International Journal of Game Theory manuscript No.
(will be inserted by the editor)

The Computational Complexity of Evolutionarily Stable Strategies

K. Etessami[1], **A. Lochbihler**[2]

[1] Laboratory for Foundations of Computer Science, School of Informatics, University of Edinburgh, Edinburgh EH9 3JZ, Scotland, UK, e-mail: `kousha@inf.ed.ac.uk`

[2] Fakultät für Mathematik und Informatik, Universität Passau, 94030 Passau, Germany, e-mail: `lochbihl@fmi.uni-passau.de`

The date of receipt and acceptance will be inserted by the editor

Abstract The concept of *evolutionarily stable strategies* (ESS) has been central to applications of game theory in evolutionary biology, and it has also had an influence on the modern development of game theory. A *regular* ESS is an important refinement the ESS concept. Although there is a substantial literature on computing evolutionarily stable strategies, the precise computational complexity of determining the existence of an ESS in a symmetric two-player strategic form game has remained open, though it has been speculated that the problem is **NP**-hard. In this paper we show that determining the existence of an ESS is both **NP**-hard and **coNP**-hard, and that the problem is contained in $\mathbf{\Sigma_2^p}$, the second level of the polynomial time hierarchy. We also show that determining the existence of a regular ESS is indeed **NP**-complete. Our upper bounds also yield algorithms for computing a (regular) ESS, if one exists, with the same complexities.

Key words computational complexity – game theory – evolutionarily stable strategies – evolutionary biology – Nash equilibria

1 Introduction

Game theoretic methods have been applied for a long time to study phenomena in evolutionary biology, most systematically since the pioneering work of Maynard Smith in the 1970's and 80's ([SP73, May82]). Since then *evolutionary game theory* has been used to understand a diverse range of sometimes counter-intuitive phenomena in biology, and it has also had an important influence on the modern development of game theory (see, e.g., [vD91, Wei97, HS98]). For an overview of evolutionary game theory and a sampling of its many applications in zoology and botany, see the survey by Hammerstein and Selten [HS94]. They mention among other applications, analysis of animal fighting and mating, offspring sex ratios, and flower size.

A central concept in evolutionary game theory is the notion of an evolutionarily stable strategy (ESS) in a symmetric two-player strategic form game, introduced by Maynard Smith and Price ([SP73]). An ESS is a particular kind of mixed (randomized) strategy, where the probabilities in the mixed strategy are now viewed as denoting percentages in a population exhibiting different possible behaviors. To be an ESS, a mixed strategy s must first constitute a Nash equilibrium, (s, s), when played against itself. This means that s is a "best response" to itself, i.e., that the expected payoff for a player who plays s against s is the maximum possible payoff of any strategy against s. Secondly, to be an ESS, s must in a precise sense be "impervious to invasion" by other strategies. Specifically, it must be the case that if a different strategy t is also a best response to s, then the expected payoff of playing s against t must be strictly greater than the payoff of playing t against t.

It was shown already by Nash [Nas51] that every symmetric strategic form game contains a symmetric Nash equilibrium (s, s). However, not all symmetric 2-player games contain an ESS: rock-paper-scissors gives a simple counterexample (see below). Thus, one may ask: what is the computational complexity of determining whether an ESS exists in a 2-player strategic game (with, say, rational payoffs)? And, if an ESS does exist, what is the complexity of actually computing one? The complexity of computing an arbitrary Nash equilibrium for a 2-player strategic form game is a well-known open problem (see, e.g., [Pap01]). It is computable in **NP** (as a function), but neither known to be **NP**-hard nor known to be computable in polynomial time. However, **NP**-hardness is known for computing Nash equilibria that satisfy any of several additional desirable conditions, such as equilibria that optimize "social welfare", and this is so even for symmetric games ([GZ89, CS03]). It has thus been speculated that finding an ESS may also be **NP**-hard, but no proof was known.

A *regular* ESS is an important refinement of the ESS concept. This is an ESS, s, where the "support set" of s, i.e., the set of pure strategies that are played with non-zero probability in s, already contains all pure strategies that are best responses to s. There are several equivalent definitions of regular ESSs. Harsanyi [Har73b] introduced regular equilibria as a refinement of the Nash equilibrium concept, and showed that "almost all" strategic form games contain only regular equilibria, where "almost all" here means that the games with irregular equilibria constitute a set of measure zero in a suitably defined measure space on games. There are other, weaker refinements of Nash equilibria, such as "quasi-strict" equilibrium, also introduced by Harsanyi [Har73a]. For symmetric 2-player games, it turns out that the definition of a regular ESS coincides with that of an ESS that is a quasi-strict Nash equilibrium. Other equivalent formulations of regular ESSs make the notion rather robust (see, e.g., [vD91, Sel83, Bom86]). See van Damme's excellent book [vD91] for a comprehensive treatment of refinements of Nash equilibria, and their ramifications for evolutionarily stable strategies.

As a simple example of ESSs, consider a parametrized (not necessarily zero-sum) version of rock-papers-scissors, which has the payoff bi- matrix:

$$\begin{pmatrix} (a,a) & (1,-1) & (-1,1) \\ (-1,1) & (a,a) & (1,-1) \\ (1,-1) & (-1,1) & (a,a) \end{pmatrix}$$

with the parameter $a \in \mathbb{R}$. One can show that, for $a < 1$, there is exactly one symmetric Nash equilibrium, namely $s = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, and that for $a \geq 1$ there are precisely three additional symmetric NEs: the pure strategies $s^1 = (1, 0, 0)$, $s^2 = (0, 1, 0)$, and $s^3 = (0, 0, 1)$. Note that any strategy t is a best response to s. Hence, for s to be an ESS, playing s against t must yield higher utility, which is $\frac{a}{3}$ for all t, than playing t against itself. The payoff for t against itself can, for $a \geq 0$, be at most a with equality holding iff $t \in \{s^1, s^2, s^3\}$, and for $a < 0$, it can be at most $\frac{a}{3}$ with equality holding iff $t = s$. Thus, s is an ESS iff $a < 0$. If so, s is regular because its support contains all pure strategies. When we look at the pure strategies s^1 to s^3, we must distinguish between $a > 1$ and $a = 1$. For $a > 1$, s^i is the only best response to s^i, therefore s^i is a regular ESS too ($1 \leq i \leq 3$). In the case $a = 1$, the extended support of s^i is $\{i, j\}$ where $j = ((i + 1) \mod 3) + 1$. Playing s^j against itself gives payoff 1 whereas playing s^i against s^j gives payoff -1. Thus, s^i is not an ESS for $i \in \{1, 2, 3\}$, when $a = 1$. All in all, we have one regular ESS for $a < 0$, no ESS for $0 \leq a \leq 1$ and three regular ESSs for $a > 1$. These games have no non-regular ESSs. If, however, the payoff for (s^3, s^3) was changed to (b, b) instead of (a, a) with $b < -1$ and $a = 1$, then s^1 would be the only non-regular ESS. This is because (s^1, s^1) is a symmetric NE and for any best response $t = (t_1, 0, 1 - t_1)$ to s^1, we have that the payoff of playing s^1 against t is $2t_1 - 1$ and that of playing t against t is $t_1^2 + b(1 - t_1)^2$. Now, $2t_1 - 1 > t_1^2 + b(1 - t_1)^2$ for all $0 \leq t_1 < 1$ iff $b < -1$. Of course, this analysis of ESSs only applies to these special parametrized rock-paper-scissors games.

In this paper we show that determining the existence of an ESS in a given symmetric 2-player strategic form game with rational payoffs is both **NP**-hard and **coNP**-hard under polynomial-time many-one reductions, and thus that it is not in **NP**, nor in **coNP**, unless **NP**=**coNP**. Furthermore, as an upper bound, we show that determining the existence of an ESS is contained in $\mathbf{\Sigma_2^P}$, the second level of the polynomial time hierarchy. (See section 2.1 for background on computational complexity.) On the other hand, we show that determining the existence of a regular ESS is **NP**-complete. Our upper bounds also yield algorithms to compute a (regular) ESS, if one exists, with the same complexities. From our bounds it also follows that computing the number of (regular) ESSs is **#P**-hard (**#P**-complete, respectively).

Our **NP**-hardness result for ESSs provides a reduction from SAT that yields a 1-1 correspondence between satisfying assignments of a CNF boolean formula and the ESSs in the game to which it is reduced. This is reminiscent of, but substantially different from, the reduction of Conitzer and Sandholm [CS03] for Nash equilibria. Furthermore, these ESSs will all be regular, and therefore **NP**-hardness for regular ESSs also follows. For **coNP**-hardness of the ESS problem, we provide a reduction from coCLIQUE to the ESS problem. In doing so, we make essential use of a classic characterization of maximum clique size via quadratic programs, due to Motzkin and Straus [MS65]. From our hardness results, we also easily derive an inapproximability result for (regular) ESSs.

Our upper bounds combine criteria for the existence of ESSs based on quadratic forms, together with known results about the complexity of quadratic programming decision problems. There is a substantial literature on computing evolutionarily stable strategies, and on its connections to mathematical programming (see, e.g., [Bom92,BP89,Bom02], and see [MWC+97] for a different computational perspective based on dynamics). In particular, Bomze [Bom92] developed criteria for ESSs, based on copositivity of a matrix over a cone, and uses these to provide an algorithm for enumerating all ESSs in a game. His criteria build on earlier criteria for ESSs developed by Haigh ([Hai75]) and Abakuks ([Aba80]). Bomze's enumeration algorithm uses a recursive elimination procedure that involves some complications including a possible numerical blowup issue (see Section 4 for an explanation). We were thus unable to deduce our $\mathbf{\Sigma_2^P}$ upper bounds for ESSs directly from Bomze's algorithms. We instead provide a self-contained development of all the criteria we need, directly building variants of the Haigh-Abakuks criteria and Bomze's criteria, and we then employ a result by Vavasis [Vav90] on the computational complexity of the quadratic programming decision problem, to obtain our $\mathbf{\Sigma_2^P}$ upper bounds for ESSs. For regular ESSs, our **NP** upper bound follows from simpler modifications of the Haigh-Abakuks criteria, together with basic facts from matrix theory.

