Approximating the Existential Theory of the Reals☆

Argyrios Deligkas[a], John Fearnley[b], Themistoklis Melissourgos[c,*], Paul G. Spirakis[b,d]

[a] Royal Holloway University of London, London, UK
[b] Department of Computer Science, University of Liverpool, Liverpool, UK
[c] Operations Research Group, Technical University of Munich, Munich, Germany
[d] Computer Engineering and Informatics Department, University of Patras, Patras, Greece

Abstract

The Existential Theory of the Reals (ETR) consists of existentially quantified Boolean formulas over equalities and inequalities of polynomial functions of real variables. In this paper we propose and study the approximate existential theory of the reals (ϵ-ETR) in which the constraints are only satisfied approximately. We first show that when the domain of the variables is the reals then ϵ-ETR $=$ ETR under polynomial time reductions, and then study the constrained ϵ-ETR problem where groups of variables are constrained to lie in bounded convex sets.

Our main result is a sampling theorem that discretizes the domain in a grid-like manner whose density depends on various properties of the ETR formula. A consequence of our theorem is that we obtain a (quasi-)polynomial time approximation scheme ((Q)PTAS) for a fragment of constrained ϵ-ETR. We use this theorem to create several new PTAS and QPTAS for problems from a variety of fields.

Keywords: Approximation schemes, Existential theory of the reals, Function problems

1. Introduction

1.1. Sampling techniques

The Lipton-Markakis-Mehta algorithm (LMM) is a well-known method for computing approximate Nash equilibria in normal form games [2]. The key idea behind their technique is to prove that there exist approximate Nash equilibria where all players use *simple* strategies.

Suppose that we have a convex set $C = \text{conv}(c_1, c_2, \dots, c_\ell)$ defined by vectors c_1 through c_ℓ. A vector $x \in C$ is *k-uniform* if it can be written as a sum of the form $(\beta_1/k)\cdot c_1 + (\beta_2/k)\cdot c_2 + \dots + (\beta_\ell/k)\cdot c_\ell$, where each β_i is a non-negative integer and $\sum_{i=1}^{\ell} \beta_i = k$. Since there are at most $\ell^{O(k)}$ k-uniform

---

☆A preliminary version of this paper appeared in the Proceedings of the 14th International Conference on Web and Internet Economics (WINE 2018) [1].

*Corresponding author

Email addresses: argyrios.deligkas@rhul.ac.uk (Argyrios Deligkas), john.fearnley@liverpool.ac.uk (John Fearnley), themistoklis.melissourgos@tum.de (Themistoklis Melissourgos), p.spirakis@liv.ac.uk (Paul G. Spirakis)

vectors, we can enumerate all k-uniform vectors in $\ell^{O(k)}$ time. For approximate equilibria in $n \times n$ bimatrix games, Lipton, Markakis, and Mehta showed that for every $\epsilon > 0$ there exists an ϵ-Nash equilibrium in which both players use k-uniform strategies where $k \in O(\log n/\epsilon^2)$, and so they obtained a quasi-polynomial time approximation scheme (QPTAS) for finding an ϵ-Nash equilibrium.

Their proof of this fact uses a sampling argument. Every bimatrix game has an exact Nash equilibrium (NE), and each player's strategy in this NE is a probability distribution. If we sample from each of these distributions k times, and consequently construct new k-uniform strategies using these samples, then for any $k \geq a \cdot \log n/\epsilon^2$, where a is a specified constant, there is positive probability the new strategies form an ϵ-NE. So by the probabilistic method, there must exist a k-uniform ϵ-NE. Finally, the aforementioned convex set containing each of the players' vectors (strategies) is now the unit $(n-1)$-simplex, and therefore it can be described as a convex hull of $\ell = n$ vectors. This means that there are $\binom{n+k-1}{k}^2$ pairs of k-uniform strategies, thus by exhaustively checking them in time $n^{O(k)}$ they find an ϵ-NE.

The sampling technique has been widely applied. It was initially used by Althöfer [3] in zero-sum games, before being applied to non-zero sum games by Lipton, Markakis, and Mehta [2]. Subsequently, it was used to produce algorithms for finding approximate equilibria in normal form games with many players [4], sparse bimatrix games [5], tree polymatrix [6], and Lipschitz games [7]. It has also been used to find constrained approximate equilibria in polymatrix games with bounded treewidth [8].

At their core, each of these results uses the sampling technique in the same way as the LMM algorithm: first take an exact solution to the problem, then sample from this solution k times, and finally prove that with positive probability the sampled vector is an approximate solution to the problem. The details of the proofs, and the value of k, are often tailored to the specific application, but the underlying technique is the same.

1.2. The existential theory of the reals

In this paper we ask the following question: *is there a broader class of problems to which the sampling technique can be applied?* We answer this by providing a sampling theorem for the existential theory of the reals. The existential theory of the reals consists of existentially quantified formulae using the connectives $\{\wedge, \vee, \neg\}$ over polynomials compared with the operators $\{<, \leq, =, \geq, >\}$. For example, each of the following is a formula in the existential theory of the reals.

$$\exists x \exists y \exists z \cdot (x = y) \wedge (x > z) \qquad \exists x \cdot (x^2 = 2)$$

$$\exists x \exists y \cdot \neg(x^{10} = y^{100}) \vee (y \geq 4) \qquad \exists x \exists y \exists z \cdot (x^2 + y^2 = z^2)$$

Given a formula in the existential theory of the reals, we must decide whether the formula is *true*, that is, whether there do indeed exist values for the variables that satisfy the formula. Throughout this paper we will use the Turing model of computation (also known as bit model). In this model, the inputs of our problems will be polynomial functions represented by tensors with rational entries which are encoded as a string of binary bits.

ETR is defined as the class that contains every problem that can be reduced in polynomial time to the typical ETR problem: Given a Boolean formula F, decide whether F is a true sentence in the existential theory of the reals. It is known that, in the Turing model, ETR $\subseteq$ PSPACE [9], and NP $\subseteq$ ETR since the problem can easily encode Boolean satisfiability. However, the class is not known to be equal to either PSPACE or NP, and it seems to be a distinct class of problems between the two. Many problems are now known to be ETR-complete, including various problems involving constrained equilibria in normal form games with at least three players [10, 11, 12, 13, 14].

1.3. Our contribution

In this paper we propose the *approximate* existential theory of the reals (ϵ-ETR), where we seek a solution that approximately satisfies the constraints of the formula. We show a subsampling theorem for a large fragment of ϵ-ETR, which can be used to obtain PTASs and QPTASs for the problems that lie within it. We believe that this will be useful for future research: instead of laboriously reproving subsampling results for specific games, it now suffices to simply write a formula in ϵ-ETR and then apply our theorem to immediately get the desired result. To exemplify this, we prove several new QPTAS and PTAS results using our theorem.

Our first result is actually that, in the computational complexity world, ϵ-ETR = ETR, meaning that the problem of computing an approximate solution to an ETR formula is as hard as finding an exact solution. However, this result crucially relies on the fact that ETR formulas can have solutions that are doubly-exponentially large. This motivates the study of *constrained* ϵ-ETR, where the solutions are required to lie within a given bounded convex set.

Our main theorem (Theorem 5) gives a subsampling result for constrained ϵ-ETR. It states that if the formula has an exact solution, then it also has a k-uniform approximate solution, where the value of k depends on various parameters of the formula, such as the number of constraints and the number of vector-variables. The theorem allows for the formula to be written using *tensor* constraints, which are a type of constraint that is useful in formulating game-theoretic problems.

The consequence of the main theorem is that, when various parameters of the formula are up to polylogarithmic in other specific parameters (see Corollary 1), we are able to obtain a QPTAS for approximating the existential theory of the reals. Specifically, this algorithm either finds an approximate solution of the constraints, or verifies that no exact solution exists. In many game theoretic and fair division applications an exact solution always exists, and so this algorithm will always find an approximate solution.

We should mention here also that our technique allows approximation of optimization problems whose objective function does not need to be described using the grammar of ETR formulas. For a discussion on this, see Remark 1. Also, we are not just applying the well-known subsampling techniques in order to derive our main theorem. The aforementioned theorem (Theorem 5) incorporates a new method for dealing with polynomials of degree d, which prior subsampling techniques were not able to deal with.

Theorem 5 can be applied to a wide variety of problems. In the game theoretic setting, we prove new results for constrained approximate equilibria in normal form games, and approximating the value vector of a Shapley game. Then we move to the fair division setting, and we show how a special case of the consensus halving problem admits a QPTAS. We also show optimization results. Specifically, we give approximation algorithms for optimizing polynomial functions over a bounded convex set, subject to polynomial constraints. We also give algorithms for approximating eigenvalues and eigenvectors of tensors. Finally, we apply our results to some problems from computational geometry.

2. The Existential Theory of the Reals

Let $x_1, x_2, \ldots, x_q \in \mathbb{R}$ be distinct *variables*, which we will treat as a vector $x \in \mathbb{R}^q$, called *vector-variable*. A *term* of a multivariate polynomial is a function $T(x) := a \cdot x_1^{d_1} \cdot x_2^{d_2} \cdot \cdots \cdot x_q^{d_q}$, where a is a non negative rational and $d_1, d_2, \ldots, d_q$ are non negative integers. A multivariate polynomial is a function $p(x) := T_1(x) + T_2(x) + \cdots + T_t(x) + c$, where each T_i is a term as defined above, and $c \in \mathbb{Q}_{\geq 0}$ is a constant.

We now define *Boolean formulae* over multivariate polynomials. The atoms of the formula are polynomials compared with $\{<, \leq, =, \geq, >\}$, and the formula itself can use the connectives $\{\wedge, \vee, \neg\}$.

Definition 1. The *existential theory of the reals* consists of every true sentence of the form $\exists x_1 \exists x_2 \ldots \exists x_q \cdot F(x)$, where F is a Boolean formula over multivariate polynomials of x_1 through x_q.

ETR is defined as the class that contains every problem that can be reduced in polynomial time to the typical ETR problem: Given a Boolean formula F, decide whether F is a true sentence in the existential theory of the reals. We will say that F has m constraints if it uses m operators from the set $\{<, \leq, =, \geq, >\}$ in its definition.

2.1. The approximate ETR

In the *approximate* existential theory of the reals, we replace the operators $\{<, \leq, \geq, >\}$ with their approximate counterparts. We define the operators $<_\epsilon$ and $>_\epsilon$ with the interpretation that $x <_\epsilon y$ holds if and only if $x < y + \epsilon$ and $x >_\epsilon y$ if and only if $x > y - \epsilon$ for some given $\epsilon > 0$. The operators $\leq_\epsilon$ and $\geq_\epsilon$ are defined analogously.

We do not allow equality tests in the approximate ETR. Instead, we require that every constraint of the form $x = y$ should be translated to $(x \leq y) \wedge (y \leq x)$ before being weakened to $(x \leq_\epsilon y) \wedge (y \leq_\epsilon x)$.

We also do not allow negation in Boolean formulas. Instead, we require that all negations are first pushed to atoms, using De Morgan's laws, and then further pushed into the atoms by changing the inequalities. So the formula $\neg((x \leq y) \wedge (a > b))$ would first be translated to $(x > y) \vee (a \leq b)$ before then being weakened to $(x >_\epsilon y) \vee (a \leq_\epsilon b)$.

Definition 2. The approximate existential theory of the reals consists of every true sentence of the form $\exists x_1 \exists x_2 \ldots \exists x_q \cdot F(x)$, where F is a negation-free Boolean formula using the operators $\{<_\epsilon, \leq_\epsilon, \geq_\epsilon, >_\epsilon\}$ over multivariate polynomials of x_1 through x_q.

Given a Boolean formula F, the ϵ-ETR problem asks us to decide whether F is a true sentence in the approximate existential theory of the reals, where the operators $\{<_\epsilon, \leq_\epsilon, \geq_\epsilon, >_\epsilon\}$ are used.

2.1.1. *Unconstrained* ϵ-ETR

Our first result is that if no constraints are placed on the value of the variables, that is if each x_i can be arbitrarily large, then ϵ-ETR = ETR for *all* values of $\epsilon > 0$. We show this via a two way polynomial time reduction between ϵ-ETR and ETR. The reduction from ϵ-ETR to ETR is trivial, since we can just rewrite each constraint $x <_\epsilon y$ as $x < y + \epsilon$, and likewise for the other operators.

For the other direction, we show that the ETR-complete problem FEASIBLE, which asks us to decide whether a system of multivariate polynomials $(p_i)_{i=1,\ldots,k}$ has a shared root, can be formulated in ϵ-ETR. We will prove this by modifying a technique of Schaefer and Stefankovic [15].

Definition 3 (FEASIBLE). Given a system of k multi-variate polynomials $p_i : \mathbb{R}^n \to \mathbb{R}^n$, $i = 1, \ldots, k$, decide whether there exists an $x \in \mathbb{R}^n$ such that $p_i(x) = 0$ for all i.

Schaefer and Stefankovic showed that this problem is ETR-complete.

