.1	0.97	0.87	1.70	1.63	1.57	1.27	80.81	1.4	1.3	0.70	0.67
3 × 3	3.00	3.00	5.10	2.00	39.47	1.6	2.2	14.27	5.57	2.73	2.67	2.70	1.56	58.02	1.8	1.4	7.53	3.11
5 × 5	4.70	5.00	10.30	2.48	24.10	2.5	2.3	839.20	180.29	4.90	4.87	6.10	2.83	46.45	2.6	2.5	269.11	57.44
7 × 7	7.00	7.00	24.21	8.12	33.54	3.1	3.5	17,402.43	2411.23	7.00	6.80	10.37	5.48	52.84	3.3	3.9	6352.17	402.88
	d = 0.75									*d* = 1.00								
2 × 2	1.60	1.60	1.47	1.20	81.82	1.0	1.2	0.60	0.35	1.60	1.57	1.33	1.33	100.0	1.6	1.5	0.80	0.43
3 × 3	2.60	2.27	1.93	1.63	84.48	1.7	1.6	5.00	1.77	2.40	2.47	1.77	1.67	94.34	1.9	1.6	5.57	1.28
5 × 5	4.63	4.63	3.47	2.63	75.96	2.4	2.1	237.91	22.25	4.60	4.60	2.43	2.37	97.26	3.5	3.1	236.43	10.42
7 × 7	6.77	6.65	4.46	4.38	97.77	3.8	3.6	4,435.92	83.10	6.63	6.96	4.86	4.82	99.07	3.9	3.7	5106.45	88.33

6 Conclusion

The book of Pardalos, Migdalas and Pitsoulis, "Pareto Optimality, Game Theory and Equilibria" [16], surveys recent developments and numerous applications of game theory and equilibrium programming in economics and engineering applications. In this paper, we used maximal cliques enumeration to find all maximal Nash subsets corresponding to a bimatrix game. We used dimension relation defined for maximal Nash and Selten subsets to implement an automatic procedure to compute their dimensions. We presented a linear programming approach to identify extreme perfect Nash equilibria, enumerate all Selten maximal subsets and compute their dimensions. Based on this linear formulation, we proposed the $E\chi$-MIPerfect and the EEE-Perfect algorithms which enumerate all extreme perfect Nash equilibria. The implementations of these algorithms allowed us to generate experimental results on bimatrix games with different size and density. These results show that the probability of meeting a perfect equilibrium profile increases with an increase in the bimatrix game's density.

Conflict of interest The authors declare that there is no conflict of interest regarding the publication of this article.

References

1. Audet, C.; Hansen, P.; Jaumard, B.; Savard, G.: Enumeration of all extreme equilibrium strategies of bimatrix games. SIAM J. Sci. Comput. **23**(1), 323–338 (2001)
2. Audet, C.; Belhaiza, S.; Hansen, P.: Enumeration of all extreme equilibria in game theory: bimatrix and polymatrix games. J. Optim. Theory Appl. **129**(3), 349–372 (2006)
3. Audet, C.; Belhaiza, S.; Hansen, P.: A new sequence form approach for the enumeration and refinement of all extreme Nash equilibria for extensive form games. Int. Game Theory Rev. **11**(4), 437–451 (2009)
4. Avis, D.; Rosenberg, G.D.; Savani, R.; von Stengel, B.: Enumeration of Nash equilibria for two-player games. Econ. Theory **42**, 9–37 (2010)
5. Belhaiza, S.; Audet, C.; Hansen, P.: On proper refinement of Nash equilibria for bimatrix games. Automatica, **48**, 297–303 (2012)
6. Borm, P.E.M.; Jansen, M.J.M, Potters, J.A.M.; Tijs, S.H.: On the structure of the set of perfect equilibria in bimatrix games. O-R Spektrum **15**, 17–20 (1993)
7. Bron, C.; Kerbosch, J.: Algorithm 457: finding all cliques of an undirected graph. Commun. ACM **16**(9), 575–579 (1973)
8. Heuer, G.A.; Millham, C.B.: On Nash subsets ad mobility chains in bimatrix games. Naval Search Logist. Q. **23**, 311–319 (1976)
9. Jansen, M.J.M.: Maximal Nash subsets for bimatrix games. Naval Search Logist. Q. **28**(1), 147–152 (1981)
10. Mangasarian, O.L.: Equilibrium points of bimatrix games. J. Soc. Ind. Appl. Math. **12**, 778–780 (1964)
11. Mangasarian, O.L.; Stone, H.: Two person nonzero-sum games and quadratic programming. J. Math. Anal. Appl. **9**, 348–355 (1964)
12. Millham, C.B.: On Nash subsets of bimatrix games. Naval Res. Logist. Q. **74**, 307–317 (1974)
13. Mills, H.: Equilibrium points in finite games. J. Soc. Ind. Appl. Math. **8**, 397–402 (1960)
14. Myerson, R.B.: Game Theory: Analysis of Conflict. Harvard University Press, Cambridge, London (1997)
15. Nash, J.F.: Equilibrium points in n-person games. Proc. Natl. Acad. Sci. **36**, 48–49 (1950)
16. Pardalos, P.M.; Migdalas, A.; Pitsoulis, L.: Pareto Optimality, Game Theory and Equilibria. Springer Optimization and Its Applications, vol. 17 (2008)
17. Selten, R.: Reexamination of the perfectness concept for equilibrium points in extensive games. Int. J. Game Theory **4**, 25–55 (1975)
18. van Damme, E.E.C.: Refinements of the Nash Equilibrium Concept, Lecture Notes in Economics and Mathematical Systems vol. 219. Springer, Berlin Heidelberg, New York (1983)
19. Von Stengel. B.; Van Den Elzen, A.; Talman, D.: Computing normal form perfect equilibria for extensive two-person games. Econometrica, **70**, 693–715 (2002)
20. Vorob'ev, N.N.: Equilibrium points in bimatrix games. Theoriya Veroyatnostej i ee Primwneniya, **3**, 297–309 (1958). [English version: Theory of Probability and its Applications, **3**, 297–309 (1958)]