The plan of the paper is as follows. Section 2 provides definitions and gives some brief background on computational complexity theory. Section 3 provides hardness results for both ESSs and regular ESSs. Section 4 provides our upper bounds for both. We conclude in Section 5.

2 Definitions and Notation

For a $n \times n$-matrix A, and subsets $I, J \subseteq \{1, \ldots, n\}$, let $A_{I,J}$ denote the submatrix of A defined by deleting the rows with indexes not in I and deleting columns with indexes not in J. Likewise, for (row) vector x, define $x_I := x_{I,\{1\}}$ ($x_{\{1\},I}$), viewing x as a $n \times 1$-matrix ($1 \times n$-matrix, respectively). Let A^T denote the transpose of A. Likewise, for x^T. Unless stated otherwise, we assume all vectors are column vectors. A real symmetric $n \times n$-matrix A is **positive definite** if $x^T Ax > 0$ for all $x \in \mathbb{R}^n - \{\mathbf{0}\}$. Recall the determinant criterion for positive definiteness: a symmetric matrix A is positive definite if and only if $\det(A_{I,I}) > 0$ for all $I = \{1, \ldots, i\}$, $1 \leq i \leq n$, where $\det$ denotes the determinant of a square matrix (see, e.g., [LT85]). Thus, in particular, positive definiteness of a rational symmetric matrix can be detected in polynomial time. A real symmetric matrix A is called **negative definite** if $(-A)$ is positive definite. Note, for any A and x, $x^T Ax = x^T A'x$, where $A' := \frac{1}{2}(A + A^T)$ is a symmetric matrix. We thus say a general $n \times n$ matrix A is *positive (negative) definite* if A' is positive (negative) definite, and we can use the determinant criterion on A' to detect this.

We now recall some basic definitions of game theory (see, e.g., [OR94]). A **finite two-player strategic form game** $\Gamma = (S_1, S_2, u_1, u_2)$ is given by finite sets of strategies S_1 and S_2 and utility, or *payoff*, functions $u_1 : S_1 \times S_2 \mapsto \mathbb{R}$ and $u_2 : S_1 \times S_2 \mapsto \mathbb{R}$ for player one and two, respectively. Such a game is called **symmetric** if $S_1 = S_2 =: S$ and $u_1(i,j) = u_2(j,i)$ for all $i,j \in S$. We are only concerned with symmetric 2-player games in this paper, so we write (S, u_1) as shorthand for (S, S, u_1, u_2), with $u_2(j,i) = u_1(i,j)$ for $i,j \in S$. For simplicity assume $S = \{1, \ldots, n\}$, i.e., pure strategies are identified with integers $i, 1 \leq i \leq n$.

In what follows we only consider finite symmetric two-player strategic form games. The **payoff matrix** $A_\Gamma = (a_{i,j})$ of $\Gamma = (S, u_1)$ is given by $a_{i,j} = u_1(i,j)$ for $i,j \in S$. (Note that A_Γ is not necessarily symmetric, even if Γ is a symmetric game.) A **mixed strategy** $s = (s(1), \ldots, s(n))^T$ for $\Gamma = (S, u_1)$ is a vector that defines a probability distribution on S. Thus, $s \in X$, where $X = \left\{s \in \mathbb{R}^n_{\geq 0} : \sum_{i=1}^n s(i) = 1\right\}$ denotes the set of mixed strategies in Γ. s is called **pure** iff $s(i) = 1$ for some $i \in S$. In that case we identify s with i. For brevity, we generally use "strategy" to refer to a mixed strategy s, and indicate otherwise when the strategy is pure. In our notation, we alternatively view a mixed strategy s as either a vector $(s_1, \ldots, s_n)^T$, or as a function $s : S \mapsto \mathbb{R}$, depending on which is more convenient in the context.

The **expected payoff** function, $U_k : X \times X \mapsto \mathbb{R}$ for player $k \in \{1,2\}$ is given by $U_k(s,t) = \sum_{i,j \in S} s(i)t(j)u_k(i,j)$, for all $s,t \in X$. Note that $U_1(s,t) = s^T A_\Gamma t$ and $U_2(s,t) = s^T A_\Gamma^T t$. Let s be a strategy for $\Gamma = (S, u_1)$. A strategy $t \in X$ is a **best response** to s if $U_1(t,s) = \max_{t' \in X} U_1(t', s)$. The **support** $\mathrm{supp}(s)$ of s is the set $\{i \in S : s(i) > 0\}$ of pure strategies which are played with non-zero probability. The **extended support** $\text{ext-supp}(s)$ of s is the set $\{i \in S : U_1(i,s) = \max_{x \in X} U_1(x,s)\}$ of all pure best responses to s.

A pair of strategies (s,t) is a **Nash equilibrium (NE)** for Γ if s is a best response to t and t is a best response to s. Note that (s,t) is a NE if and only if $\mathrm{supp}(s) \subseteq \text{ext-supp}(t)$ and $\mathrm{supp}(t) \subseteq \text{ext-supp}(s)$. A NE (s,t) is **symmetric** if $s = t$. It was shown already in [Nas51] that every symmetric game contains a symmetric NE.

Definition 1 *A mixed strategy $s \in X$ in a 2-player symmetric game Γ is an **evolutionarily stable strategy (ESS)** of Γ if:*

1. *(s,s) is a symmetric Nash equilibrium of Γ, and*
2. *if $t \in X$ is any best response to s and $t \neq s$, then $U_1(s,t) > U_1(t,t)$.*

*An ESS s is **regular** if $\mathrm{supp}(s) = \text{ext-supp}(s)$.*

2.1 Background from computational complexity

For the benefit of readers unfamiliar with computational complexity theory we briefly review some of its basic concepts and definitions. For proper treatments please see, e.g., the books [GJ79, Pap94].

A yes/no decision problem can be described by a set of strings L (i.e., a *language*) over a finite alphabet Σ, by using a fixed encoding scheme. Namely, $L \subseteq \Sigma^*$ contains exactly those strings $w \in \Sigma^*$ that encode problem instances for which the decision answer is “yes”. Of course, there are many possible encoding schemes for the same problem, some more concise than others. The encodings we use for problems we consider in this paper are standard, and will either be made explicit or will be clear from the context. Turing Machines (TMs) are a universal model of computation, and can be used as acceptors of languages. A deterministic Turing Machine (DTM), M, is said to accept an input string $w \in \Sigma^*$ iff the computation of M on input w halts in a designated accepting state, and otherwise it is said to reject w. A non-deterministic Turing Machine (NTM) is said to accept an input w iff there exists at least one computation of M on w that halts in an accepting state. For a decision problem L, we say that a TM M decides L, or accepts L, if on all inputs w all computations of M halt, and furthermore for every $w \in \Sigma^*$, M accepts w if and only if $w \in L$. For a function $f : \mathbb{N} \mapsto \mathbb{N}$, we say that a TM, M, is $f(n)$-time bounded, if for $w \in \Sigma^*$ every computation of M on w takes no more than $f(n)$ steps to halt, where $n = |w|$ is the length of the string w. A decision problem L is said to have (non-deterministic) time complexity at most f if there exists a $f(n)$-time bounded (non-deterministic) Turing machine M that decides L. Let $\mathrm{DTIME}(f)$ ($\mathrm{NTIME}(f)$) denote the set of all languages with (non-deterministic) time complexity at most f.

Standard complexity classes like **P** and **NP** consist of a set of decision problems, i.e., a set of languages, namely those that are accepted by Turing machines with given resource constraints. The complexity classes we consider in this paper are as follows: $\mathbf{P} = \bigcup\{\mathrm{DTIME}(f) | f \text{ a polynomial in } n\}$ is the set of all languages L for which a polynomial-time bounded DTM exists that decides L. $\mathbf{NP} = \bigcup\{\mathrm{NTIME}(f) | f(n) \text{ a polynomial in } n\}$ is the non-deterministic analog of **P**. Equivalently, $L \in \mathbf{NP}$ iff there is a polynomial-time DTM M and a polynomial $p(n)$ such that for all $x \in \Sigma^*$, $x \in L$ iff there exists a $y \in \Sigma^{p(|x|)}$ such that M accepts the pair (x, y). $\mathbf{coNP} = \{\Sigma^* - L | L \subseteq \Sigma^* \wedge L \in \mathbf{NP}\}$ is the class of complements of languages in **NP**. $\mathbf{\Sigma_2^P}$ is the second level of the polynomial-time hiearchy, and can be defined as the set of all languages L for which there exists a polynomial-time DTM, M, and polynomials $p_1(n)$ and $p_2(n)$ such that for all $x \in \Sigma^*$, $x \in L$ iff there exists a $y \in \Sigma^{p_1(|x|)}$ such that for all $z \in \Sigma^{p_2(|x|)}$, M accepts the triple (x, y, z).

By contrast to these classes of decision problems, **#P** is a complexity class for counting problems. These classes consist of a set of functions $g : \Sigma^* \mapsto \mathbb{N}$. A counting problem g is in **#P** iff there exists a polynomial-time bounded NTM, M, such that the number of accepting computations of M on input w is equal to $g(w)$.