Theorem 1 ([15])**.** FEASIBLE *is* ETR*-complete.*

We will reduce FEASIBLE to ϵ-ETR. Let $P = (p_i)_{i=1,\ldots,k}$ be an instance of FEASIBLE, and let L be the number of bits needed to represent this instance. We define $\text{gap}(P) = 2^{-2^{L+5}}$. The following lemma was shown by Schaefer and Stefankovic.

Lemma 2 ([15])**.** *Let $P = (p_i)_{i=1,\ldots,k}$ be an instance of* FEASIBLE*. If there does not exist an $x \in \mathbb{R}^n$ such that $p_i(x) = 0$ for all i, then for every $x \in \mathbb{R}^n$ there exists an i such that $|p_i(x)| > \text{gap}(P)$.*

In other words, if the instance of FEASIBLE is not solvable, then at any given point, some polynomial will be bounded away from 0 by at least $\text{gap}(P)$.

The reduction.. The first task is to build an ϵ-ETR formula that ensures that a variable $t \in \mathbb{R}$ satisfies $t \geq \epsilon / \text{gap}(P)$. This can be done by the standard trick of repeated squaring, but we must ensure that the ϵ-inequalities do not interfere with the process. We define the following formula over the variables $t, g_1, g_2, \ldots, g_{L+6} \in \mathbb{R}^n$, where all of the following constraints are required to hold.

$$
\begin{aligned}
g_1 &\geq_\epsilon 2 + \epsilon, \\
g_j &\geq_\epsilon g_{j-1}^2 + \epsilon, && \text{for all } j \in \{2, 3, \ldots, L+6\}. \\
t &\geq_\epsilon \epsilon \cdot g_{L+6} + \epsilon.
\end{aligned}
$$

In other words, this requires that $g_1 \geq 2$, and $g_j \geq g_{j-1}^2$. So we have $g_{L+6} \geq 2^{2^{L+5}}$, and hence $t \geq \epsilon / \operatorname{gap}(P)$. Note that the size of this formula is polynomial in L, i.e., the size of instance P.

Given an instance $P = (p_i)_{i=1,\dots,k}$ of FEASIBLE we create the following ϵ-ETR instance ψ, where all of the following are required to hold.

$$t \cdot p_i(x) \leq_\epsilon 0 \qquad \text{for all } i, \tag{1}$$

$$t \cdot p_i(x) \geq_\epsilon 0 \qquad \text{for all } i, \tag{2}$$

$$t \geq_\epsilon \epsilon / \operatorname{gap}(P) + \epsilon, \tag{3}$$

where the final inequality is implemented using the construction given above.

Lemma 3. *ψ is satisfiable if and only if P has a solution.*

PROOF. First, let us assume that P has a solution. This means that there exists an $x \in \mathbb{R}^n$ such that $p_i(x) = 0$ for all i. Note that x clearly satisfies inequalities (1) and (2), while inequality (3) can be satisfied by fixing t to be any number greater than $\epsilon / \operatorname{gap}(P)$. So we have proved that ψ is satisfiable.

For the other direction of the equivalence, now we will assume that $x \in \mathbb{R}^n$ satisfies ψ. Note that we must have

$$p_i(x) \leq \epsilon / t \leq \operatorname{gap}(P)$$

and likewise

$$p_i(x) \geq -\epsilon / t \geq -\operatorname{gap}(P),$$

and hence $|p_i(x)| \leq \operatorname{gap}(P)$ for all i. But Lemma 2 states that this is only possible in the case where P has a solution.

This completes the proof of the following theorem.

Theorem 4. *ϵ-ETR $=$ ETR for all $\epsilon \geq 0$.*

2.1.2. Constrained ϵ-ETR

In our negative result for unconstrained ϵ-ETR, we abused the fact that variables could be arbitrarily large to construct the doubly-exponentially large number t. So, it makes sense to ask whether ϵ-ETR gets easier if we *constrain* the problem so that variables cannot be arbitrarily large.

In this paper, we consider ϵ-ETR problems that are constrained by a Cartesian product of bounded convex sets, each being a subset of $\mathbb{R}^q$. For a fixed $i \in [n]$, and some given vectors $c_1^i, c_2^i, \ldots, c_\ell^i \in \mathbb{R}^q$, we use $\text{conv}(c_1^i, c_2^i, \ldots, c_\ell^i) := C_i$ to denote the set containing every vector that lies in the convex hull of c_1^i through c_ℓ^i. In the *constrained* ϵ-ETR, we require that in a solution $x := (x_1, \ldots, x_n)$ of the ϵ-ETR problem (with n vector-variables), we have $x_i \in C_i$ for every $i \in [n]$. In other words, the solution x lies in the Cartesian product of individual vector-variables' domains, that is, $\bigtimes_{i=1}^n C_i$.

Definition 4. Given vectors $c_1, c_2, \ldots, c_\ell \in \mathbb{R}^q$ and a Boolean formula F that uses the operators $\{<_\epsilon, \leq_\epsilon, \geq_\epsilon, >_\epsilon\}$, the constrained ϵ-ETR problem asks us to decide whether

$$\exists x_1 \exists x_2 \ldots \exists x_q \cdot \big(x \in \text{conv}(c_1, c_2, \ldots, c_\ell) \wedge F(x)\big).$$

Note that, unlike the constraints used in F, the convex hull constraints are not weakened. So the resulting solution $x_1, x_2, \ldots, x_q$, must actually lie in the convex hull.

3. Approximating Constrained ϵ-ETR

3.1. Polynomial classes

To state our main theorem, we will use a certain class of polynomials where the coefficients are given as a tensor. This will be particularly useful when we apply our theorem to certain problems, such as normal form games. To be clear though, this is not a further restriction on the constrained ϵ-ETR problem, since all polynomials can be written down in this form.

As mentioned earlier, we use the term *vector-variable* to refer to a p-dimensional vector; for example, in Definition 4, the q-dimensional vector x would be called vector-variable under this terminology. The variables of the polynomials we study in this paper will be grouped, without loss of generality, into p-dimensional vector-variables denoted as $x_1, x_2, \ldots, x_n$, where $x_j(i)$ will denote the i-th element ($i \in [p]$) of vector x_j, and is called *variable*. The coefficients of the polynomials will be captured by a tensor denoted by A. Given a $\times_{j=1}^n p$ tensor A, we denote by $a(i_1, \ldots, i_n)$ its element with coordinates $(i_1, \ldots, i_n)$ on the tensor's dimensions $1, \ldots, n$, respectively, and by α we denote the maximum absolute value of these elements. We define the following two classes of polynomials.

- **Simple tensor multivariate.**

We will use $\mathrm{STM}(A, x_1^{d_1}, \dots, x_n^{d_n})$ to denote an STM polynomial with n vector-variables where each vector-variable x_j, $j \in [n]$ is applied d_j times on tensor A that defines the coefficients. Tensor A has $\sum_{j=1}^{n} d_j$ dimensions with p indices each. We will say that an STM polynomial is of maximum degree d, if $d = \max_j d_j$. Note that in an STM polynomial, in each of its terms a variable from each of all vector-variables appears. Here is an example of a degree 2 simple tensor multivariate polynomial with two vector-variables:

$$\mathrm{STM}(A, x^2, y) = \sum_{i=1}^{p} \sum_{j=1}^{p} \sum_{k=1}^{p} x(i) \cdot x(j) \cdot y(k) \cdot a(i, j, k).$$

This polynomial itself is written as follows.

$$\begin{aligned}&\mathrm{STM}(A, x_1^{d_1}, \dots, x_n^{d_n}) = \\ &\sum_{i_{1,1} \in [p]} \cdots \sum_{i_{n,d_n} \in [p]} (x_1(i_{1,1})) \cdot \dots \cdot (x_1(i_{1,d_1})) \cdot \dots \cdot (x_n(i_{n,1})) \cdot \dots \cdot (x_n(i_{n,d_n})) \cdot \\ &\qquad \cdot a(i_{1,1}, \dots, i_{1,d_1} \dots, i_{n,1}, \dots, i_{n,d_n}).\end{aligned}$$

- **Tensor multivariate.** A tensor multivariate (TMV) polynomial is the sum over a number of simple tensor multivariate polynomials. We will use $\mathrm{TMV}(x_1, \dots, x_n)$ to denote a tensor multivariate polynomial with n vector-variables, which is formally defined as

$$\mathrm{TMV}(x_1, \dots, x_n) = \sum_{i \in [t]} \mathrm{STM}(A_i, x_1^{d_{i1}}, \dots, x_n^{d_{in}}),$$

where the exponents $d_{i1}, \dots, d_{in}$ depend on i, and t is the number of simple multivariate polynomials. We will say that $\mathrm{TMV}(x_1, \dots, x_n)$ has length t if it is the sum of t STM polynomials, and that it is of degree d if $d = \max_{i \in [t], j \in [n]} d_{ij}$. Observe that $t \leq (d+1)^n$; it could be the case that a TMV polynomial is a sum of STM polynomials, each of which has a distinct combination of exponents $d_{i1}, \dots, d_{in}$ in its vector-variables, where $d_{ij} \in \{0, 1, \dots, d\}$.

3.2. ϵ-ETR with tensor constraints

We focus on ϵ-ETR instances F where all constraints are of the form $\mathrm{TMV}(x_1, \dots, x_n) \bowtie 0$, where $\bowtie$ is an operator from the set $\{<_\epsilon, \leq_\epsilon, >_\epsilon, \geq_\epsilon\}$. Recall that each TMV constraint considers vector-variables. We consider the number of vector-variables used in F (denoted as n) to be the number of vector-variables used in the TMV constraints. So the value of n used in our main theorem may be constant in the case that a constant number of vector-variables are used, even if the underlying ϵ-ETR instance actually has a non-constant number of variables. For example, if x, y and w are p-dimensional probability distributions and A_1 and A_2 are $p \times p$ tensors, the TMV constraint $x^T A_1 y + w^T A_2 x > 0$ has three vector-variables, degree 1, length two; though the underlying problem has $3 \cdot p$ variables.

Note that every ϵ-ETR constraint can be written as a TMV constraint, because all multivariate polynomials can be written down as a TMV polynomial. Every term of a TMV can be written as a STM polynomial where the tensor entry is non zero for exactly the combination of variables used in the term, and 0 otherwise. Then a TMV polynomial can be constructed by summing over the STM polynomials for each individual term.

3.2.1. The main theorem

Given an ϵ-ETR formula F, we define $\text{exact}(F)$ to be a Boolean formula in which every approximate constraint is replaced by its exact variant, meaning that every instance of $x \leq_\epsilon y$ is replaced with $x \leq y$, and likewise for the other operators. We also call by *k-uniform solution* a solution whose each vector-variable is a k-uniform vector.

Our main theorem is as follows.

Theorem 5. *Let F be an ϵ-ETR instance with n vector-variables and m multivariate-polynomial constraints each one of length at most t and maximum degree d. Let each vector-variable x_i be constrained in the convex hull C_i defined by ℓ vectors $c_1^i, c_2^i, \ldots, c_\ell^i \in \mathbb{R}^p$. Let α be the maximum absolute value of the coefficients of constraints of F, and let $\gamma = \max_{i\in[n]} \max_{j\in[\ell]} \|c_j^i\|_\infty$. If $\text{exact}(F)$ has a solution in $\bigtimes_{i=1}^n C_i$, then F has a k-uniform solution in $\bigtimes_{i=1}^n C_i$ where*

$$k = \frac{512 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot d^6 \cdot n^6 \cdot t^5 \cdot \ln(2 \cdot \alpha' \cdot \gamma' \cdot d \cdot n \cdot t \cdot m)}{\epsilon^5},$$

where $\alpha' := \max(\alpha, 1), \gamma' := \max(\gamma, 1)$.

3.2.2. Consequences of the main theorem

Our main theorem gives a QPTAS for approximating a fragment of ϵ-ETR. The total number of k-uniform vectors in a convex set $C = \text{conv}(c_1, c_2, \ldots, c_\ell)$ is $\binom{\ell+k-1}{k}$ which is $\min\{\ell^{O(k)}, k^{O(\ell)}\}$. In most of the applications (see Section 5), we have $\ell >> k$, that is why for ease of presentation we will assume in the sequel that $\binom{\ell+k-1}{k} \in \ell^{O(k)}$. So, if the parameters α, γ, d, t, and n are all polylogarithmic in m, then our main theorem tells us that the total number of k-uniform vectors for each vector-variable is $\ell^{O(\text{poly}\log m)}$, where m is the number of constraints. Therefore, if we consider all k-uniform vectors for each of the n vector-variables, we can check whether F holds for each individual n-tuple of k-uniform vectors, and if it does, we can output it as a solution. If no such n-tuple exists that satisfies F, then we can conclude that $\text{exact}(F)$ has no solution. This gives us the following result.