For a decision complexity class C, a decision problem (encoded as language $L \subseteq \Sigma^*$) is said to be C**-hard** iff there is a polynomial-time many-one (a.k.a., Karp) reduction from every problem L' in C to L. A polynomial-time many-one reduction from $L' \subseteq \Gamma^*$ to $L \subseteq \Sigma^*$, is a function $f : \Gamma^* \mapsto \Sigma^*$ computable in polynomial time by a DTM such that $x \in L'$ iff $f(x) \in L$. If such a reduction exists, we say that L' is (many-one P-time) reducible to L and denote it by $L' \leq_p L$. If $L \in C$ and L is C-hard we say that L is C**-complete**. Note that polynomial-time many-one reductions are transitive, i. e. if $L \leq_p L'$ and $L' \leq_p L''$, then also $L \leq_p L''$. Hence, to show that L is C-hard it is sufficient to show $L' \leq_p L$ for some already known C-hard problem L'. Appropriate notions of reduction, hardness, and completeness also exist for counting classes such as **#P**. In particular, a "parsimonious" polynomial-time many-one reduction, f, from one counting problem, g to another counting problem h is one that preserves the number of solutions, i.e., $g(x) = h(f(x))$. There are **#P**-complete problems under such reductions, such as #SAT (see below). For more details see, e.g., [Pap94, Val79].

In this paper, we look at the following decision and counting problems: let ESS (REG-ESS) denote the decision problem of whether a symmetric 2-player game Γ, specified by a rational payoff matrix A_Γ for player 1, has at least one (regular) evolutionarily stable strategy. The encoding used for rational values is the standard one: numerator and denominator are given in binary. Let #ESS (#REG-ESS) denote the counting problem of computing how many (regular) ESSs the game Γ has.

Our reductions will involve some standard known complete problems for various complexity classes. Here we recall some of them. As usual, an undirected graph $G = (V, E)$ has vertices V and a symmetric edge set $E \subset V \times V$ where $(i, j) \in E \Rightarrow (j, i) \in E$, and $(i, i) \notin E$, for all $i, j \in V$. Let A_G denote the symmetric adjacency matrix of undirected graph G. A **clique** $C \subseteq V$ of $G = (V, E)$ is a subset of V such that $(C \times C) - E = \{(i, i) \mid i \in C\}$. Let $\omega(G)$ denote the maximum cardinality of a clique in G. Let coCLIQUE $= \{(G, c) \mid c \in \mathbb{N}$ and $\omega(G) < c\}$. Thus coCLIQUE denotes the decision problem of, given an undirected graph G and $c \in \mathbb{N}$, determining whether G does <u>not</u> have a clique of size c. coCLIQUE is **coNP**-complete. The satisfiability problem SAT (#SAT) asks whether there exists a satisfying assignment (or how many satisfying assignments there are, respectively) for a given Boolean formula in conjunctive normal form. SAT is **NP**-complete and #SAT is **#P**-complete.

3 Hardness results

3.1 ESS *is* **coNP**-*hard*

We first show that ESS is **coNP**-hard by providing a polynomial-time (many-one) reduction from coCLIQUE to ESS. In doing so, we make essential use of the following classic result due to Motzkin and Straus [MS65].

Theorem 1 [MS65] *Let $G = (V, E)$ be an undirected graph with maximum clique size d. Let $\Delta_1 = \left\{x \in \mathbb{R}_{\geq 0}^{|V|} : \sum_{i=1}^{|V|} x_i = 1\right\}$. Then $\max_{x \in \Delta_1} x^T A_G x = \frac{d-1}{d}$.*

Note that one direction of this theorem is immediate: given a clique C of size d of G, if we choose x to be the vector in which $x_i = 1/d$ if and only if vertex i is in the clique C, then it is easily checked that $x^T A_G x = \frac{d-1}{d}$, and thus $\max x \in \Delta_1 x^T A_G x \geq \frac{d-1}{d}$.

Corollary 1 *Let $G = (V, E)$ be an undirected graph with maximum clique size d and let $l \in \mathbb{R}_{\geq 0}$. Let $\Delta_l = \left\{x \in \mathbb{R}_{\geq 0}^{|V|} : \sum_{i=1}^{|V|} x_i = l\right\}$. Then $x^T A_G x \leq \frac{d-1}{d} l^2$ for all $x \in \Delta_l$.*

$$
\begin{array}{c|cccccccc|ccc}
 & 1 & 2 & \cdot & \cdot & \cdot & \cdot & n-1 & n & a & b & c \\
\hline
1 & & & & & & & & & 1 & 0 & 0 \\
2 & & & & & & & & & 1 & 0 & 0 \\
\cdot & & & & & & & & & \cdot & \cdot & \cdot \\
\cdot & & & & A_G & & & & & \cdot & \cdot & \cdot \\
\cdot & & & & & & & & & \cdot & \cdot & \cdot \\
\cdot & & & & & & & & & \cdot & \cdot & \cdot \\
n-1 & & & & & & & & & 1 & 0 & 0 \\
n & & & & & & & & & 1 & 0 & 0 \\
\hline
a & \frac{(k-1)}{k} & \frac{(k-1)}{k} & \cdot & \cdot & \cdot & \cdot & \cdot & \frac{(k-1)}{k} & 1 & 0 & 0 \\
b & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 0 & 0 & 0 \\
c & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 0 & 0 & 0
\end{array}
$$

Fig. 1 The payoff matrix for player 1 in the game $\Gamma_k(G)$.

Proof For $l = 0$, $\Delta_0 = \{\mathbf{0}\}$ and thus $\mathbf{0}^T A_G \mathbf{0} = 0$. Suppose $l > 0$. Let $x \in \Delta_l$ and set $y = \frac{1}{l}x$. Then $y \in \Delta_1$, and by Theorem 1, $x^T A_G x = (ly)^T A_G (ly) = l^2 y^T A_G y \leq l^2 \frac{d-1}{d}$. □

Definition 2 *For an undirected graph $G = (V, E)$ and $k \in \mathbb{N}$ define the game $\Gamma_k(G) = (S, u_1)$ where*

- $S = V \cup \{a, b, c\}$ *are the strategies for the players where* $a, b, c \notin V$.
- *The payoffs (to player 1) are given by:*
  - $u_1(i, j) = 1$ *for all* $i, j \in V$ *with* $(i, j) \in E$.
  - $u_1(i, j) = 0$ *for all* $i, j \in V$ *with* $(i, j) \notin E$.
  - $u_1(z, a) = 1$ *for all* $z \in S - \{b, c\}$.
  - $u_1(a, i) = \frac{k-1}{k}$ *for all* $i \in V$.
  - $u_1(y, i) = 1$ *for all* $y \in \{b, c\}$ *and* $i \in V$.
  - $u_1(y, a) = 0$ *for all* $y \in \{b, c\}$.
  - $u_1(z, y) = 0$ *for all* $z \in S$ *and* $y \in \{b, c\}$.

In other words, the payoff matrix for player 1 looks like the matrix depicted in Figure 1 (where the submatrix A_G denotes the adjacency matrix of the graph G).

The idea behind this matrix is that only a can be a ESS (dependent on $k < d$). a is not an ESS iff there is no clique C of size (at least) k in G because playing all strategies from C with equal probability is a best response to a that violates the stability requirement for a being an ESS.

Theorem 2 *Let $G = (V, E)$ be an undirected graph. $\Gamma_k(G)$ has an ESS iff G has no clique of size k. Thus,* ESS *is* **coNP***-hard.*

Proof Let $G = (V, E)$ be an undirected graph with maximum clique size d. We consider the game $\Gamma_k(G)$. We first show that any ESS s of $\Gamma_k(G)$ must satisfy $\mathrm{supp}(s) \cap \{b, c\}$, $\mathrm{supp}(s) \not\subseteq V$, and $s(a) = 1$. Next, we show that (a, a) is in fact a NE and that a is an ESS iff $d < k$.

Suppose s is an ESS of $\Gamma_k(G)$. Then $\mathrm{supp}(s) \cap \{b, c\} = \emptyset$, because if not let $t \neq s$ be a strategy with $t(i) = s(i)$ for $i \in V$, $t(y) = s(b) + s(c)$ and $t(y') = 0$ where $y, y' \in \{b, c\}$ such that $y \neq y'$ and $s(y) = \min\{s(b), s(c)\}$. Since $u_1(b, z) = u_1(c, z)$ for all $z \in S$,

$$U_1(t,s) = \sum_{i \in V} \underbrace{t(i)}_{=s(i)} U_1(i,s) + \underbrace{(t(b)+t(c))}_{=s(b)+s(c)} \underbrace{U_1(b,s)}_{=U_1(c,s)} = U_1(s,s)$$

and so t is a best response to s. An identical argument shows that $U_1(s, t) = U_1(t, t)$, but this is a contradiction to s being an ESS. Furthermore, $\mathrm{supp}(s) \not\subseteq V$, because if not, by Theorem 1

$$U_1(s,s) = \sum_{i,j \in V} s(i)s(j)u_1(i,j) = x^T A_G x \le \frac{d-1}{d} < 1 = U_1(b,s)$$

where $x = (s(v_1), \ldots, s(v_{|V|}))^T \in \Delta_1$ and so (s, s) is not a NE.

Thus $s(a) > 0$. Suppose for contradiction $s(a) < 1$. Since (s, s) is a NE, a is a best response to s and $a \neq s$. Then $U_1(s, a) = \sum_{z \in \mathrm{supp}(s)} s(z)u_1(z, a) = 1 = U_1(a, a)$, which is a contradiction to s being an ESS. Therefore the only possible ESS of $\Gamma_k(G)$ is a. (a, a) is a symmetric NE because $u_1(z, a) \le 1 = u_1(a, a)$ for all $z \in S$. (Notice $\mathrm{supp}(a) \neq \text{ext-supp}(a)$, thus a is never regular.)