Corollary 1. *Let F be an ϵ-*`ETR` *instance constrained by the convex hull defined by $c_1, c_2, \ldots, c_\ell$. If α, γ, n, d, and t are polylogarithmic in m, then we have an algorithm that runs in time $\ell^{O\left(\frac{poly \log m}{\epsilon^5}\right)}$ and either finds a solution to F, or determines that $\text{exact}(F)$ has no solution.*

Let N be the input size of the given problem. If m is constant and ℓ is polynomial in N then this gives a PTAS, while if m and ℓ are polynomial in N, then this gives a QPTAS.

In Section 5 we will show that the problem of approximating the best social welfare achievable by an approximate Nash equilibrium in a two-player normal form game can be written down as a constrained ϵ-`ETR` formula where α, γ, d, and n are constant (and recall that $t \le (d+1)^n$). It has been shown that, assuming the exponential time hypothesis, this problem cannot be solved faster than quasi-polynomial time [16, 17], so this also implies that constrained ϵ-`ETR` where α, γ, d, and n are constant cannot be solved faster than quasi-polynomial time unless the exponential time hypothesis is false.

Many ϵ-`ETR` problems are naturally constrained by sets that are defined by the convex hull of exponentially many vectors. The cube $[0, 1]^p$ is a natural example of one such set. Brute force enumeration does not give an efficient algorithm for these problems, since we need to enumerate $\ell^{O(k)}$ vectors, and ℓ is already exponential in the dimension parameter p. However, our main theorem is able to provide non-deterministic polynomial time algorithms for these problems.

This is because each k-uniform vector is, by definition, the convex combination of at most k of the vectors in the convex set, and this holds even if ℓ is exponential. So, provided that k is polynomial in the input size, we can guess the subset of vectors that are used, and then verify efficiently that the formula holds. This is particularly useful for problems where $\text{exact}(F)$ always has a solution, which is often the case in game theory applications, since it places the approximation problem in `NP`, whereas deciding the existence of an exact solution may be `ETR`-complete.

Corollary 2. *Let F be an ϵ-*`ETR` *instance constrained in $\bigtimes_{i=1}^{n} C_i$, where $C_i = \mathrm{conv}(c_1^i, c_2^i, \ldots, c_\ell^i)$. If α, γ, d, t, n, are polynomial in the input size, then there is a non-deterministic polynomial time algorithm that either finds a solution to F, or determines that* $\mathrm{exact}(F)$ *has no solution. Moreover, if* $\mathrm{exact}(F)$ *is guaranteed to have a solution, then the problem of finding an approximate solution for F is in* `NP`.

3.2.3. Approximation notions

According to the relaxation procedure for `ETR` that we have described, each atom A_i of the `ETR` formula is relaxed additively by a positive quantity ϵ. The main theorem (Theorem 5) and the intermediate results, give a sufficiently fine discretization (distance at most $1/k$ for some $k \in \mathbb{N}^*$) of the domain of the `ETR` instance's variables, such that if there exists an exact solution $x^* = (x_1^*, \ldots, x_n^*)$ of the formula then there exists a k-uniform solution in the discretized domain that ϵ-satisfies every A_i. In particular we prove that if $A_i = (p(x) \bowtie 0)$, where $p(x)$ is a multivariate polynomial and $\bowtie \in \{<, \leq, =, \geq, >\}$, then there exists a k-uniform vector x' such that $|p(x') - p(x^*)| \leq \epsilon$. This implies the ϵ-satisfaction of each A_i by the triangle inequality.

In fact, by this work we do not aim to output an "approximate yes/no" to an `ETR` instance, i.e. to give a yes/no answer to the relaxed `ETR` instance, but instead to output an *approximate solution* (if an exact solution exists) to the `ETR` instance. Therefore, more accurately we should refer to this approximation of `ETR` as an approximation of `Function ETR` (`FETR`), where `FETR` is the function problem extension of the decision problem complexity class `ETR`. As `ETR` is the analogue of `NP`, `FETR` is the analogue of `FNP` in the Blum-Shub-Smale computation model [18].

Definition 5 (ϵ-approximation). Consider a given `ETR` instance with domain D and formula F. If x^* is a solution to the instance and x' is a solution to the respective ϵ-ETR instance for a given $\epsilon > 0$, then x' is called an *ϵ-approximation* of x^*.

Definition 6 (PTAS/QPTAS). Consider a function problem P with input size N, whose objective is to output a solution x^*. An algorithm that computes an ϵ-approximation x' of P in time polynomial in N for any fixed $\epsilon > 0$ is a Polynomial Time Approximation Scheme (PTAS). An algorithm that computes x' in time $O(N^{\mathrm{poly}\log N})$ is a Quasi-Polynomial Time Approximation Scheme (QPTAS).

Remark 1. Our technique that finds an x' such that $|p(x') - p(x^*)| \leq \epsilon$ provides one with more power than showing that polynomial inequalities weakened by ϵ hold for x'. In fact, it allows for approximation of solutions that need not be described by an ETR formula. A simple example of such a case is the one presented in Section 4.1 where we seek an approximation of the maximum of the quadratic function in the simplex. The maximization objective does not need to be written in an ETR formula. Instead, we show that any point $f(x)$ of the quadratic function, for x in the simplex, can be approximated by a point $f(x')$ where x' is in a discrete simplex with a small number of points. Then we find the maximum of $f(x')'s$ which is smaller than $\max(f(x))$ by at most ϵ.

The fact that operation "max" can be executed in time linear in the number of points of the discretized simplex allows us to use our method for expressions with "max" which is forbidden in the grammar of ETR. More generally, the following theorem shows that even more complicated objectives, such as "$\max_{x_1} \min_{x_2}$" can be treated by a modification of the algorithm described in Section 3.2.2.

Theorem 6. *Let F be a multi-objective optimization instance whose objective functions are multivariate polynomials, with n vector-variables constrained in $\bigtimes_{i=1}^{n} C_i$, where $C_i = \mathrm{conv}(c_1^i, c_2^i, \ldots, c_\ell^i)$. Let k be the quantity specified in Theorem 5 with m being the number of polynomial functions in the instance, meaning the ones in the objectives and constraints. If every objective on the functions has a polynomial time algorithm to be performed on a discrete domain, then there is an algorithm that runs in time $\min\{\ell^{O(k \cdot n)}, k^{O(\ell \cdot n)}\}$, and either finds a solution which satisfies every objective of F within additive ϵ, or determines that F has no solution.*

Proof. As explained at the beginning of this section, our technique discretizes the domain of the variables with a density sufficient to approximate any point of any of the polynomial functions that are given as part of the atoms of an ETR formula. That is, for any x^* in the continuous domain it guarantees the existence of a discrete x' such that for every polynomial p in the atoms, it is $|p(x') - p(x^*)| \leq \epsilon$. Note now that the technique works for any given set of polynomials when we require that for every polynomial in the set, every point x^* has a discrete x'. This is regardless of what the atoms' operators from $\{<, \leq, =, \geq, >\}$ are or with what logical operators from $\{\wedge, \vee\}$ the atoms connect to each other.

In view of the above, observe that any objective (with the properties of the statement of the theorem) on functions, takes time polynomial in the size of the discretized space, therefore it does not change asymptotically the total running time of the algorithm described at the beginning of Section

3.2.2. That is because first, the aforementioned algorithm will brute-force through all of the points in the discretized domain and for these points it will check if all of the constraints of F are satisfied. Now the algorithm we propose will deviate from the aforementioned algorithm and for the points that satisfy the constraints of F (feasible points), for each objective it will run the efficient respective algorithm of the objective on the feasible points and check whether all objectives of the relaxed by ϵ instance are satisfied for some point. This can be done in time polynomial in the size of the discretized domain, i.e. $\binom{\ell+k-1}{k}^n$ (the exponent n comes from the fact that the algorithm will check all combinations of n many k-uniform vectors.). If a discrete point is found that ϵ-satisfies F, then the algorithm returns it, otherwise there is no point in the continuous domain that satisfies F according to Theorem 5.

3.3. A theorem for non-tensor constraints

One downside of Theorem 5 is that it requires that the formula is written down using tensor constraints. We have argued that every ETR formula can be written down in this way, but the translation introduces a new vector-variable for each group of variables that are constrained in a bounded convex set in the ETR formula. When we apply Theorem 5 to obtain PTASs or QPTASs we require that the number of vector-variables is at most polylogarithmic, and so this limits the application of the theorem to ETR formulas that have at most polylogarithmically many groups of variables that are under the same bounded domain.

Theorem 9 is a sampling result for ϵ-ETR with non-tensor constraints, which is proved via some intermediate results. First, we will use the following theorem of Barman.

Theorem 7 ([5]). *Let $c_1, c_2, \ldots, c_\ell \in \mathbb{R}^q$ with $\max_i \|c_i\|_\infty \leq 1$. For every $x \in conv(c_1, c_2, \ldots, c_\ell)$ and every $\epsilon > 0$ there exists an $O(\log \ell/\epsilon^2)$-uniform vector $x' \in \mathrm{conv}(c_1, c_2, \ldots, c_\ell)$ such that $\|x - x'\|_\infty \leq \epsilon$.*

The following lemma shows that if we take two vectors x and x' that are close in the L_∞ norm, then for all polynomials p the value of $|p(x) - p(x')|$ cannot be too large.

We denote by $consts(p)$ the maximum absolute coefficient in polynomial p, and by $terms(p)$ the number of terms of p.

Lemma 8. *Let $p(x)$ be a multivariate polynomial over $x \in \mathbb{R}^q$ with degree d and let $\epsilon \in (0, \gamma]$ for some constant $\gamma > 0$. For every pair of vectors $x, x' \in [0, \gamma]^q$ with $\|x - x'\|_\infty \leq \epsilon$ we have:*

$$|p(x) - p(x')| \leq \gamma^{d-1} \cdot (2^d - 1) \cdot consts(p) \cdot terms(p) \cdot \epsilon.$$

Proof. Consider a term of $p(x)$, which can without loss of generality be written as $t(x) = c \cdot \prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} x_i^{d_i}$, where d_i is the degree of coordinate x_i (resp. x_i'). We have

$$
\begin{aligned}
|t(x) - t(x')| &= \left| c \cdot \prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} x_i^{d_i} - c \cdot \prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} (x_i')^{d_i} \right| \\
&= c \cdot \left| \prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} x_i^{d_i} - \prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} (x_i')^{d_i} \right| \\
&\le c \cdot \left[\prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} (x_i^{d_i} + \epsilon) - \prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} x_i^{d_i} \right] \\
&\le c \cdot \left[\left[\prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} x_i^{d_i} + \binom{d}{1} \gamma^{d-1} \epsilon + \binom{d}{2} \gamma^{d-2} \epsilon^2 + \cdots + \binom{d}{d} \gamma^0 \epsilon^d \right] - \prod_{\substack{i \in [q] \\ \sum_i d_i \le d}} x_i^{d_i} \right] \\
&\le c \cdot \epsilon \cdot \sum_{k=1}^{d} \binom{d}{k} \gamma^{d-1} \\
&= c \cdot \epsilon \cdot \gamma^{d-1} \cdot \sum_{k=1}^{d} \binom{d}{k} \\
&= \epsilon \cdot c \cdot \gamma^{d-1} \cdot (2^d - 1),
\end{aligned}
$$

where the fourth and third to last lines use the fact that x_i's, and ϵ are all at most γ.

Next consider a term $t(x)$ of $p(x)$ of degree $d' \le d$. This can be written similarly to the aforementioned term. Then $|t(x) - t(x')| \le c \cdot \epsilon \cdot \gamma^{d-1} \cdot (2^{d'} - 1) \le c \cdot \epsilon \cdot \gamma^{d-1} \cdot (2^d - 1)$. Since there are $terms(p)$ many terms in p, we have

$$|p(x) - p(x')| \le \gamma^{d-1} \cdot (2^d - 1) \cdot consts(p) \cdot terms(p) \cdot \epsilon.$$

We now apply this to prove the following theorem.

Theorem 9. *Let F be an ϵ-ETR instance with n vector-variables, where the i-th vector variable is constrained over the convex hull $C_i = conv(c_1^i, c_2^i, \ldots, c_\ell^i) \subset \mathbb{R}^q$. Let $\gamma = \max_i \|c_i\|_\infty$, let α be the largest constant coefficient used in F, let r be the number of terms used in total in all polynomials of F, and let d be the maximum degree of the polynomials in F. If $\text{exact}(F)$ has a solution in $\bigtimes_{i=1}^n C_i$, then F has a k-uniform solution in $\bigtimes_{i=1}^n C_i$ where*

$$k = \alpha^2 \cdot \gamma^{2d-2} \cdot (2^d - 1)^2 \cdot r^2 \cdot \log \ell / \epsilon^2.$$

Proof. Let x be the solution to $\text{exact}(F)$. First we apply Theorem 7 to find a point y that is k-uniform, where $k = \alpha^2 \cdot \gamma^{2d-2} \cdot (2^d - 1)^2 \cdot r^2 \cdot \log \ell / \epsilon^2$, such that

$$\|x - y\|_\infty \leq \epsilon / (\alpha \cdot \gamma^{d-1} \cdot (2^d - 1) \cdot r).$$

Next we can apply Lemma 8 to argue that, for each polynomial p used in F, we have

$$\begin{aligned} |p(x) - p(y)| &\leq \alpha \cdot \gamma^{d-1} \cdot (2^d - 1) \cdot r \cdot \left(\frac{\epsilon}{\alpha \cdot \gamma^{d-1} \cdot (2^d - 1) \cdot r} \right) \\ &= \epsilon. \end{aligned}$$

Since all constraints of F have a tolerance of ϵ, and since x satisfies $\text{exact}(F)$, we can conclude that $F(y)$ is satisfied.

The key feature here is that the number of variables, and most importantly, the number of vector-variables (n), does not appear in the formula for k, which allows the theorem to be applied to some formulas for which Theorem 5 cannot. However, since the theorem does not allow tensor constraints, its applicability is more limited because the number of terms r will be much larger in non-tensor formulas. For example, as we will see in Section 5, we can formulate bimatrix games using tensor constraints over constantly many vector-variables, and this gives a positive result using Theorem 5. No such result can be obtained via Theorem 9, because when we formulate the problem without tensor constraints, the number of terms r used in the inequalities becomes polynomial in the dimension.

4. The Proof of the Main Theorem

In this section we prove Theorem 5. Before we proceed with the technical results, let us illustrate via an example the crucial idea for proving that the special vectors we have defined (i.e. the k-uniform vectors for some $k \in \mathbb{N}^*$) inside a discretized convex hull can be used to approximate not only multilinear polynomials, but also multivariate polynomials of degree $d \geq 2$. At the same time, we show that the discretization of the domain (points in distance at most $1/k$ from each other) does not need to be very fine in order to achieve an additive approximation ϵ at any point of such a function. Our example is in approximating the quadratic polynomial over the simplex.

Let us provide a roadmap for this section. We begin by the detailed aforementioned example. Then we proceed by considering two special cases, namely Lemma 12 and Lemma 14, which when combined will be the backbone of the proof of the main theorem.

Firstly, we will show how to deal with problems where every constraint of the Boolean formula is a *multilinear polynomial*, which we will define formally later. We deal with this kind of problems using Hoeffding's inequality and the union bound, which is similar to how such constraints have been handled in prior work.

Then, we study problems where the Boolean formula consists of a *single* degree d polynomial constraint. We reduce this kind of problems to a constrained $\epsilon/2$-ETR problem with multilinear constraints, so we can use our previous result to handle the reduced problem. Sampling techniques in degree d polynomial problems have not been considered in previous work, and so this reduction is a novel extension of sampling-based techniques to a broader class of ϵ-ETR formulas.

Finally, we deal with the main theorem: we reduce the original ETR problem with multivariate constraints to a set of ϵ'-ETR problems with a single standard degree d constraint, and then we use the last result to derive a bound on k.

As a byproduct of our main result one can get the same result as that of [19] in which a PTAS for fixed degree polynomial minimization over the simplex was presented. Even though the PTAS that follows from our result on the same optimization problem has roughly the same running time as that of [19], the proof presented here (which is independent of the aforementioned work) is significantly simpler. Nevertheless, the result in the current work generalizes previous results on polynomial optimization over the simplex, by providing a universal algorithm for multi-objective optimization problems, and showing how its running time depends on the parameters of the problem (see Theorem 6).

4.1. Example: A simple PTAS for quadratic polynomial optimization over the simplex

Definition 7 (Standard quadratic optimization problem (SQP)). Given a $p \times p$ matrix A with entries normalized in $[0, 1]$, find the value

$$v^* := \max_{x \in \Delta_p} x^T A x, \quad \text{where } \Delta_p \text{ is the } (p-1) - \textit{simplex}.$$

SQP is a strongly NP-hard problem, even for the case where A has entries in $\{0, 1\}$; in a theorem of Motzkin and Straus [20] it is shown that if matrix A is the adjacency matrix of a graph on p vertices whose maximum clique has c vertices, then $v^* = 1 - 1/c$. The problem of finding the size of the maximum clique in a general graph is known to be (strongly) NP-hard since its decision version is one of Karp's 21 NP-complete problems [21]. Therefore, unless P = NP there is no Fully Polynomial Time Approximation Scheme for SQP and the best thing we can hope for the problem is a PTAS. We present a PTAS for SQP (Corollary 3), which has almost the same running time as that of [22], but we claim that our proof is significantly simpler.

Let $x^* \in \arg(v^*)$. Consider the set $\Delta_p(k)$ of all k-uniform vectors, for $k = 16 \ln(3/\epsilon)/\epsilon^2$, with items $x^{(i)} \in \Delta_p(k)$, for $i = 1, 2, \ldots, |\Delta_p(k)|$.

Lemma 10. *There exists a multiset $\mathcal{X}$ of $\Delta_p(k)$ with $|\mathcal{X}| = 2/\epsilon$ such that for every $x^{(i)}, x^{(j)} \in \mathcal{X}$ with $i \neq j$, it is*

$$x^{*T} A x^* - x^{(i)T} A x^{(j)} < \epsilon/2.$$

Proof. Note that although $i \neq j$, $x^{(i)}$ could be equal to $x^{(j)}$ since the two k-uniform vectors belong to a multiset of $\Delta_p(k)$. The proof is by the probabilistic method. Let us create the events

$$\begin{aligned}
E_i &= \left\{ x^{*T} A x^* - x^{(i)T} A x^* < \epsilon/4 \right\}, && \forall i \text{ for which } x^{(i)} \in \mathcal{X}, \\
F_{i,j} &= \left\{ x^{(i)T} A x^* - x^{(i)T} A x^{(j)} < \epsilon/4 \right\}, && \forall i, j \text{ with } i \neq j, \text{ for which } x^{(i)}, x^{(j)} \in \mathcal{X}, \\
G_{i,j} &= \left\{ x^{*T} A x^* - x^{(i)T} A x^{(j)} < \epsilon/2 \right\}, && \forall i, j \text{ with } i \neq j, \text{ for which } x^{(i)}, x^{(j)} \in \mathcal{X}.
\end{aligned}$$

Observe that $E_i \cap F_{i,j} \subseteq G_{i,j}$. Now, let each of k i.i.d. random variables be drawn from x^*. The sample space for each is $[p]$. For any $x^{(i)}, x^{(j)} \in \Delta_p(k)$, the expectation of $x^{(i)T} A x^*$ is $x^{*T} A x^*$, and the expectation of $x^{(i)T} A x^{(j)}$ (for fixed $x^{(i)}$) is $x^{(i)T} A x^*$. Let us denote $r := |\mathcal{X}| = 2/\epsilon$. By using a

Höffding bound [23], we get

$$\Pr\{\overline{E}_i\} \le e^{-k\epsilon^2/8}, \quad \forall i \text{ for which } x^{(i)} \in \mathcal{X}, \text{ and}$$
$$\Pr\{\overline{F}_{i,j}\} \le e^{-k\epsilon^2/8}, \quad \forall i,j \text{ with } i \neq j, \text{ for which } x^{(i)}, x^{(j)} \in \mathcal{X}.$$

Consider now the event H that captures the condition that needs to be satisfied by the lemma. It is

$$H = \bigcap_{\substack{i,j\in\mathcal{X}\\ i\neq j}} G_{i,j}.$$

Therefore

$$\overline{H} = \bigcup_{\substack{i,j\in\mathcal{X}\\ i\neq j}} \overline{G}_{i,j} \subseteq \bigcup_{i\in\mathcal{X}} \overline{E}_i \bigcup_{\substack{i,j\in\mathcal{X}\\ i\neq j}} \overline{F}_{i,j}.$$

Hence

$$\begin{aligned}\Pr\{\overline{H}\} &\le re^{-k\epsilon^2/8} + r(r-1)e^{-k\epsilon^2/8}\\ &= r^2 e^{-k\epsilon^2/8}\\ &< 1.\end{aligned}$$

The above strict inequality means that $\Pr\{H\} > 0$, therefore, there exists a set $\mathcal{X}$ that satisfies the statement of the lemma.

The following theorem corresponds to the general Lemma 14, for the case $\alpha = \gamma = 1$, $d = 2$.

Theorem 11. *There exists a $\frac{32\ln(3/\epsilon)}{\epsilon^3}$-uniform vector x, such that $v^* - x^T Ax < \epsilon$.*

Proof. Consider the multiset $\mathcal{X}$ of $\Delta_p(k)$ of Lemma 10, and recall that $r := |\mathcal{X}| = 2/\epsilon$. Let us create the vector

$$x := \frac{1}{r}\sum_{i\in\mathcal{X}} x^{(i)}.$$

Then, it is

$$
\begin{aligned}
x^{*T}Ax^* - x^TAx &= x^{*T}Ax^* - \left(\frac{1}{r}\sum_{x^{(i)}\in\mathcal{X}} x^{(i)T}\right) A \left(\frac{1}{r}\sum_{x^{(i)}\in\mathcal{X}} x^{(i)}\right) \\
&= x^{*T}Ax^* - \frac{1}{r^2}\sum_{x^{(i)},x^{(j)}\in\mathcal{X}} x^{(i)T}Ax^{(j)} \\
&= x^{*T}Ax^* - \frac{1}{r^2}\left(\sum_{\substack{x^{(i)},x^{(j)}\in\mathcal{X}\\ i\neq j}} x^{(i)T}Ax^{(j)} + \sum_{x^{(i)}\in\mathcal{X}} x^{(i)T}Ax^{(i)}\right) \\
&= \frac{1}{r^2}\left(r(r-1)x^{*T}Ax^* - \sum_{\substack{x^{(i)},x^{(j)}\in\mathcal{X}\\ i\neq j}} x^{(i)T}Ax^{(j)} + rx^{*T}Ax^* - \sum_{x^{(i)}\in\mathcal{X}} x^{(i)T}Ax^{(i)}\right) \\
&< \frac{1}{r^2}\left(r(r-1)\frac{\epsilon}{2} + r\right) \\
&\leq \frac{\epsilon}{2} + \frac{1}{r} \\
&= \epsilon,
\end{aligned}
$$

where the second to last inequality is implied from Lemma 10 which applies for every $x^{(i)}, x^{(j)} \in \mathcal{X}$ when $i \neq j$, and from the fact that $x^{*T}Ax^* - x^{(i)T}Ax^{(i)}$ is upper bounded by 1 for every $x^{(i)} \in \mathcal{X}$ (recall that the entries of A are in $[0,1]$).

The proof is concluded by observing that the vector x we created is a kr-uniform vector, for $k = 16\ln(3/\epsilon)/\epsilon^2$ and $r = 2/\epsilon$.

Corollary 3. *There is a PTAS for* SQP.

Proof. By Theorem 11, since the desired probability vector x that is suitable for the approximation is the mean of r many k-uniform vectors, x is kr-uniform. Therefore, it can be found by exhaustively searching through all possible multisets of $[p]$ created by sampling with replacement $kr = 32\ln(3/\epsilon)/\epsilon^3$ times. The number of all those possible multisets is $\binom{p+kr-1}{kr} \in O(p^{kr})$. For each multiset, i.e. vector x that the search algorithm takes into account, it picks the one that makes x^TAx maximum. This value is guaranteed to be ϵ-close to v^* by Theorem 11.

Hence, if we desire a $(1-\epsilon)$-approximation of SQP *in the weak sense* according to Definition 2.2 of [24], the described algorithm runs in time $O\left(p^{\ln(\frac{3}{\epsilon})/\epsilon^3}\right)$.

4.2. The general proof

4.2.1. Problems with multilinear constraints

We begin by considering constrained ϵ-ETR problems where the Boolean formula F consists of tensor multilinear polynomial constraints. We will use $\text{TML}(A, x_1, \ldots, x_n)$ to denote a tensor multilinear polynomial with n vector-variables and coefficients defined by tensor A of size $\times_{j=1}^{n} p$. Formally,

$$\text{TML}(A, x_1, \ldots, x_n) = \sum_{i_1 \in [p]} \cdots \sum_{i_n \in [p]} x_1(i_1) \cdot \ldots \cdot x_n(i_n) \cdot a(i_1, \ldots, i_n).$$

We will use α to denote the maximum entry of tensor A in the absolute value sense and γ to denote the infinite norm of the convex set that constrains the variables.

Lemma 12. *Let F be a Boolean formula with n vector-variables $x_1, x_2, \ldots, x_n$ and m TML constraints. Also, let $C_i = \text{conv}(c_1^i, c_2^i, \ldots, c_\ell^i)$ be the domain of x_i and $\mathcal{Y} = \bigtimes_{i=1}^{n} C_i$ be the domain of the variables. If the constrained* `ETR` *problem defined by* $\text{exact}(F)$ *and $\mathcal{Y}$ has a solution, then the constrained ϵ-ETR problem defined by F and $\mathcal{Y}$ has a k-uniform solution where*

$$k = \frac{2 \cdot \alpha^2 \cdot \gamma^2 \cdot n^2 \cdot \ln(3 \cdot n \cdot m)}{\epsilon^2}.$$

Proof. For every $i \in [n]$, let x_i' be a k-uniform vector sampled independently from x_i^*. To prove the lemma, we will show that, because of the choice of k, with positive probability the sampled vectors satisfy every constraint of the ϵ-ETR problem. Then, by the probabilistic method, the lemma will follow.