Suppose $d < k$. Let $t \neq a$ be a best response to a. Then $\mathrm{supp}(t) \subseteq V \cup \{a\}$. Let $r = \sum_{i \in V} t(i)$. So $r > 0$ and $t(a) = 1 - r$. So using Corollary 1:

$$\begin{aligned}
U_1(t,t) - U_1(a,t) = &\underbrace{\sum_{i,j \in V} t(i)t(j)u_1(i,j)}_{\le \frac{d-1}{d} r^2} + r \cdot t(a)1 + t(a) \cdot r\frac{k-1}{k} + t(a)^2 \cdot 1 \\
&- \left(r \cdot \frac{k-1}{k} + t(a) \cdot 1 \right) \\
\le &\frac{d-1}{d} r^2 - \frac{k-1}{k} r^2 + r(1-r) + (1-r)^2 - (1-r) \\
= &\left(\frac{d-1}{d} - \frac{k-1}{k} \right) r^2 = \frac{r^2}{dk}(d-k) < 0
\end{aligned}$$

So a is an ESS. Now suppose $d \ge k$. Let $C \subseteq V$ be a clique of G of size k. Then t with $t(i) = \frac{1}{k}$ for $i \in C$ and $t(j) = 0$ for $j \in S - C$ is a best response to a and $t \neq a$, but $U_1(t, t) = \sum_{i,j \in C} t(i)t(j)u_1(i, j) = \frac{1}{k^2} \cdot (k-1)k \cdot 1 = \frac{k-1}{k} = U_1(a, t)$, so a is not an ESS. $\square$

3.2 ESS *and* REG-ESS *are both* **NP**-*hard*

We now show ESS is **NP**-hard by providing a polynomial-time (many-one) reduction from SAT to ESS. Moreover, the same reduction shows that REG-ESS is **NP**-hard. First, two key lemmas that construct our matrix gadgets:

Lemma 1 *Let $n \in \mathbb{N}$ and $k \in \mathbb{R}_{\geq 0}$. Let $A = (a_{i,j})$ be the $n \times n$-matrix in which all entries are 1 except all diagonal entries which are all 0. Consider the mapping $f : \mathbb{R}^n \mapsto \mathbb{R}$, $f(x) = x^T Ax$. Then, the only maximum of f subject to $\sum_{i=1}^n x_i = k$ is $x^* = \left(\frac{k}{n}, \frac{k}{n}, \ldots, \frac{k}{n}\right)^T$ with $f(\frac{k}{n}, \ldots, \frac{k}{n}) = \frac{n-1}{n}k^2$.*

Proof Note $f(x) = \sum_{i=1}^n x_i \sum_{\substack{j=1 \\ j\neq i}}^n x_j$. Since $\sum_{j=1}^n x_j = k$,

$$f(x) = \sum_{i=1}^n x_i(k - x_i) = k\sum_{i=1}^n x_i - \sum_{i=1}^n x_i^2 = k^2 - \sum_{i=1}^n x_i^2.$$

Let $\langle x, y\rangle = \sum_i x_i y_i$ denote the standard inner product of vectors x and y. Let $\mathbf{1} = (1, \ldots, 1)^T$ denote the all 1 vector of length n. We thus want to minimize $\langle x, x\rangle = \sum_{i=1}^n x_i^2$, subject to $\langle x, \mathbf{1}\rangle = k$. It is easy to see that x^* is the unique such minimum. For completeness, we provide a proof. Suppose $\langle y, \mathbf{1}\rangle = \langle x^*, \mathbf{1}\rangle = k$. Note, for any vector x, $\langle x, x^*\rangle = \frac{k}{n}\langle x, \mathbf{1}\rangle$. Now,

$$\begin{aligned}\langle y, y\rangle - \langle x^*, x^*\rangle &= \langle y, y\rangle - \langle x^*, x^*\rangle + 2\frac{k}{n}\langle x^*, \mathbf{1}\rangle - 2\frac{k}{n}\langle y, \mathbf{1}\rangle \\ &= \langle y, y\rangle + \langle x^*, x^*\rangle - 2\langle y, x^*\rangle \;=\; \langle y - x^*, y - x^*\rangle \geq 0\end{aligned}$$

Moreover, $\langle y - x^*, y - x^*\rangle = 0$ if and only if $y = x^*$. Thus, x^* is the unique minimum. □

Lemma 2 *Let $n \in \mathbb{N}$ and $k \in \mathbb{R}_{\geq 0}$. Let $B = (b_{i,j})$ be the $2n \times 2n$-matrix where, for $i, j \in \{1, \ldots, 2n\}$:*

$$b_{i,j} = \begin{cases} 0 & \textit{if } i = j \\ -2 & \textit{if } j = i + 1 \textit{ and } i \textit{ is odd} \\ -2 & \textit{if } i = j + 1 \textit{ and } i \textit{ is even} \\ 1 & \textit{otherwise} \end{cases}$$

In other words, the matrix B looks as follows:

$$B = \begin{pmatrix} 0 & -2 & 1 & 1 & \cdots & 1 & 1 & 1 \\ -2 & 0 & 1 & 1 & \cdots & 1 & 1 & 1 \\ 1 & 1 & 0 & -2 & \cdots & 1 & 1 & 1 \\ 1 & 1 & -2 & 0 & \cdots & 1 & 1 & 1 \\ \vdots & & & & \ddots & & & \vdots \\ 1 & 1 & 1 & 1 & \cdots & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & \cdots & 1 & 0 & -2 \\ 1 & 1 & 1 & 1 & \cdots & 1 & -2 & 0 \end{pmatrix}$$

Consider the mapping $f : \mathbb{R}^{2n}_{\geq 0} \mapsto \mathbb{R}, f(x) = x^T Bx$. Then,
$x^* = (x^*_1, x^*_2, \ldots, x^*_{2n}) \in \mathbb{R}^{2n}_{\geq 0}$ is a global maximum of f subject to $\sum_{i=1}^{2n} x_i = k$ if and only if it satisfies $x^*_{2i+1} + x^*_{2i+2} = \frac{k}{n}$ and $x^*_{2i+1}x^*_{2i+2} = 0$ for all $0 \leq i < n$. In that case, $f(x^*) = \frac{n-1}{n}k^2$.

Proof Note that

$$f(x) = \sum_{i=0}^{n-1} \left(\left(\sum_{\substack{j=0 \\ j \neq i}}^{n-1} (x_{2i+1} + x_{2i+2})(x_{2j+1} + x_{2j+2}) \right) - 4x_{2i+1}x_{2i+2} \right)$$

Suppose, for contradiction, that x^* is a global maximum but that for some $i \in \{0, \ldots, n-1\}$, $x^*_{2i+1} > 0$ and $x^*_{2i+2} > 0$. Let x' be identical to x^* except that $x'_{2i+1} = x^*_{2i+1} + x^*_{2i+2}$, and $x'_{2i+2} = 0$. Note that x' satisfies the constraints $\sum_{j=1}^{2n} x'_j = k$, and $x' \geq 0$. However, $f(x') > f(x^*)$, because $(x'_{2j+1} + x'_{2j+2}) = (x^*_{2j+1} + x^*_{2j+1})$ for all $j = 0, \ldots, n-1$, but $4x^*_{2i+1}x^*_{2i+1} > 4x'_{2i+1}x'_{2i+2} = 0$. Contradiction. Therefore at any global maximum x^*, $x^*_{2i+1}x^*_{2i+2} = 0$, for all $i = 0, \ldots, n-1$. Consider such a vector x^*. Let I be the set of indices such that for each $i = 0, \ldots, n-1$, exactly one of $2i+1$ and $2i+2$ is in I and such that $x^*_j = 0$ for every index j that is not in I. Note that for any such x^*, $f(x^*) = (x^*)^T Bx^* = (x^*)^T_I B_{I,I} x^*_I$. Note that $B_{I,I}$ has exactly the form of matrix A of Lemma 1, and that $\langle x^*_I, \mathbf{1} \rangle = k$. Therefore, by Lemma 1 the unique maximum of $(x)^T B_{I,I} x$, subject to $\langle x, \mathbf{1} \rangle = k$, is $x^*_I = (\frac{k}{n}, \ldots, \frac{k}{n})^T$. From this the statement of Lemma 2 follows. $\square$

Definition 3 *Let Φ be a Boolean formula in Conjunctive Normal Form (CNF), $V = \{x_1, \ldots, x_n\}$ the set of its variables, $L = \{x_1, \neg x_1, x_2, \neg x_2, \ldots, x_n, \neg x_n\}$ the set of literals over V, and $C = \{c_1, \ldots, c_r\} \subseteq 2^L - \{\emptyset\}$ the set of clauses of Φ (the empty clause is not allowed). The function $v : L \mapsto V$ gives the variable corresponding to a literal, e.g. $v(x_1) = v(\neg x_1) = x_1$. Define the function $\chi : C \times L \mapsto \{\frac{n-1}{n}, -1\}$ as follows:*

$$\chi(c, l) = \begin{cases} \frac{n-1}{n} & \text{if } l \notin c \\ -1 & \text{if } l \in c \end{cases}$$

Define the game $\Gamma(\Phi) = (S, u_1)$ where:

- *$S = L \cup C$ are the strategies for player 1 and 2 and*
- *the payoffs are given by*
  - *$u_1(l_1, l_2) = 1$ for all $l_1, l_2 \in L$ with $v(l_1) \neq v(l_2)$.*
  - *$u_1(l, l) = 0$ for all $l \in L$.*
  - *$u_1(l, \neg l) = -2$ for all $l \in L$.*
  - *$u_1(l, c) = -1$ for all $l \in L$ and all $c \in C$.*
  - *$u_1(c, l) = \chi(c, l)$ for all $c \in C$ and $l \in L$.*
  - *$u_1(c, c') = -1$ for all $c, c' \in C$.*

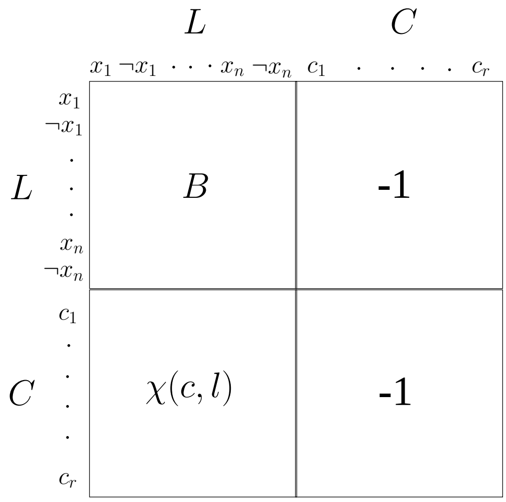

Fig. 2 The payoff matrix for player 1 in the game $\Gamma(\Phi)$.