Let $\text{TML}_j(A_j, x_1, \ldots, x_n)$ be a multilinear polynomial that defines a constraint of F. For every $j \in [m]$ we define the following event

$$|\,\text{TML}_j(A_j, x_1', \ldots, x_n') - \text{TML}_j(A_j, x_1^*, \ldots, x_n^*)| \leq \epsilon. \tag{4}$$

Observe that if $x_1', \ldots, x_n'$ satisfy inequality (4) for every $j \in [m]$, then the lemma follows.

For every $j \in [m]$, we replace the corresponding event (4) with n events that are *linear* in each variable. For notation simplicity, let us denote by ML_j^i the multilinear polynomial $TML_j(A_j, x_1, \ldots, x_n)$ in which we have additionally set $x_1 = x_1', x_2 = x_2', \ldots, x_i = x_i'$ and $x_{i+1} = x_{i+1}^*, x_{i+2} = x_{i+2}^*, \ldots, x_n = x_n^*$. Furthermore, let $ML_j^0 = ML_j(A_j, x_1^*, \ldots, x_n^*)$. Then, for every $i \in [n]$ consider the event

$$|ML_j^i - ML_j^{i-1}| \leq \frac{\epsilon}{n}. \tag{5}$$

Observe that, if for a given $j \in [m]$ all n events defined in (5) are satisfied, then by the triangle inequality, the corresponding event (4) is satisfied as well.

Consider now ML_j^i. This can be seen as a random variable that depends on the choice of x_i' and takes values in $[-\gamma \cdot \alpha, \gamma \cdot \alpha]$. But recall that the x_i''s are sampled from x_i^* using k samples, and that they are mutually independent, so $\mathbb{E}\big[ML_j^i\big] = ML_j^{i-1}$. Thus, we can bound the probability that a constraint (5) is not satisfied, i.e. bound the probability that $|ML_j^i - ML_j^{i-1}| > \frac{\epsilon}{n}$, using Hoeffding's inequality [23]. So,

$$\begin{aligned}
\Pr\left(\left|ML_j^i - ML_j^{i-1}\right| > \frac{\epsilon}{n}\right) &= \Pr\left(\left|ML_j^i - \mathbb{E}\big[ML_j^i\big]\right| > \frac{\epsilon}{n}\right) \\
&\leq 2 \cdot exp\left(-\frac{2 \cdot k^2 \cdot \left(\frac{\epsilon}{n}\right)^2}{4 \cdot k \cdot \gamma^2 \cdot \alpha^2}\right) \\
&= 2 \cdot exp\left(-\frac{k \cdot \epsilon^2}{2 \cdot n^2 \cdot \gamma^2 \cdot \alpha^2}\right). \qquad (6)
\end{aligned}$$

Recall, that we have $n \cdot m$ events of the form (5). We can bound the probability that any of those events is violated, via the union bound. So, using (6) and the union bound, the probability that any of these events is violated is upper bounded by

$$2 \cdot m \cdot n \cdot exp\left(-\frac{k \cdot \epsilon^2}{2 \cdot n^2 \cdot \gamma^2 \cdot \alpha^2}\right). \qquad (7)$$

Hence, if the value of (7) is strictly less than 1, then there are $x_1', \ldots, x_m'$ such that all of the $n \cdot m$ events of (5) are realized with positive probability, therefore the events of (4) are realized with positive probability and thus the lemma follows. By requiring (7) to be strictly less than 1, and solving for k we get

$$k > \frac{2 \cdot \alpha^2 \cdot \gamma^2 \cdot n^2 \cdot \ln(2 \cdot n \cdot m)}{\epsilon^2}$$

which holds, by our choice of k.

4.2.2. Problems with a standard degree d constraint

We now consider constrained ϵ-ETR problems with *exactly one* tensor polynomial constraint of standard degree d. We will use $\mathrm{TSD}(A, x, d)$ to denote a standard degree d tensor-polynomial with coefficients defined by the $\times_{j=1}^{d} p$ tensor A. Here, d identical vectors x are applied on A. Formally,

$$\mathrm{TSD}(A, x, d) = \sum_{i_1 \in [p]} \cdots \sum_{i_d \in [p]} x(i_1) \cdot \ldots \cdot x(i_d) \cdot a(i_1, \ldots, i_d).$$

To prove the following lemma we consider the vector-variable x to be defined as the average of $r = O(\frac{\alpha^2 \cdot \gamma^d \cdot d^2}{\epsilon})$ variables. This allows us to "break" the standard degree d tensor polynomial to a sum of multilinear tensor polynomials and to a sum of not-too-many multivariate polynomials. Then, the choice of r allows us to upper bound by $\frac{\epsilon}{2}$ the error occurred by the sampled multivariate polynomials. Then, we observe that in order to prove the lemma we can write the sum of multilinear tensor polynomials as an $\frac{\epsilon}{2}$-ETR problem with r variables and roughly r^d multilinear constraints. This allows us to use Lemma 12 to complete the proof.

Lemma 13. *Let F be a Boolean formula with a single vector-variable and a single TSD constraint of standard degree d, let $\mathcal{Y}$ be a bounded convex set, and let $r = \frac{2 \cdot \alpha^2 \cdot \gamma^d \cdot d^2}{\epsilon}$. If the constrained* ETR *problem* $\text{exact}(F)$ *has a solution in $\mathcal{Y}$, then there exists a satisfiable constrained $\frac{\epsilon}{2}$-*ETR *problem Π_{ML} with r variables, where each variable is a k-uniform vector for $k = \frac{16 \cdot \alpha^4 \cdot \gamma^d \cdot d^4}{\epsilon^3}$. The Boolean formula of Π_{ML} is the conjunction of $\prod_{i=0}^{d-1}(r-i)$ many TML constraints, and every solution of Π_{ML} in $\mathcal{Y}$ can be transformed to a solution for the constrained ϵ-ETR problem defined by F and $\mathcal{Y}$.*

Proof. Assume that $x^* \in \mathcal{Y}$ is a solution for F. Let $\text{TSD}(A, x, d)$ denote the tensor polynomial of standard degree d used in F. For notation simplicity, let $\text{TSD}(A, x, d) = A(x^d)$. Create r new k-uniform variables $x_1, \ldots, x_r \in \mathcal{Y}(k)$ by sampling each one from x^*, where $\mathcal{Y}(k)$ is the discretized set made from $\mathcal{Y}$ by using k-uniform vectors, and set $x = \frac{1}{r}(x_1 + \ldots + x_r)$. Let $\mathcal{X} = \bigcup_{i=1}^{r}\{x_i\}$ be a *multiset* of $\mathcal{Y}(k)$ with cardinality r, meaning that multiple copies of an element of $\mathcal{Y}(k)$ are allowed in $\mathcal{X}$. In the sequel we will treat the elements of $\mathcal{X}$ as distinct, even though some might correspond to the same element of $\mathcal{Y}(k)$. Then, note that $A(x^d)$ can be written as a sum of simple tensor multivariate polynomials where some of them are multilinear and have as variables $x_1, \ldots, x_r$. Now, let $\mathcal{S}$ be the set of all ordered d-tuples that can be made by drawing d elements from $\mathcal{X}$ with replacement. Formally, $\mathcal{S} = \{(\hat{x}_1, \ldots, \hat{x}_d) : \hat{x}_1, \ldots, \hat{x}_d \in \mathcal{X}\}$. Let us also define $\mathcal{S}_d$ to be the set of all ordered d-tuples that can be made by drawing d elements from $\mathcal{X}$ without replacement. Formally, $\mathcal{S}_d = \{(\hat{x}_1, \ldots, \hat{x}_d) : \hat{x}_1, \ldots, \hat{x}_d \in \mathcal{X}, \quad \hat{x}_1, \ldots, \hat{x}_d \text{ are pairwise different}\}$, and observe that $|S_d| = \prod_{i=0}^{d-1}(r-i)$. So, any element of $\mathcal{S}_d$, combined with tensor A, produces a multilinear polynomial. Hence, using the notation introduced, we get that $|A(x^d) - A(x^{*^d})|$ is less than or equal to the sum of the following two sums

$$\frac{1}{r^d} \sum_{(\hat{x}_1,\ldots,\hat{x}_d)\in\mathcal{S}_d} \left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right| \qquad \text{and} \tag{8}$$

$$\frac{1}{r^d} \sum_{(\hat{x}_1,\ldots,\hat{x}_d)\in\mathcal{S}-\mathcal{S}_d} \left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right|. \tag{9}$$

Observe, $|\mathcal{S}-\mathcal{S}_d| = r^d - |\mathcal{S}_d|$ and that $\left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right| \leq \gamma^d \cdot \alpha$ for every $A(\hat{x}_1,\ldots,\hat{x}_d)$. Then, for the sum given in (9) we get

$$\begin{aligned}
&\frac{1}{r^d} \sum_{(\hat{x}_1,\ldots,\hat{x}_d)\in\mathcal{S}-\mathcal{S}_d} \left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right| \\
&\qquad \leq \left(1 - \frac{r\cdot(r-1)\cdots(r-d+1)}{r^d}\right)\cdot\gamma^d\cdot\alpha \\
&\qquad \leq \left(1 - \left(1-\frac{1}{r}\right)\left(1-\frac{2}{r}\right)\cdot\left(1-\frac{d-1}{r}\right)\right)\cdot\gamma^d\cdot\alpha \\
&\qquad \leq \left(1 - \left(1-\frac{d-1}{r}\right)^{d-1}\right)\cdot\gamma^d\cdot\alpha \\
&\qquad \leq \left(1 - \left(1-\frac{(d-1)^2}{r}\right)\right)\cdot\gamma^d\cdot\alpha \quad \text{(Bernoulli's inequality)} \\
&\qquad = \frac{(d-1)^2}{r}\cdot\gamma^d\cdot\alpha \\
&\qquad \leq \frac{\epsilon}{2}.
\end{aligned}$$

Hence, in order for the original constraint to be satisfied, it suffices to satisfy the constraint

$$\frac{1}{r^d} \sum_{(\hat{x}_1,\ldots,\hat{x}_d)\in\mathcal{S}_d} \left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right| \leq \frac{\epsilon}{2}. \tag{10}$$

Observe that $|\mathcal{S}_d| = \prod_{i=0}^{d-1}(r-i) < r^d$, therefore, instead of the constraint (10), it suffices to satisfy the following $|S_d|$ constraints (we introduce one constraint for every $(\hat{x}_1,\ldots,\hat{x}_d)\in\mathcal{S}_d$)

$$\left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right| \leq \frac{\epsilon}{2}. \tag{11}$$

Note that each constraint (11) is the relaxed by $\epsilon/2$ version of a constraint with a multilinear function equal to 0; multilinearity is due to the fact that $\hat{x}_1,\ldots,\hat{x}_d$ are pairwise different by definition of the set S_d. The proof is completed by using Lemma 12 for $n = d$, $m = |S_d|$ and $\epsilon/2$ instead of ϵ to show that indeed there exists a collection S_d of tuples $\hat{x}_1,\ldots,\hat{x}_d$, where each $\hat{x}_i$, $i\in[d]$ is a k-uniform vector with $k \geq \frac{8\cdot\alpha^2\cdot\gamma^2\cdot d^2(d+2)\cdot\ln r}{\epsilon^2}$ such that all $|S_d|$ constraints of (11) are satisfied. The latter inequality is true by our choice of k and r.

Now we can prove the following lemma.

Lemma 14. *Let F be a Boolean formula with a single vector-variable x and a single TSD constraint of standard degree d, and let the domain of x, namely $\mathcal{Y}$, be a bounded convex set. If the constrained* `ETR` *problem defined by* $\mathrm{exact}(F)$ *and $\mathcal{Y}$ has a solution, then the constrained ϵ-ETR problem defined by F and $\mathcal{Y}$ has a k-uniform solution where*

$$k = \frac{32\cdot\alpha^6\cdot\gamma^{2d}\cdot d^6}{\epsilon^4}.$$

Proof. First, we use Lemma 13 to construct the constrained $\frac{\epsilon}{2}$-`ETR` problem Π_{ML} with tensor multilinear constraints. Recall that Π_{ML} has $r = \frac{2\cdot\alpha^2\cdot\gamma^d\cdot d^2}{\epsilon}$ variables and if Π_{ML} is satisfiable, then there exist $\frac{k}{r}$-uniform vectors $\hat{x}_1 \in \mathcal{Y}, \ldots, \hat{x}_r \in \mathcal{Y}$ that $\epsilon/2$-satisfy Π_{ML}. Then, let us construct the k-uniform vector $\hat{x} = \frac{1}{r}\cdot(\hat{x}_1 + \ldots + \hat{x}_r)$. Note that, according to Lemma 13, it is

$$\begin{aligned} |A(\hat{x}^d) - A(x^{*^d})| &\leq \frac{1}{r^d} \sum_{(\hat{x}_1,\ldots,\hat{x}_d)\in\mathcal{S}_d} \left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right| \\ &\quad + \frac{1}{r^d} \sum_{(\hat{x}_1,\ldots,\hat{x}_d)\in\mathcal{S}-\mathcal{S}_d} \left|A(\hat{x}_1,\ldots,\hat{x}_d) - A(x^{*^d})\right| \\ &\leq \frac{\epsilon}{2} + \frac{\epsilon}{2} \\ &= \epsilon. \end{aligned}$$

This completes the proof of the lemma.

4.2.3. Problems with simple multivariate constraints

We now assume that we are given a constrained-ϵ-ETR problem defined by a Boolean formula F of simple tensor multivariate polynomial constraints and a bounded convex set $\mathcal{Y}$. In the sequel, we will denote the maximum absolute value of a coordinate of a vector in $\mathcal{Y}$ by $\|\mathcal{Y}\|_\infty$. As before, $\gamma = \|\mathcal{Y}\|_\infty$ and let α be the maximum absolute value of the coefficients of the constraints. We will say that the constraints are of maximum degree d if d is the maximum degree among all vector-variables. The main idea of the proof of the following lemma is to rewrite the problem as an equivalent problem with standard degree d constraints and then apply Lemmas 14 and 12 to derive the bound for k.

Lemma 15. *Let F be a Boolean formula with n vector-variables $x_1, x_2, \ldots, x_n$ and m many STM polynomial constraints. Also, let $C_i = \operatorname{conv}(c_1^i, c_2^i, \ldots, c_\ell^i)$ be the domain of x_i and $\mathcal{Y} = \bigtimes_{i=1}^n C_i$ be the domain of the variables. If the constrained* ETR *problem defined by* $\operatorname{exact}(F)$ *and $\mathcal{Y}$ has a solution, then the constrained ϵ-ETR problem defined by F and $\mathcal{Y}$ has a k-uniform solution where*

$$k = \frac{512 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot n^6 \cdot d^6 \cdot \ln(2 \cdot \alpha' \cdot \gamma' \cdot d \cdot n \cdot m)}{\epsilon^5},$$

where $\alpha' := \max(\alpha, 1), \gamma' := \max(\gamma, 1)$.

Proof. Let $x_1^*, \ldots, x_n^*$ be a solution for $\operatorname{exact}(F)$ and let x_i', $i \in [n]$ be a k-uniform vector-variable sampled from x_i^*. We will prove that if k equals at least the quantity of the statement of the lemma, then there exist vectors $x_1', \ldots, x_n'$ that constitute a solution to the constrained ϵ-ETR problem defined by F and $\mathcal{Y}$.

Consider the j-th constraint where $j \in [m]$ defined by the simple tensor multivariate polynomial $\operatorname{STM}(A_j, x_1^{d_{j1}}, \ldots, x_n^{d_{jn}})$. We will use the same technique we used in Lemma 12 to create n constraints, where constraint $i \in [n]$ is defined via a simple degree d_{ji} polynomial. Again, for notation simplicity for every $i \in [m]$ we use STM_j^i to denote the polynomial $\operatorname{STM}(A_j, x_1^{d_{j1}}, \ldots, x_n^{d_{jn}})$ where we set $x_1 = x_1', \ldots, x_i = x_i'$ and $x_{i+1} = x_{i+1}^*, \ldots, x_n = x_n^*$. Let $\operatorname{STM}_j^0 := \operatorname{STM}(A_j, (x_1^*)^{d_{j1}}, \ldots, (x_n^*)^{d_{jn}})$. Then, for every $j \in [m]$ we define the following n constraints

$$|\operatorname{STM}_j^i - \operatorname{STM}_j^{i-1}| \leq \frac{\epsilon}{n}. \tag{12}$$

Observe that for some $j \in [m]$, every constraint i of the form (12) defines a simple degree d_{ji} polynomial with respect to variable x_i'. Furthermore, observe that if every such constraint is satisfied, then the initial constraint defined by $\operatorname{STM}(A_j, x_1^{d_{j1}}, \ldots, x_n^{d_{jn}})$ is satisfied too. Then, we convert each such constraint to a set of $\prod_{i=0}^{d-1}(r-i)$ multilinear constraints with $r = \frac{2 \cdot \alpha^2 \cdot \gamma^d \cdot d^2}{\epsilon}$ variables, using Lemma 13 where we demand that every multilinear constraint is $\frac{\epsilon}{2n}$-satisfied (we restrict the current $\frac{\epsilon}{n}$ to half of it in order to use Lemma 13). The proof is then completed by using Lemma 12 where we observe that we have $r \cdot n = \frac{2 \cdot \alpha^2 \cdot \gamma^d \cdot d^2 \cdot n}{\epsilon}$ variables and $\prod_{i=0}^{d-1}(r-i) \cdot n \cdot m < r^d \cdot n \cdot m$ constraints and we set ϵ to $\frac{\epsilon}{2n}$.

To arrive to the actual size k of the required uniform vector, we start from the size k' prescribed by Lemma 12 and sequentially set proper values for the parameters as dictated by our method for transforming the constraints. We have

$$
\begin{aligned}
k' &= \frac{2 \cdot \alpha^2 \cdot \gamma^2 \cdot n^2 \cdot \ln(3 \cdot n \cdot m)}{\epsilon^2} \\
&= \frac{8 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot n^2 \cdot d^4 \cdot \ln(6 \cdot \alpha^2 \cdot \gamma^d \cdot d^2 \cdot n \cdot m/\epsilon)}{\epsilon^4} \qquad (n \leftarrow n \cdot r) \\
&= \frac{8 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot n^2 \cdot d^4 \cdot \ln(6 \cdot \alpha^{2d+2} \cdot \gamma^{d^2+d} \cdot d^{2d+2} \cdot n^2 \cdot m/\epsilon^{d+1})}{\epsilon^4} \qquad (m \leftarrow r^d \cdot n \cdot m) \\
&= \frac{128 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot n^6 \cdot d^4 \cdot \ln(6 \cdot 2^{d+1} \cdot \alpha^{2d+2} \cdot \gamma^{d^2+d} \cdot d^{2d+2} \cdot n^{d+3} \cdot m/\epsilon^{d+1})}{\epsilon^4} \qquad (\epsilon \leftarrow \frac{\epsilon}{2n}) \\
&\leq \frac{128 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot n^6 \cdot d^4 \cdot \ln(2 \cdot \max(\alpha,1) \cdot \max(\gamma,1) \cdot d \cdot n \cdot m/\epsilon)^{4d^2}}{\epsilon^4} \qquad (\text{for any } d \geq 1) \\
&= \frac{512 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot n^6 \cdot d^6 \cdot \ln(2 \cdot \max(\alpha,1) \cdot \max(\gamma,1) \cdot d \cdot n \cdot m/\epsilon)}{\epsilon^4} \\
&\leq \frac{512 \cdot \alpha^6 \cdot \gamma^{2d+2} \cdot n^6 \cdot d^6 \cdot \ln(2 \cdot \max(\alpha,1) \cdot \max(\gamma,1) \cdot d \cdot n \cdot m)}{\epsilon^5}.
\end{aligned}
$$

We want $k \geq k'$, therefore it suffices to bound from below k by the upper bound of k'. This completes the proof.

4.2.4. Putting everything together

Proof. For the final step of the proof of Theorem 5, assume that $x_1^*, \ldots, x_n^* \in \mathcal{Y}$ is a solution for exact(F). Consider now a multivariate constraint $i \in [m]$ of F defined by $TMV_i(x_1, \ldots, x_n)$. First, we replace this constraint by

$$|TMV_i(x_1, \ldots, x_n) - TMV_i(x_1^*, \ldots, x_n^*)| \leq \epsilon. \tag{13}$$

Then, replace constraint (13) by t constraints of the form

$$|STM_{i,j}(x_1, \ldots, x_n) - STM_{i,j}(x_1^*, \ldots, x_n^*)| \leq \frac{\epsilon}{t} \tag{14}$$

where $STM_{i,1}(x_1, \ldots, x_n), \ldots, STM_{i,t}(x_1, \ldots, x_n)$ are the simple tensor multivariate polynomials $TMV_i(x_1, \ldots, x_n)$ consists of. By the triangle inequality we get that if all t constraints given by (14) hold, then constraint (13) holds as well. Hence, we can reduce the problem to an equivalent problem with the same n variables and $m \cdot t$ constraints that all of them are simple tensor multivariate polynomials. So, we can apply Lemma 15 where we replace m with $m \cdot t$ and ϵ with $\frac{\epsilon}{t}$. This completes the proof of the theorem.

5. Applications

We now show how our theorems can be applied to derive new approximation algorithms for a variety of problems. In order to conclude that Corollary 1 provides a PTAS or QPTAS for some given problem, one has to carefully determine the actual input size of the problem and show that the running time of the corollary's algorithm satisfies the PTAS or QPTAS definition.

5.1. Constrained approximate Nash equilibria

A *constrained* Nash equilibrium is a Nash equilibrium that satisfies some extra constraints, like specific bounds on the payoffs of the players. Constrained Nash equilibria attracted the attention of many authors, who proved NP-completeness for two-player games [25, 26, 10] and ETR-completeness for three-player games [10, 11, 12, 13, 14] for constrained *exact* Nash equilibria.

Constrained approximate equilibria have been studied, but so far only lower bounds have been derived [27, 28, 16, 17, 8]. It has been observed that sampling methods can give QPTASs for finding constrained approximate Nash equilibria for certain constraints in two player games [17].

By applying Theorem 5, we get the following result for games with number of players up to polylogarithmic in the number of pure strategies (here n is the number of players): *Any property of an approximate equilibrium that can be formulated in ϵ-ETR where α, γ, d, t and n are up to polylogarithmic in the number of pure strategies has a QPTAS.* This generalises past results to a much broader class of constraints, and provides results for games with more than two players, which had not previously been studied in this setting.

A game is defined by the set of players, the set of actions for every player, and the payoff function of every player. In normal form games, the payoff function is given by a multilinear function on a tensor of appropriate size. Consider an n-player game where every player has ℓ many actions, and let A_j denote the payoff tensor of player j with elements in $[0,1]$; A_j has size $\times_{i=1}^{n} \ell$. The interpretation of the tensor A_j is the following: the element $A_j(i_1, \ldots, i_n)$ of the tensor corresponds to the payoff of player j when Player 1 chooses action i_1, Player 2 chooses action i_2, and so on. To play the game, every player j chooses a probability distribution $x_j \in \Delta^{\ell}$, a.k.a. a *strategy*, over their actions. A collection of strategies is called *strategy profile*. The expected payoff of player j under the strategy profile $(x_1, \ldots, x_n)$ is given by $ML(A_j, x_1, \ldots, x_n)$. For notation simplicity, let $u_j(x_j, x_{-j}) := ML(A_j, x_1, \ldots, x_n)$, where x_{-j} is the strategy profile of all players except player j. A strategy profile

$(x_1^*, \ldots, x_n^*)$ is a Nash equilibrium if for every player j it holds that $u_j(x_j^*, x_{-j}^*) \geq u_j(x_j, x_{-j}^*)$ for every $x_j \in \Delta^\ell$, or equivalently $u_j(x_j^*, x_{-j}^*) \geq u_j(s_p, x_{-j}^*)$ for every possible s_p, where s_p denotes the probability distribution of player j where her p-th action has probability 1.

Our framework formally describes a broad family of constrained Nash equilibrium problems for which we can get a QPTAS.

Theorem 16. *Let Γ be an n-player ℓ-action normal form game Γ. Furthermore, let F be a Boolean formula with $c \in poly(\ell)$ TMV constraints of degree d. If $n, d \in poly\log(\ell)$, then in quasi-polynomial time we can compute an approximate NE of Γ constrained by F, or decide that no such constrained approximate NE exists.*

Proof. Observe that we can write the problem of the existence of a constrained Nash equilibrium as an `ETR` problem. The constraints of the problem will be the constraints of F plus the constraint

$$u_j(s_p, x_{-j}) - u_j(x_j, x_{-j}) \leq 0$$

for every player $i \in [m]$ and every action s_p of player j.

Thus, we can use Theorem 5 and complete the proof since we produced an ϵ-ETR problem with $m = c + n \cdot \ell = \text{poly}(\ell)$ constraints, which is polynomial in the input size; d and t are polylogarithmic in ℓ by assumption (it always holds that $t \leq d$); $\gamma = 1$ since every variable is a probability distribution; $\alpha = 1$ by the definition of normal form games.