In other words, the payoff matrix for player 1 is the matrix depicted in Figure 2. There, the submatrix B is the matrix defined in Lemma 2, the submatrices marked by "-1" denote matrices all of whose entries are -1, and the submatrix marked by "$\chi(c,l)$" denotes a $(|C| \times |L|)$-matrix whose (c,l)-entry is $\chi(c,l)$.

Let $(l_1, \dots, l_n)$, with $l_i \in \{x_i, \neg x_i\}$, correspond to a truth assignment to the variables in V.

Theorem 3 *Let Φ be a CNF Boolean formula with n variables. If $(l_1, \dots, l_n)$ corresponds to a satisfying assignment for Φ, then the mixed strategy s with $s(l_i) = \frac{1}{n}$ for $1 \le i \le n$ and $s(y) = 0$ for $y \in S - \{l_1, \dots, l_n\}$ is a regular ESS for the game $\Gamma(\Phi)$. Conversely, if s is an ESS for $\Gamma(\Phi)$, then s has the above form and $(l_1, \dots, l_n)$ corresponds to a satisfying assignment of Φ (and thus s is also a regular ESS).*

Proof Let Φ be a CNF Boolean formula with n variables. We consider the strategic game $\Gamma(\Phi)$. The idea behind $\Gamma(\Phi)$ is that only strategies which correspond to truth assignments to the variables in V are potentially an ESS. Such a strategy s does not satisfy a clause c (and thus Φ) iff playing c is a best response to s that violates the stability condition for s.

Let s be an ESS. First, we show that $\mathrm{supp}(s) \cap C = \emptyset$. Assume not. Then, there is a clause $c \in C$ such that $s(c) > 0$. If $s(c) = 1$, then any literal l of c is a best response to s since $U_1(l,s) = u_1(l,c) = -1 = u_1(c,c) = U_1(s,s)$, but $U_1(l,l) = u_1(l,l) = 0 > -1 = u_1(c,l) = U_1(c,l)$, a contradiction to s being an ESS. So suppose $0 < s(c) < 1$. Since s is a NE, we know that $c \neq s$ is a best response to s and

$$U_1(s,c) = \sum_{x \in S} s(x) \underbrace{u_1(x,c)}_{=-1} = -1 = u_1(c,c) = U_1(c,c)$$

contradicting s being an ESS. Next, we show that $v(\mathrm{supp}(s)) = V$, i.e. for each variable at least one corresponding literal is played. Assume not. Then, there is a literal $l \in L$ such that $s(l) = 0$ and $s(\neg l) = 0$. Enumerating the literals in such a way that $l^{2i+1} = x_i$ and $l^{2i+2} = \neg x_i$ for all $0 \leq i < n$, let $B = (b_{i,j})_{1 \leq i,j \leq 2n}$ be the $2n \times 2n$-matrix where $b_{i,j} = u_1(l^i, l^j)$ and $s' = (s(l^1), \dots, s(l^{2n}))^T$. Note B is the matrix B in Lemma 2. So by Lemma 2, $\frac{n-1}{n} \geq s'^T B s' = \sum_{i,j=1}^{2n} s'_i b_{i,j} s'_j = \sum_{i,j=1}^{2n} s(l^i) s(l^j) u_1(l^i, l^j) = U_1(s,s)$. But, $U_1(l,s) = 1 \cdot \sum_{l \in \mathrm{supp}(s)} s(l) = 1 > \frac{n-1}{n} \geq U_1(s,s)$, so s is not a NE.

Next, we show that if s is an ESS, then there are n pairwise different literals $(l_1, \dots, l_n)$ such that $s(l_i) = \frac{1}{n}$ and $l_i \neq \neg l_j$ for $1 \leq i,j \leq n$. Suppose not. Since $v(\mathrm{supp}(s)) = V$, we can pick n pairwise different literals $(l'_1, \dots, l'_n)$ such that $l'_i \in \mathrm{supp}(s)$ and $l'_i \neq \neg l'_j$ for $1 \leq i,j \leq n$. Set $t(l'_i) = \frac{1}{n}$ for $1 \leq i \leq n$ and $t(i) = 0$ for all $i \in S - \{l'_1, \dots, l'_n\}$. Since (s,s) is a NE, every $l \in \mathrm{supp}(s)$ is a best response to s, i.e. $U_1(l,s) = U_1(s,s)$. Hence $U_1(t,s) = \sum_{i=1}^{n} t(l'_i) \sum_{j \in S} s(j) u_1(l'_i, j) = \frac{1}{n} \sum_{i=1}^{n} U_1(l'_i, s) = U_1(s,s)$, so t is a best response to s. Then

$$\begin{aligned} U_1(s,t) = &\sum_{l,l' \in L} s(l) t(l') u_1(l,l') = \sum_{l',l \in L} t(l') s(l) u_1(l',l) = U_1(t,s) \\ =&U_1(s,s) \leq \frac{n-1}{n} \end{aligned} \tag{1}$$

and

$$\begin{aligned} U_1(t,t) =& \sum_{i=1}^{n} t(l'_i) \left(t(l'_i) u_1(l'_i, l'_i) + \sum_{\substack{j=1 \\ j \neq i}}^{n} t(l'_j) u_1(l'_i, l'_j) \right) \\ =& \sum_{i=1}^{n} \frac{1}{n} \left(\frac{1}{n} \cdot 0 + \sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{1}{n} \cdot 1 \right) = \frac{n-1}{n} \end{aligned} \tag{2}$$

so $U_1(s,t) \leq U_1(t,t)$, contradicting s being an ESS.

What remains to be shown is that if s is a mixed strategy such that $s(l_i) = \frac{1}{n}$ for n different $l_i \in L$ with $l_i \neq \neg l_j$ for all $1 \leq i,j \leq n$ then s is an ESS iff $(l_1, \dots, l_n)$ is a satisfying assignment for Φ (and that s is then in fact a regular ESS). Suppose s is such a mixed strategy. First, we show that (s,s) is a symmetric NE. We know from equation 2 that $U_1(s,s) = \frac{n-1}{n}$. Let $L^* = \{l_1, \dots, l_n\}$. Playing any of the $l \in L^*$ gives payoff $U_1(l,s) = \frac{1}{n} \cdot 0 + \frac{1}{n}(n-1) \cdot 1 = \frac{n-1}{n}$. Playing any of the $l \in L - L^*$ gives payoff $U_1(l,s) = \frac{1}{n} \cdot (-2) + \frac{1}{n}(n-1) \cdot 1 < \frac{n-1}{n}$. Playing any of the $c \in C$ gives payoff at most $U_1(c,s) = \frac{1}{n} n \cdot \frac{n-1}{n} = \frac{n-1}{n}$.Therefore, (s,s) is a symmetric NE.

Suppose $(l_1, \dots, l_n)$ is not a satisfying assignment. Then, there is a clause c such that none of its literals is played. Therefore, $U_1(c,s) = \sum_{l \in L^*} s(l) u_1(c,l) =$

$\sum_{l \in L^*} s(l)\frac{n-1}{n} = \frac{n-1}{n}$. So c is a best response to s and $c \neq s$. Then $U_1(c,c) = -1 = \sum_{l \in L^*} s(l)u_1(l,c) = U_1(s,c)$, so s is not an ESS. Conversely, suppose $(l_1,\ldots,l_n)$ is a satisfying assignment. Then, every clause contains a literal that is played. Hence, for all $c \in C$,

$$\begin{aligned} U_1(c,s) = \sum_{l \in L^*} s(l)u_1(c,l) = (-1)\sum_{l \in L^* \cap c} s(l) + \frac{n-1}{n}\sum_{l \in L^* - c} s(l) \\ < \frac{n-1}{n} = U_1(s,s). \end{aligned}$$

Thus $\mathrm{supp}(s) = \text{ext-supp}(s) = L^*$. So, suppose t is a best response to s. Then $\mathrm{supp}(t) \subseteq \text{ext-supp}(s) = L^*$. As in equation (1), we get $U_1(s,t) = U_1(t,s) = U_1(s,s) = \frac{n-1}{n}$. Let $A = (a_{i,j})_{1 \leq i,j \leq n}$ be the $n \times n$-matrix where $a_{i,j} = u_1(l_i,l_j)$ and let $t' = (t(l_1),\ldots,t(l_n))^T$. Note that A is the matrix A in Lemma 1, so by Lemma 1

$$\frac{n-1}{n} \geq t'^T A t' = \sum_{i,j=1}^{n} t'_i a_{i,j} t'_j = \sum_{i,j=1}^{n} t(l_i)t(l_j)u_1(l_i,l_j) = U_1(t,t)$$

with equality holding only if $t(l_i) = \frac{1}{n}$ for all $1 \leq i \leq n$. Hence, we have that $t \neq s$ implies $U_1(s,t) > U_1(t,t)$. Therefore, s is an ESS, and it is regular because $\mathrm{supp}(s) = \text{ext-supp}(s)$. $\square$

Corollary 2 ESS *and* REG-ESS *are* **NP**-*hard. (Moreover,* #ESS *and* #REG-ESS *are* **#P**-*hard.)*

Proof Clearly, $\Gamma(\Phi)$ can be constructed from Φ in **P**-time. Theorem 3 shows $(l_1,\ldots,l_n)$ is a satisfying assignment for Φ iff s with $s(l_i) = \frac{1}{n}$ for $1 \leq i \leq n$ and $s(y) = 0$ for $y \in S - \{l_1,\ldots,l_n\}$ is an ESS for $\Gamma(\Phi)$, and that these are the only possible ESSs. Moreover, it shows that in this case s is a regular ESS. Therefore, both ESS and REG-ESS are **NP**-hard.
The number of (necessarily regular) ESSs in $\Gamma(\Phi)$ is the number of satisfying assignments of Φ. Counting satisfying assignments of a CNF formula is **#P**-hard ([Val79]). $\square$

3.3 Inapproximability of ESSs

We now address whether an ESS, if one exists, can be efficiently "approximated". Care is needed to define this, since no ESS may exist. One formulation is a polynomial time algorithm that, given the game and $\epsilon > 0$, outputs a mixed strategy s such that *if* there exists a (regular) ESS, then there exists a (regular) ESS s^* such that $\|s^* - s\| < \epsilon$, under some norm $\|\cdot\|$. For concreteness, let $\|s\| = \max_{i \in \{1,\ldots,n\}} |s_i|$ be the L_∞ norm (any norm L_j, $j \geq 1$, is fine too). Call this a **P**-time ϵ-approximation of (regular) ESSs.

Corollary 3 *There is no polynomial time $\frac{1}{m}$-approximation algorithm for finding an ESS in a game $\Gamma = (S, u_1)$ where $m = |S|$, nor for finding a regular ESS in Γ, unless* **P** = **NP**.

Proof Suppose there was such an algorithm. For a boolean formula Φ, we run that algorithm on the game $\Gamma(\Phi) = (S, u_1)$ from definition 3 with $|S| = m = 2|V| + |C|$, where $|V| = n$ is the number of variables of Φ, and $|C|$ is the number of clauses. This would yield a strategy s such that if there exists a (regular) ESS in $\Gamma(\Phi)$, then there exists s^* with $\|s^* - s\| < \frac{1}{m}$. Thus $|s_i^* - s_i| < \frac{1}{m}$ for all $1 \leq i \leq m$. Note however that by Theorem 3, the only candidate (regular) ESSs s^* in that game have, in every coordinate, either probability $\frac{1}{|V|} = \frac{1}{n} > \frac{2}{m}$ or probability 0. Thus if $s_i > \frac{1}{m}$, then the only possible candidate for s_i^* is $s_i^* = \frac{1}{n}$, and if $s_i < \frac{1}{m}$, then the only possible candidate is $s_i^* = 0$. If $s_i = \frac{1}{m}$, then neither is a candidate and hence s is not within distance $< \frac{1}{m}$ of any ESS, therefore no ESS exists. So, we can build the candidate s^*, check that the probabilities in it sum to 1, and that it corresponds to a truth assignment to variables, meaning exactly one of the two pure strategies corresponding to the two literals for each variable has non-zero probability, and no other strategy has non-zero probability. We then check whether this is a satisfying assignment of Φ. If so, Φ is satisfiable, otherwise Φ is not. Thus we would have solved SAT in **P**-time using our purported approximation algorithm. (An obvious variant of this corollary can be phrased for randomized polynomial time $\frac{1}{m}$-approximation of ESSs.) □

Note that corollary 3 does not rule out the possibility of polynomial time ϵ-approximation algorithms, for arbitrarily small but fixed constants $\epsilon > 0$, as the size m goes to infinity.

4 Upper bounds

4.1 REG-ESS *is in* **NP**

In [Hai75], Haigh claimed to show that a strategy s is an ESS for $\Gamma = (S, u_1)$ if and only if (s, s) is a NE and the $(m-1) \times (m-1)$-matrix $C = (c_{i,j})$ is negative definite, where $m = |\text{ext-supp}(s)|$ and $c_{i,j} = u_1(i,j) + u_1(m,m) - u_1(i,m) - u_1(m,j)$ for $i, j \in \text{ext-supp}(s) - \{m\}$ (where, w.l.o.g., $\text{ext-supp}(s) = \{1, \ldots, m\}$). In [Aba80], Abakuks pointed out that there is an error in the "only if" part of Haigh's claim. Namely, Abakuks showed that the existence of an ESS only implies the negative definiteness of the matrix C if in addition $s(m) > 0$ and $|\text{ext-supp}(s)| - |\text{supp}(s)| \leq 1$. As we will see, the Haigh-Abakuks criteria can be used to show that REG-ESS is in **NP**. By a suitable modification of these criteria, we can obtain necessary and sufficient conditions for the existence of arbitrary ESSs which will allow us to show that ESS is in $\mathbf{\Sigma_2^P}$. Essentially identical conditions, based on copositivity of matrices over a cone, were developed by Bomze and used by him in an algorithm for enumerating all ESSs of a game (compare Theorem 6 below with [Bom92]'s Theorem 3.2, whose proof relies on the substantial developments in the book [BP89]). Bomze's enumeration algorithm, however, uses a recursive elimination procedure that involves some complications including a possible numerical difficulty. Namely, we could not preclude the possibility that iterating the procedure outlined in Theorem 3.3 of [Bom92] may cause an exponential blow-up in numerical values. We were thus unable to deduce our upper bounds for ESS directly from Bomze's algorithms. We will instead give a self-contained development of the criteria we shall use, with elementary proofs building directly on the work of [Hai75] and [Aba80], and we will then (in the case of ESS) rely on a well known result by Vavasis about the complexity of the quadratic programming decision problem ([Vav90]), multiple applications of which allows us to obtain our upper bounds for ESS.

Lemma 3 (cf. [Aba80], Lemma 1) *Let* $m \in \mathbb{N}$, $x \in \mathbb{R}^m_{\geq 0}$ *with* $\sum_{i=1}^m x_i = 1$. *Let*

$$Y_x = \left\{y \in \mathbb{R}^m_{\geq 0} : \textstyle\sum_{i=1}^m y_i = 1\right\} - \{x\} \text{ and}$$
$$Z_x = \{z \in \mathbb{R}^m : \textstyle\sum_{i=1}^m z_i = 0, \forall 1 \leq i \leq m : x_i = 0 \Rightarrow z_i \geq 0\} - \{\mathbf{0}\}$$

Let $B \in \mathbb{R}^{m \times m}$. *Then* $z^T B z < 0$ *for all* $z \in Z_x$ *iff* $(y-x)^T B (y-x) < 0$ *for all* $y \in Y_x$.

Lemma 1 in [Aba80] says the following: if $Y = \left\{y \in \mathbb{R}^m_{\geq 0} : \sum_{i=1}^m y_i = 1\right\}$ and $Z = \{z \in \mathbb{R}^m : z \neq 0, \sum_{i=1}^m z_i = 0\}$ and B is a real $m \times m$-matrix, then

– $z^T B z < 0$ for all $z \in Z$ implies that $(x-y)^T B (x-y) < 0$ for all $y \in Y$ with $y \neq x$ and
– if at most one component of x is zero then $(x-y)^T B (x-y) < 0$ for all $y \in Y$ with $y \neq x$ implies that $z^T B z < 0$ for all $z \in Z$.

Lemma 3, which we now prove, is a variation of Abakuks' Lemma 1.

Proof Suppose $z^T B z < 0$ for all $z \in Z_x$. Let $y \in Y_x$. Then $y - x \neq \mathbf{0}$, $\sum_{i=1}^m (y_i - x_i) = \sum_{i=1}^m y_i - \sum_{i=1}^m x_i = 1 - 1 = 0$ and for all $1 \leq i \leq m$ with $x_i = 0$ we get $y_i - x_i = y_i \geq 0$, hence $y - x \in Z_x$ and so $(y-x)^T B (y-x) < 0$. Conversely, suppose $(y-x)^T B (y-x) < 0$ for all $y \in Y_x$. Let $z \in Z_x$. Set $\lambda = \min\left\{\frac{x_i}{|z_i|} : 1 \leq i \leq m, x_i > 0, z_i \neq 0\right\}$. Then $\lambda > 0$. Choose $y = x + \lambda z \neq x$. Then $y \geq 0$ because $x \geq 0$ and if $z_i < 0$ then $x_i > 0$ and $y_i = x_i - \lambda |z_i| \geq x_i - \frac{x_i}{|z_i|}|z_i| = 0$ for $1 \leq i \leq m$. Note that

$$\sum_{i=1}^m y_i = \sum_{i=1}^m x_i + \lambda \sum_{i=1}^m z_i = 1 + \lambda \cdot 0 = 1$$

Hence $y \in Y_x$ and thus $z^T B z = \left(\frac{1}{\lambda}(y-x)\right)^T B \left(\frac{1}{\lambda}(y-x)\right) = \frac{1}{\lambda^2}(y-x)^T B (y-x) < \frac{1}{\lambda^2} 0 = 0$. $\square$

Lemma 4 [Hai75] *Let* $C = (c_{i,j})_{i,j}$ *be a real* $m \times m$*-matrix,* $m \geq 2$. *Let* $D = (d_{i,j})$ *be the* $(m-1) \times (m-1)$*-matrix given by* $d_{i,j} = c_{i,j} + c_{m,m} - c_{i,m} - c_{m,j}$. *Let* $x \in \mathbb{R}^m$ *such that* $\sum_{i=1}^m x_i = 0$ *and set* $x' = x_{\{1,\ldots,m-1\}}$. *Then* $x^T C x = x'^T D x'$.