5.2. Shapley games

Shapley's stochastic games [29] describe a two-player infinite-duration zero-sum game. The game consists of N *states*. Each state specifies a two-player $M \times M$ bimatrix game where the players compete over: (1) a reward (which may be negative) that is paid by player two to player one, and (2) a probability distribution over the next state of the game. So each round consists of the players playing a bimatrix game at some state s, which generates a reward, and the next state s' of the game. The reward in round i is discounted by λ^{i-1}, where $0 < \lambda < 1$ is a *discount factor*. The overall payoff to player 1 is the discounted sum of the infinite sequence of rewards generated during the course of the game.

Shapley showed that these games are determined, meaning that there exists a value vector v, where v_s is the value of the game starting at state s. A polynomial time algorithm has been devised for computing the value vector of a Shapley game when the number of states N is constant [30]. However, since the values may be irrational, this algorithm needs to deal with algebraic numbers, and the *degree* of the polynomial is $O(N)^{N^2}$, so if N is even mildly super-constant, then the algorithm is not polynomial.

Furthermore, Shapley showed that the value vector is the unique solution of a system of polynomial optimality equations, which can be formulated in ETR. Any approximate solution of these equations gives an approximation of the value vector, and applying Theorem 5 gives us a QPTAS. This algorithm works when $N \in O(\sqrt[6]{\log M})$, which is a value of N that prior work cannot handle. The downside of our algorithm is that, since we require the solution to be bounded by a convex hull defined by finitely many points, the algorithm only works when the value vector is reasonably small. Specifically, the algorithm takes a constant bound $B \in \mathbb{R}$, and either finds the approximate value of the game, or verifies that the value is strictly greater than B.

To formally define a Shapley game, we use N to denote the number of states, and M to denote the number of actions. The game is defined by the following two functions.

- For each $s \leq N$ and $j, k \leq M$ the function $r(s, j, k)$ gives the reward at state s when player one chooses action j and player two chooses action k.
- For each $s, s' \leq N$ and $j, k \leq M$ the function $p(s, s', j, k)$ gives the probability of moving from state s to state s' when player one chooses action j and player two chooses action k. It is required that $\sum_{s'=1}^{N} p(s, s', j, k) = 1$ for all s, j, and k.

The game begins at a given starting state. In each round of the game the players are at a state s, and play the matrix game at that state by picking an action from the set $\{1, 2, \ldots, M\}$. The players are allowed to use randomization to make this choice. Supposing that the first player chose action j and the second player chose the action k, the first player receives the reward $r(s, j, k)$, and then a new state s' is chosen according to the probability distribution given by $p(s, \cdot, j, k)$.

The reward in future rounds is *discounted* by a factor of λ where $0 < \lambda < 1$ in each round. So if r_1, r_2, ... is the infinite sequence of rewards, the total reward paid by player two to player one is $\sum_{i=1}^{\infty} \lambda^{i-1} \cdot r_i$, which, due to the choice of λ, is always a finite value.

The two players play the game by specifying a probability distribution at each state, which represents their strategy for playing at that state. Let Δ^M denote the M-dimensional simplex, which represents the strategy space for both players at a single state. For each $x, y \in \Delta^M$, we overload notation by defining the expected reward and next state functions.

$$r(s,x,y) = \sum_{j=1}^{M}\sum_{k=1}^{M} x(j) \cdot y(k) \cdot r(s,i,j),$$
$$p(s,s',x,y) = \sum_{j=1}^{M}\sum_{k=1}^{M} x(j) \cdot y(k) \cdot p(s,s',i,j).$$

Shapley showed that these games are *determined* [29], meaning that there is a unique vector $v \in \mathbb{R}^N$ such that v_s is the *value* of the game starting at state s: player one has a strategy to ensure that the expected reward is at least $v(s)$, while player two has a strategy to ensure that the expected reward is at most $v(s)$. Furthermore, Shapley showed that this value vector is the unique solution of the following *optimality equations* [29]. For each state s we have the equation

$$v(s) = \min_{x \in \Delta^M} \max_{y \in \Delta^M} \left(r(s,x,y) + \lambda \cdot \sum_{s'=1}^{N} p(s,s',x,y) \cdot v_{s'} \right). \tag{15}$$

In other words, v_s must be the value of the one-shot zero-sum game at s, where the payoffs of this zero-sum game are determined by the values of the other states given by $v_{s'}$.

Theorem 17. *Let Γ be a Shapley game with $N \in O(\sqrt[6]{\log M})$, unbounded number of actions per state, and rewards in $[-c, c]$ for every state-action combination, where c is a constant. Furthermore, let s be the starting state of the game. Let $B \in \mathbb{R}$ be a constant. In quasi-polynomial time we can approximately compute the value of Γ starting from s, if the value of every state is less than or equal to B, or decide that at least one of these values is greater than or equal to B.*

Proof. Let $v = (v(1), v(2), \ldots, v(N))$, and for every state s let x_s and y_s denote the strategy player one and player two choose at state s respectively. Observe that $r(s, x_s, y_s)$ is an STM polynomial with variables x and y of the form

$$\mathrm{STM}(A_{s1}, x_s, y_s) = \sum_{j=1}^{M}\sum_{k=1}^{M} x_s(j) \cdot y_s(k) \cdot a_{s1}(j,k)$$

where $a_{s1}(i,j,k) = r(s,j,k)$.

Observe also that $\lambda \cdot \sum_{s'=1}^{N} p(s,s',x_s,y_s) \cdot v_{s'}$ can be written as an STM polynomial with variables

x, y and v of the form

$$\mathrm{STM}(A_{s2}, x, y, v) = \sum_{j=1}^{M}\sum_{k=1}^{M}\sum_{l=1}^{N} x_s(j)\cdot y_s(k)\cdot v(l)\cdot a_{s2}(j,k,l)$$

where $a_{s2}(i,j,k) = \lambda \cdot p(s,l,j,k)$.

Let us define $\mathrm{TMV}_s(x_s, y_s, v) = \mathrm{STM}(A_{s1}, x_s, y_s) + \mathrm{STM}(A_{s2}, x_s, y_s, v)$; $\mathrm{TMV}_s(x_s, y_s, v)$ has length 2 and degree 1.

Note that we can replace equation (15) with the following $2 \cdot M$ TMV polynomial constraints

$$\begin{aligned} \mathrm{TMV}(x_s, y_s, v) - \mathrm{TMV}(j, y_s, v) &\leq 0 && \text{for every action } j \leq M \text{ of player one} \\ \mathrm{TMV}(x_s, k, v) - \mathrm{TMV}(x_s, y_s, v) &\geq 0 && \text{for every action } k \leq M \text{ of player two.} \end{aligned}$$

So, to approximate $v(s)$ it suffices to solve the ϵ-ETR problem defined by the $2 \cdot M \cdot N$ constraints defined as above for every state $s \leq N$. Observe, the ϵ-ETR problem has: $2N + 1$ variables (x_1 through x_N, y_1 through y_N, and v); $2 \cdot M \cdot N$ TMV constraints; $\gamma = \max\{1, \max_s v(s)\}$; $\alpha = \max\{c, \lambda \cdot \max_{s,s',j,k} p(s,s',j,k)\} = \max\{c, 1\}$, since $\lambda < 1$ and $\max_{s,s',j,k} p(s,s',j,k) < 1$. So, if $N \in O(\sqrt[6]{\log M})$, $\max_s v(s)$ is constant, and c is a constant, we can use Theorem 5 and derive a QPTAS for (15).

Finally, we note that an approximate solution to (15) gives an approximation of the value vector itself. This is because Shapley has shown that, when v is treated as a variable, the optimality equation given in (15) is a *contraction map*. The value vector is a fixed point of this contraction map, and the uniqueness of the value vector is guaranteed by Banach's fixed point theorem. Our algorithm produces an approximate fixed point of the optimality equations. It is easy to show, using the contraction map property, that an approximate fixed point must be close to an exact fixed point.

5.3. Approximate consensus halving

In this section we show that an approximate solution to the consensus halving problem can be found in quasi-polynomial time when each agent's valuation function is a single polynomial of constant or even polylogarithmic degree. We will do so by formulating the problem as a constrained ϵ-ETR instance, and then applying Theorem 5.

In the consensus halving problem, typically, we consider n agents, each having a valuation function $f_i : [0,1] \mapsto \mathbb{R}$ over the interval $[0,1]$. We often consider an equivalent version of the problem whose input consists of the cumulative valuations F_i instead, i.e. $F_i(x) := \int_0^x f(y)dy$. A solution to the problem is given by a k-cut, i.e., a partition of $[0,1]$ into $k+1$ sub-intervals, and a labeling of each as "$+$" or "$-$", such that the total valuation of each agent in her positive parts A_+ equals that of the negative parts A_-. In other words, we should have $F_i(A_+) = F_i(A_-)$ for every $i \in [n]$. It was proven in [31] that there is always such a solution for $k = n$, and it is also easy to check that in the worst case that many cuts are necessary: consider each of the valuations of the agents having support that is a single sub-interval, and all the agents' sub-intervals are disjoint. In the approximate version of the problem, for a given $\epsilon > 0$, we are asking for a cut and a labeling such that $|F_i(A_+) - F_i(A_-)| \leq \epsilon$.

The result of this section first appeared in [32, 33] and implies that instances in which each agent's valuation function is a single polynomial, can be solved approximately using a polylogarithmic number of cuts. Furthermore, the cuts have a special form, that is, they are k-uniform. We note that this is one of the most general classes of instances for which we could hope to prove such a result: any instance in which n agents desire completely disjoint portions of the object can only be solved by an n-cut, and piecewise linear functions are capable of producing such a situation. So in a sense, we are exploiting the fact that this situation cannot arise when the agents have non-piecewise polynomial valuation functions.

Lemma 18. *For every* Consensus Halving *instance with n agents, and every $\epsilon > 0$, if each agent's valuation function F_i is a single polynomial of degree at most $O(poly \log n)$, then there exists a k-cut, where $k := O(poly \log n)/\epsilon^5$, and parts A_+ and A_- such that:*

- *every cut point is a multiple of $1/k = \frac{\epsilon^5}{O(poly \log n)}$;*
- *$|F_i(A_+) - F_i(A_-)| \leq \epsilon$, for every agent i.*

Proof. Since each agent i has a polynomial valuation function, there is a $d \in O(\log n)$ and constants $a_0, a_1, \ldots, a_d$ such that each function F_i can be written as $F_i(t) = \sum_{j=0}^{d} a_j \cdot t^j$.

To prove the lemma, we will formulate the problem as a constrained ϵ-ETR instance, and apply Theorem 5, which proves the claim. We first write a simple `ETR` formula for consensus halving with polynomial valuation functions. If a consensus halving instance has a solution, then it also has one in which the cuts are *strictly alternating*, meaning that

$$F_i(A_+) = \sum_{j=1}^{\lfloor n/2 \rfloor} \big(F_i(t_{2j}) - F_i(t_{2j-1})\big),$$
$$F_i(A_-) = \sum_{j=1}^{\lceil n/2 \rceil} \big(F_i(t_{2j-1}) - F_i(t_{2j-2})\big),$$

where the cut is the tuple $(t_1, t_2, \ldots, t_n)$, with $0 \leq t_1 \leq \cdots \leq t_n \leq 1$ and $t_0 := 0$, $t_{n+1} := 1$.

In this encoding, we have no need to encode which set a particular cut belongs to, and so we can encode an n-cut as an element of the n-simplex $x := (x_1, x_2, \ldots, x_{n+1}) \in \Delta^{n+1}$, where $x_i := t_i - t_{i-1}$. From the latter, it is easy to see that

$$t_i := \sum_{j=1}^{i} x_j. \tag{16}$$

For $j \in \{0, 1, \ldots, n\}$, let us denote by 1^j and 0^j a j-tuple of 1's and 0's respectively. Let us also define the n-dimensional vector $v_j := (0^j, 1^{n-j})$. Now observe that any n-cut $t := (t_1, t_2, \ldots, t_n)$ can be represented by an n-dimensional point which is in fact a convex combination of the $n+1$ vectors v_j, $j \in \{0, 1, \ldots, n\}$. In particular, from (16) it is easy to see that

$$t := (t_1, t_2, \ldots, t_n) = \sum_{j=1}^{n+1} x_j \cdot v_{j-1}.$$

Hence, we can encode the problem as an ETR formula

$$\exists t \cdot \left(\bigwedge_{i=1}^{n} F_i(A_+) = F_i(A_-) \right) \wedge t \in C,$$

where C is the convex hull of the vectors $v_0, v_1, \ldots, v_n$. This formula has n constraints, one for each agent, and a single constraint bounding the variables in the convex set C which can be expressed by $n+1$ vectors, namely v_j, $j \in \{0, 1, \ldots, n\}$.

Theorem 5 allows us to leave the constraint $t \in C$ unchanged, but insists that we weaken the others. Specifically each constraint is weakened so that only $F_i(A_+) - F_i(A_-) \leq \epsilon$ and $F_i(A_+) - F_i(A_-) \geq -\epsilon$ are enforced, which implies that $|F_i(A_+) - F_i(A_-)| \leq \epsilon$. This is sufficient to encode an approximate solution to the problem.