Proof A proof for this lemma was given by Haigh in [Hai75]. For completeness, we provide it here. Let $x \in \mathbb{R}^m$ such that $\sum_{i=1}^{m} x_i = 0$, i. e. $x_m = -\sum_{i=1}^{m-1} x_i$. Then

$$\begin{aligned} x^T C x &= \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} x_i c_{i,j} x_j + x_m \left(\sum_{j=1}^{m-1} c_{m,j} x_j + \sum_{i=1}^{m-1} x_i c_{i,m} \right) + x_m^2 c_{m,m} \\ &= \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} \left(x_i c_{i,j} x_j + (-x_i) c_{m,j} x_j + x_i c_{i,m} (-x_j) + (-x_i) c_{m,m} (-x_j) \right) \\ &= \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} x_i \left(c_{ij} + c_{mm} - c_{mj} - c_{im} \right) x_j = x'^T D x' \end{aligned}$$

□

Lemma 5 [Hai75] *Let (s, s) be a symmetric NE for the game $\Gamma = (S, u_1)$ and let $M = \text{ext-supp}(s)$, with $m = |M|$. Let $x = s_M$ and $C = (A_\Gamma)_{M,M}$. Let Y_x be defined as in Lemma 3. Then s is an ESS if and only if $(y-x)^T C (y-x) < 0$ for all $y \in Y_x$.*

Proof A proof for this can be found in [Hai75]. For completeness, we provide it here, too. Let t be any best response to s. Consider

$$\begin{aligned} U_1(t,t) - U_1(s,t) =& t^T A_\Gamma t - s^T A_\Gamma t = (t-s)^T A_\Gamma (t - s + s) \\ =& (t-s)^T A_\Gamma (t-s) + \underbrace{t^T A_\Gamma s - s^T A_\Gamma s}_{= U_1(t,s) - U_1(s,s) = 0} \\ =& (t-s)^T A_\Gamma (t-s) \end{aligned}$$

Note that $t(z) = s(z) = 0$ for $z \in S - M$. Let $y = t_M$. Then $(t-s)^T A_\Gamma (t-s) = (y-x)^T C (y-x)$.

Suppose s is an ESS. Then for any $y' \in Y_x$, let $t' \in X$ with $t'(z) = y'(z)$ for $z \in M$ and $t'(z) = 0$ for $z \in S - M$. Then $t' \neq s$ is a best response to s, because $\text{supp}(t)$ is contained in $M = \text{ext-supp}(s)$. Thus, $(y'-x)^T C (y'-x) = (t'-s)^T A_\Gamma (t'-s) = U_1(t',t') - U_1(s,t') < 0$.

Conversely, suppose $(y-x)^T C (y-x) < 0$ for all $y \in Y_x$. For any best response $t \neq s$ to s, set $y'' = t_M$. Then $y'' \in Y_x$ and so $U_1(t,t) - U_1(s,t) = (y''-x)^T C (y''-x) < 0$. □

Lemma 6 (cf. [Bom92], Thm 3.2) *Let (s, s) be a NE for Γ with $M = \text{ext-supp}(s)$, and $m = |M| \geq 2$. Identify M with $\{1, \ldots, m\}$ such that $s(m) > 0$ and let $x = s_M$. Let $C = (A_\Gamma)_{M,M}$. Define D as in Lemma 4. Let*

$$W_x = \left\{ w \in \mathbb{R}^{m-1} : (\forall i \in \{1, \ldots, m-1\} : x_i = 0 \Rightarrow w_i \geq 0) \right\} - \{\mathbf{0}\}$$

Then s is an ESS if and only if $w^T D w < 0$ for all $w \in W_x$.

Proof Let Y_x and Z_x be defined as in Lemma 3. From Lemma 5 we know that $U_1(t,t) - U_1(s,t) < 0$ for all best responses $t \neq s$ to s is equivalent to $(y-x)^T C(y-x) < 0$ for $y \in Y_x$ which itself is equivalent to $z^T Cz < 0$ for all $z \in Z_x$ by Lemma 3. Now suppose that $w^T Dw < 0$ for all $w \in W_x$. Let $z \in Z_x$. Set $w' = z_{\{1,\ldots,m-1\}}$. Then $w' \in W_x$ and so with Lemma 4, we get $z^T Cz = w'^T Dw' < 0$. Conversely, suppose that $z^T Cz < 0$ for all $z \in Z_x$. Let $w \in W_x$. Set $z'_i = w_i$ for $1 \leq i \leq m-1$ and $z'_m = -\sum_{i=1}^{m-1} w_i$. Then $z' = (z'_1, \ldots, z'_m)^T \in Z_x$ because $s(m) > 0$ and so with Lemma 4, we get $w^T Dw = z'^T Cz' < 0$. □

Proposition 1 *If s is an ESS, (t,t) a symmetric NE, and $\mathrm{supp}(t) \subseteq \text{ext-supp}(s)$, then $t = s$.*

Proof Suppose $t \neq s$. Since $\mathrm{supp}(t) \subseteq \text{ext-supp}(s)$, t is a best response to s, but since (t,t) is a NE, $U_1(s,t) \leq U_1(t,t)$. Contradiction to s being an ESS. □

Theorem 4 REG-ESS *is in* **NP**.

Proof Given a game $\Gamma = (S, u_1)$ $(n = |S|)$ with rational payoffs, guess the extended support set $M \subseteq S$ of a (purported) regular ESS s for Γ and let $m = |M|$. Identify S with $\{1, \ldots, n\}$ such that $M = \{1, \ldots, m\}$. Find a symmetric NE (s,s) of Γ with $\mathrm{supp}(s) \subseteq M$ by solving (in **P**-time) the following linear program in variables $s_1, \ldots, s_n, w$, where $s = (s_1, \ldots, s_n)^T$:

– $U_1(i,s) = w$ for all $i \in M$; and $U_1(i,s) \leq w$ for all $i \in S - M$.
– $\sum_{i=1}^{n} s_i = 1$; and $s_i \geq 0$ for all $i \in M$; and $s_i = 0$ for all $i \in S - M$.

Let s be an arbitrary solution. By proposition 1 if s is an ESS then s is the only solution to the system above. Thus it doesn't matter what solution we find (if we don't find any, then there is no NE and no ESS with support set M). Check that $\mathrm{supp}(s) = \text{ext-supp}(s) = M$. This can be done in **P**-time, by trying each pure strategy outside $\mathrm{supp}(s)$ against s.

Note that if $|M| = 1$ then the pure strategy s is the only best response to itself, and thus s is a regular ESS. Suppose $|M| \geq 2$, and let $x = s_M$. Let D and W_x be defined as in Lemma 6. By Lemma 6, s is an ESS iff $w^T(-D)w > 0$ for all $w \in W_x$. Set $D' = \frac{1}{2}\left(D + D^T\right)$. D' is a symmetric matrix, and note that $w^T D' w = w^T Dw$ for all w. Note that $W_x = \mathbb{R}^{m-1} - \{\mathbf{0}\}$ because $\mathrm{supp}(s) = M$. Hence s is an ESS if and only if $(-D')$ is positive definite. Positive definiteness of a symmetric matrix can be checked in **P**-time via the determinant criterion (see section 2). Therefore checking whether there is an ESS s for Γ with $\mathrm{supp}(s) = \text{ext-supp}(s) = M$ for the guessed set M can be done in polynomial time. Thus REG-ESS is in **NP**. □

Corollary 4 #REG-ESS *is* **#P***-complete.*

Proof **#P**-hardness was established in Corollary 2. The proof of Theorem 4 gives an NP-algorithm for deciding whether a game has a regular ESS. Each accepting computation yields a different support set and thus a different regular ESS. Therefore, #REG-ESS is in **#P**. □

4.2 ESS *is in* $\mathbf{\Sigma_2^P}$

We next show ESS is in $\mathbf{\Sigma_2^P}$. Vavasis established the following result on quadratic programming (see also, e.g., [MK87]).

Definition 4 *Let* QP *denote the following decision version of the quadratic programming problem: given a $n \times n$-matrix H and a $m \times n$-matrix A, both with integer coefficients, $K \in \mathbb{Q}$, $c \in \mathbb{Z}^m$, and $b \in \mathbb{Z}^m$, is there a vector $x \in \mathbb{R}^n$ with $Ax \geq b$ such that $x^T Hx + c^T x \leq K$?*

Theorem 5 [Vav90] QP *is in* **NP**.

Theorem 6 ESS *is in* $\mathbf{\Sigma_2^P}$.

Proof Given a game $\Gamma = (S, u_1)$ ($n = |S|$) with rational payoffs, guess the extended support set $M \subseteq S$ for an ESS s for Γ and set $m = |M|$. As in the proof of Theorem 4, compute a symmetric NE (s, s) with $\mathrm{supp}(s) \subseteq M$. Check that $\mathrm{ext\text{-}supp}(s) = M$ (again, this check is **P**-time). Set $l = m - |\mathrm{supp}(s)|$. If $l = 0$ then proceed as in the algorithm in Theorem 4.

Suppose $l > 0$, and thus $m \geq 2$. Let $x = s_M$. Let D and W_x be defined as in Lemma 6. By Lemma 6, s is an ESS if and only if $w^T(-D)\,w > 0$ for all $w \in W_x$. In other words, s is not an ESS iff there exists $w \in W_x$ such that $w^T(-D)\,w \leq 0$. This is the case iff there exists $w \neq \mathbf{0}$ such that $w_i \geq 0$ for all i such that $x_i = 0$, such that $w^T(-D)\,w \leq 0$. This is the case iff

> there exists a w such that $w_i \geq 0$ for all i where $x_i = 0$, and such that for some $j \in \{1, \ldots, m-1\}$, $w_j \geq 1$ or $-w_j \geq 1$, and such that $w^T(-D)\,w \leq 0$.

To see the last claim, note that if $w^T(-D)w \leq 0$, then for any constant $c > 0$, $(cw)^T(-D)(cw) = c^2 w^T(-D)w \leq 0$. Thus, for $w \neq \mathbf{0}$ where $w^T(-D)w \leq 0$, we can choose a constant $c > 0$ large enough so that either for some positive coefficient w_j, $cw_j \geq 1$ or for some negative coefficient w_j, $-(cw_j) \geq 1$. Thus the vector (cw) will satisfy the desired conditions.

Now, we can check these conditions by solving $2(m-1)$ quadratic programming decision problems. Namely, we check for all $1 \leq j \leq m-1$ and for each $\sigma \in \{+1, -1\}$, whether there exists a $w \in \mathbb{R}^{m-1}$ satisfying $w^T(-D)w \leq 0$, and satisfying the linear constraints: $w_i \geq 0$ for each i such that $x_i = 0$, and $\sigma w_j \geq 1$. As described, the matrix $(-D)$ is a rational matrix, and the QP problem was formulated in terms of integer matrices. However, we can easily "clear denominators" in $(-D)$, finding the least common multiple $\lambda > 0$ of the denominators of entries of D and setting $H = -\lambda D$ (this can be done in **P**-time). Then H is a $(m-1) \times (m-1)$-matrix with integer entries, and $w^T Hw \leq 0$ if and only if $w^T(-D)w \leq 0$, for any $w \in \mathbb{R}^{m-1}$.

Thus, checking that this s is <u>not</u> an ESS can be done in **NP**. Thus, to determine the existence of an ESS involves existentially guessing a support set M, finding s with support set M such that (s, s) is a NE (using linear programming), and then checking that s is an ESS in **coNP**, by checking (in **NP**) that s is not an ESS. Thus ESS is in $\mathbf{\Sigma_2^P}$. $\square$

(We remark that it follows from the proof of Theorem 6 that #ESS is in the counting class **#NP**, a class defined in [Val79].)

5 Concluding remarks

We have shown that the ESS problem is both **NP**-hard and **coNP**-hard under many-one reductions, and thus not in **NP** nor in **coNP** unless **NP** $=$ **coNP**, and that it is contained in $\mathbf{\Sigma_2^p}$, the second level of the polynomial-time hierarchy. On the other hand, we have shown REG-ESS is **NP**-complete.

Our results leave open whether the general ESS problem is $\mathbf{\Sigma_2^p}$-complete or belongs to some "intermediate" class above **NP** and **coNP** but below $\mathbf{\Sigma_2^p}$. Natural decision problems in such a position are relatively rare, lending the ESS problem an intriguing complexity-theoretic status.

In a recent note, Nisan [Nis06] has slightly strengthened our hardness theorem for general ESSs by showing that the ESS problem is hard for the complexity class **coDP**.[1] **coDP** consists of those languages which are the union of an **NP** language and a **coNP** language (see [PY82, Pap94] for background on this class). **coDP** clearly contains both **NP** and **coNP**, and is contained in $\mathbf{\Sigma_2^p}$. Nisan's hardness proof can be viewed as a strengthening of Theorem 2, our coNP-hardness result for general ESSs. Like our proof of Theorem 2, his proof uses the Motzkin-Straus characterization of clique size in an essential way. A hard problem for **coDP** is to decide for a given graph G and given k, whether the maximum clique size of G is not exactly k (i.e., a "yes" instance is a graph G whose maximum clique size is not k). Nisan reduces this problem to the general ESS problem. Nisan also observes as a consequence of his proof that deciding whether a given mixed strategy is an ESS is **coNP**-complete. But this latter fact already follows from our results. Namely, the proof of Theorem 2 gives a reduction from coClique to ESS such that there is a clique of size $\geq k$ in the original graph iff the specific pure strategy a in the resulting game is not an ESS. Thus, it follows that checking whether a is an ESS in such games is **coNP**-hard. Moreover, a **coNP** upper bound for checking whether a given mixed strategy is an ESS follows from the proof of Theorem 6, our $\mathbf{\Sigma_2^p}$ upper bound for general ESSs. In that proof we observe that, once a given extended support set is guessed, a symmetric NE (s, s) with that extended support can be computed in P-time if it exists (and it must exist and be unique if s is an ESS), and we showed that one can then also check whether s is an ESS in **coNP**. Now, if we are given s to begin with, we can clearly first check in P-time whether (s, s) is an NE, and then check whether it is an ESS in **coNP**. Note also that the **coDP**-hardness result does not make redundant our **NP**-hardness result for ESSs, because our **NP**-hardness result also shows that deciding the existence of *regular* ESSs is already NP-hard. As we show in Theorem 4, unlike general ESSs, checking the existence of regular ESSs can be done in **NP** (and thus is not **coNP**-hard nor **coDP**-hard, under standard complexity assumptions). It also

[1] The results of our present paper were made available in 2004 as an ECCC tech report [EL04], and Nisan's note was made available in 2006 as an ECCC tech report [Nis06].

follows from the proof of Theorem 4 that checking whether a given strategy is a regular ESS can be done in **P**-time.

It remains an intriguing open problem to determine what complexity class captures the precise computational complexity of the general ESS problem.

References

[Aba80] Abakuks A (1980) Conditions for evolutionarily stable strategies. J Appl Prob 17:559–562

[BCSS98] Blum L, Cucker F, Shub M, and Smale S (1998) Complexity and Real Computation. Springer-Verlag.

[Bom86] Bomze I (1986) Non-cooperative 2-person games in biology: a classification. Int J Game Theory 15:31–59

[Bom92] Bomze I (1992) Detecting all evolutionarily stable strategies. J of Optimization Theory and Applications

[Bom02] Bomze I (2002) Regularity versus degeneracy in dynamics, games, and optimization: a unified approach to different aspects. SIAM Review 44(3):394–414

[BP89] Bomze I and Pötscher B M (1989) Game Theoretic Foundations of Evolutionary Stability. Springer

[CS03] Conitzer V and Sandholm T (2003) Complexity results about nash equilibria. In: 18th Int Joint Conf on Artificial Intelligence (IJCAI), pp. 765–771

[EL04] Etessami K and Lochbihler A (2004) The computational complexity of evolutionarily stable strategies. Electronic Colloquium on Computational Complexity (ECCC) technical report TR04-055

[GJ79] Garey M and Johnson D (1979) Computers and Intractability. W. H. Freeman and Company

[GZ89] Gilboa I and Zemel E (1989) Nash and correlated equilibria: some complexity considerations. Games and Economic Behavior 1:80–93

[Hai75] Haigh J (1975) Game theory and evolution. Adv Appl Prob 7:8–11

[HS94] Hammerstein P and Selten R (1994) Game theory and evolutionary biology (chapter 2). In Aumann R J, Hart S (eds.) Handbook of Game Theory, vol. 2, chap. 2, Elsevier, pp. 929–993

[Har73a] Harsanyi J (1973) Games with randomly distributed payoffs: a new rational for mixed strategy equilibrium points. Int J Game Theory 2:1–23

[Har73b] Harsanyi J (1973) Oddness of the number of equilibrium points: a new proof. Int J Game Theory 2:235–250

[HS98] Hofbauer J and Sigmund K (1998) Evolutionary Games and Population Dynamics. Cambridge University Press

[LT85] Lancaster P and Tismenetsky M (1985) The Theory of Matrices. Academic Press, 2nd ed.

[May82] Maynard Smith J (1982) Evolution and the Theory of Games. Cambridge University Press

[MWC+97] McNamara J M, Webb J N, Collins E J, Szekely T, and Houston A I (1997) A general technique for computing evolutionarily stable strategies based on dynamics. J Theoretical Biology 189:211–225

[MS65] Motzkin T S and Straus E G (1965) Maxima for graphs and a new proof of a theorem of Turan. Canadian Journal of Mathematics 17:533–540

[MK87] Murty K G and Kabadi S N Some NP-complete problems in quadratic and non-linear programming. Mathematical Programming, 39:117–129

[Nas51] Nash J (1951) Non-cooperative games. Annals of Mathematics

[Nis06] Nisan N (2006) A note on the computational hardness of evolutionarily stable strategies (DRAFT). Electronic Colloquium on Computational Complexity (ECCC) technical report TR06-076

[OR94] Osborne M J and Rubinstein A (1994) A Course in Game Theory. MIT press

[Pap94] Papadimitriou C (1994) Computational Complexity. Addison-Wesley

[Pap01] Papadimitriou C (2001) Algorithms, games and the internet. In: Proc of ACM Symp on Theory of Computing, pp. 249–253

[PY82] Papadimitriou C and Yannakakis M The complexity of facets (and some facets of complexity), 14th ACM Symp. on Theory of Computing, pp. 255-260

[Sel83] Selten R (1983) Evolutionary stability in extensive 2-person games. Mathematical Social Sciences 5:269–363

[SP73] Maynard Smith J and Price G R (1973) The logic of animal conflict. Nature, London, 246:15–18.

[Val79] Valiant L (1979) The complexity of computing the permanent. Theor Comput Sci 8:189–201

[vD91] van Damme E (1991) Stability and Perfection of Nash Equilibria. Springer-Verlag, 2nd ed.

[Vav90] Vavasis S A (1990) Quadratic programming is in NP. Inf Process Lett 36(2):73–77

[Wei97] Weibull J (1997) Evolutionary Game Theory. MIT Press