The constructed ϵ-ETR instance has one vector-variable $t \in C$ and $2n$ constraints. Let us now study one of the constraints of the ϵ-ETR instance.

$$\sum_{j=1}^{\lfloor n/2 \rfloor} \big(F_i(t_{2j}) - F_i(t_{2j-1})\big) - \sum_{j=1}^{\lceil n/2 \rceil} \big(F_i(t_{2j-1}) - F_i(t_{2j-2})\big) \leq \epsilon.$$

Using the representation of F_i, we can write down a constraint as $\sum_{k=0}^{d} a_k \cdot h_k(t_1, t_2, \dots, t_n) \leq \epsilon$, where $h_k(t_1, t_2, \dots, t_n)$ is a sum of monomials, each one of degree d. F_i depends on t_0 and t_{n+1} as well, but recall that these are 0 and 1 respectively.

The term $a_k \cdot h_k(t_1, t_2, \dots, t_n)$ is a simple tensor multivariate polynomial with one variable of degree k, which we will denote by $STM(H_k, t^k)$. Under this notation H_k is a k-dimensional tensor where vector t is applied k times. Hence, every constraint is a sum of $d+1$ simple tensor multivariate polynomials, i.e. a TMV polynomial of maximum degree d constructed by $d+1$ STM polynomials. Furthermore, $||v_j||_\infty \leq 1$ for all $j \in \{0, 1, \dots, n\}$, and for every constraint, the maximum absolute coefficient is constant by definition, and the degree d is $O(poly \log n)$. Hence, we can apply Theorem 5 and get the claimed result.

As a consequence, we can perform a brute force search over all possible k-cuts to find an approximate solution, which can be carried out in $n^{O(poly \log n/\epsilon^5)}$ time.

Theorem 19. Consensus Halving *admits a QPTAS when the valuation function of every agent is a single polynomial of degree $O(poly \log n)$.*

5.4. Optimization problems

Our framework can provide approximation schemes for optimization problems with one vector-variable $x \in \mathbb{R}^p$ with polynomial constraints over bounded convex sets. Formally,

$$\begin{aligned} \max \quad & h(x) \\ s.t. \quad & h_1(x) \geq 0, \dots, h_m(x) \geq 0 \\ & x \in \text{conv}(c_1, \dots, c_\ell) \end{aligned}$$

where $h(x), h_1(x), \dots, h_m(x)$ are polynomials with respect to vector x; for example $h(x) = x^T A x$, where A is a $p \times p$ matrix, subject to $h_1(x) = x^T x - \frac{1}{10} \geq 0$ and $x \in \Delta^p$. We will call the polynomials

h_i *solution-constraints*. Optimization problems of this kind received a lot of attention over the years [24, 34, 35, 36].

For optimization problems, we sample from the solution that achieves the maximum when we apply Theorem 5, in order to prove that there is a k-uniform solution that is close to the maximum. Our algorithm enumerates all k-uniform profiles, and outputs the one that maximizes the objective function. Using this technique, Theorem 5 gives the following results.

1. There is a PTAS if $h(x)$ is a STM polynomial of maximum degree independent of p, the number of solution-constraints is independent of p, and $\ell = \text{poly}(p)$.

2. There is QPTAS if $h(x)$ is a STM polynomial of maximum degree up to $\text{poly} \log p$, the number of solution-constraints is $\text{poly}(p)$, and $\ell = \text{poly}(p)$.

To the best of our knowledge, the second result is new. The first result was already known, however it was proven using completely different techniques: in [22] it was proven for the special case of degree two, in [36] it was extended to any fixed degree, and alternative proofs of the fixed degree case were also given in [34, 35]. We highlight that in all of the aforementioned results solution constraints were not allowed. Note that unless $\texttt{NP} = \texttt{ZPP}$ there is no FPTAS for quadratic programming even when the variables are constrained in the simplex [24]. Hence, our results can be seen as a partial answer to the important question posed in [24]: *What is a complete classification of functions that allow a PTAS?*

Furthermore, as shown in Theorem 6 this technique yields a generalized algorithm for multi-objective optimization problems which, to the best of our knowledge, is a completely new result.

5.5. Tensor problems

Our framework provides quasi-polynomial time algorithms for deciding the existence of approximate eigenvalues and approximate eigenvectors of tensors in $\mathbb{R}^{p\times p\times p}$, where the elements are bounded by a constant, where the solutions are required to be in a bounded convex set. In [37] it is proven that there is no PTAS for these problems when the domain is unrestricted. To the best of our knowledge this is the first positive result for the problem even in this, restricted, setting.

Definition 8. The nonzero vector $x \in \mathbb{R}^p$ is an *eigenvector* of tensor $A \in \mathbb{R}^{p\times p\times p}$ if there exists an *eigenvalue* $\lambda \in \mathbb{R}$ such that for every $k \in [p]$ it holds that

$$\sum_{i}^{n} \sum_{j}^{n} a(i,j,k) \cdot x(i) \cdot x(j) = \lambda \cdot x(k). \tag{17}$$

Theorem 20. *Let A be an $\mathbb{R}^{p\times p\times p}$ tensor with entries in $[-c,c]$, where c is a constant. Furthermore, let $B \in \mathbb{R}$ be a constant and let $\mathcal{Y}$ be a bounded convex set where $\|\mathcal{Y}\|_\infty$ is a constant. In a quasi-polynomial time we can compute an eigenvalue-eigenvector pair (λ, x) that approximately satisfy* (17) *such that $\lambda \leq B$ and $x \in \mathcal{Y}$, or decide that no such pair exists.*

Proof. Observe that $\sum_i^n \sum_j^n a(i,j,k) \cdot x(i) \cdot x(j)$ can be written as an STM polynomial $\mathrm{STM}(A_1, x^2)$ where $a_1(i,j) = a(i,j,k)$. Furthermore, let ℓ be a p-dimensional vector. Then, $\lambda \cdot x(k)$ can be written as an STM polynomial $\mathrm{STM}(A_2, x, \ell)$, where $a_2(k,1) = 1$ and zero otherwise.

So, Equation (17) can be written as an TMV polynomial constraint of degree 2 and length 2, with two vector variables, x and ℓ. So, the problem of computing an eigenvalue-eigenvector pair that approximately satisfy (17) can be written as an ϵ-ETR problem with p TMV polynomial constraints of degree 2 and length 2 and two vector variables. Hence, we can use Theorem 5 with $\gamma = \|\mathcal{Y}\|_\infty$ which is constant, $\alpha = c$, $n = 2$, $t = 2$, $d = 2$, and $m = p$ to find an approximate solution if an exact one exists, or decide that no exact solution exists.

5.6. Computational geometry

Finally, we note that our theorem can be applied to problems in computational geometry, although the results are not as general as one may hope. Many problems in this field are known to be ETR-complete, including, for example, the Steinitz problem for 4-polytopes, inscribed polytopes and Delaunay triangulations, polyhedral complexes, segment intersection graphs, disk intersection graphs, dot product graphs, linkages, unit distance graphs, point visibility graphs, rectilinear crossing number, and simultaneous graph embeddings. We refer the reader to the survey of Cardinal [38] for further details.

All of these problems can be formulated in ϵ-ETR, and indeed our theorem does give results for these problems. However, our requirement that the bounding convex set be given explicitly limits their applicability. Most computational geometry problems are naturally constrained by a cube, so while Corollary 2 does give NP algorithms, we do not get QPTASs unless we further restrict the convex set. Here we formulate QPTASs for the segment intersection graph and the unit disk intersection graph problems when the solutions are restricted to lie in a simplex. While it is not clear that either problem has natural applications that are restricted in this way, we do think that future work may be able to derive sampling theorems that are more tailored towards the computational geometry setting.

5.6.1. Segment intersection graphs

Definitions.. Let G be an undirected graph with vertex set $\{v_1, v_2, \ldots, v_n\}$. We say that G is a *segment graph* if there are straight segments $s_1, s_2, \ldots, s_n$ in the plane such that, for every $i, j, 1 \leq i < j \leq n$, the segments s_i and s_j intersect if and only if $\{v_i, v_j\} \in E(G)$.

By a suitable rotation of the co-ordinate system we can achieve that none of the segments is vertical. Then the segment s_i representing vertex v_i can be algebraically described as the set $\{(x, y) \in \mathbb{R}^2 : y = a_i x + b_i, c_i \leq x \leq d_i\}$ for some real numbers a_i, b_i, c_i, d_i. We say that G is a *simplex K segment graph* if the real numbers $a_i, b_i, c_i, d_i,\ i = 1, 2, \ldots n$ are under the constraints

$$a_i, b_i, c_i, d_i \geq 0, \text{ for every } i = 1, 2, \ldots n, \text{ and}$$
$$\sum_{i=1}^{n}(a_i + b_i + c_i + d_i) = K, \quad \text{where } K > 0 \text{ is a given constant.}$$

We let SIM-K-SEG denote the class of all simplex K segment graphs with parameter $K > 0$.

The problem ϵ-RECOG(SIM-K-SEG) is defined as follows. Given an abstract undirected graph G, does it belong with tolerance ϵ to SIM-K-SEG?

Formulation of ϵ-RECOG(SIM-K-SEG).. We first give a description for the problem with $\epsilon = 0$ and then we generalize for arbitrary $\epsilon \geq 0$. The formulation is taken from [39].

Letting l_i be the line containing s_i, we note that $s_i \cap s_j \neq \emptyset$ if l_i and l_j intersect in a single point whose x-coordinate lies in both the intervals $[c_i, d_i]$ and $[c_j, d_j]$. It is easy to see that the x-coordinate equals $\frac{b_j - b_i}{a_i - a_j}$.

Now we turn to the general case where $\epsilon \geq 0$. Let us introduce variables A_i, B_i, C_i, D_i representing the unknown quantities $a_i, b_i, c_i, d_i,\ i = 1, 2, \ldots, n$. By the problem's definition we require the vector $(A_1, B_1, C_1, D_1, \ldots, A_n, B_n, C_n, D_n)$ to be in the $(4n-1)$-simplex with parameter K. Then $s_i \cap s_j \neq \emptyset$

can be expressed by the following predicate:

$$
\begin{aligned}
\mathrm{INTS}(A_i, B_i, C_i, D_i, &A_j, B_j, C_j, D_j) = \\
&(A_i >_\epsilon A_j \wedge C_i(A_i - A_j) \le_\epsilon B_j - B_i \le_\epsilon D_i(A_i - A_j) \\
&\wedge C_j(A_i - A_j) \le_\epsilon B_j - B_i \le_\epsilon D_j(A_i - A_j)) \\
&\vee(A_i <_\epsilon A_j \wedge C_i(A_i - A_j) \ge_\epsilon B_j - B_i \ge_\epsilon D_i(A_i - A_j) \\
&\wedge C_j(A_i - A_j) \ge_\epsilon B_j - B_i \ge_\epsilon D_j(A_i - A_j))
\end{aligned}
$$

(this is only correct if we "globally" assume that $C_i \le_\epsilon D_i$ for all i). The existence of a SEG-representation of G can then be expressed by the formula

$$
\begin{aligned}
(\exists A_1 B_1 C_1 D_1 \dots A_n B_n C_n D_n K) &\left(\bigwedge_{i=1}^{n} C_i \le_\epsilon D_i \right) \\
\wedge &\left(\bigwedge_{\{i,j\} \in E} \mathrm{INTS}(A_i, B_i, C_i, D_i, A_j, B_j, C_j, D_j) \right) \\
\wedge &\left(\bigwedge_{\{i,j\} \notin E} \neg\mathrm{INTS}(A_i, B_i, C_i, D_i, A_j, B_j, C_j, D_j) \right).
\end{aligned}
$$

Theorem 21. *There is an algorithm that runs in time $n^{O(K^2 \cdot \log n/\epsilon^2)}$ and either finds a vector $(A_1, B_1, C_1, D_1, \dots, A_n, B_n, C_n, D_n)$ that is a solution to ϵ-RECOG(SIM-K-SEG), or determines that there is no solution to* 0*-RECOG(SIM-K-SEG).*

Proof. We set $x = (A_1, B_1, C_1, D_1, \dots, A_n, B_n, C_n, D_n)$ and $F(x)$ to be the above formula that we constructed. Their combination makes an ϵ-ETR instance. Vector x is constrained over the convex hull defined by the vertices of the $(4n-1)$-simplex, i.e. vectors $v_i \in \mathbb{R}^{4n}$, $i \in \{1, 2, \dots 4n\}$ with their i-th element equal to K and the rest equal to 0. Therefore the cardinality of our convex set is $m = 4n$, and $\gamma = K$. By looking at the formula we can conclude that $a = 1$, $t = 4$, and $d = 2$. By Theorem 9 the result follows.

5.6.2. Unit disk intersection graphs

Definitions.. Let G be an undirected graph with vertex set $\{v_1, v_2, \dots, v_n\}$. We say that G is a *unit disk intersection graph* or *unit disk graph* if there are disks $d_1, d_2, \dots, d_n$ (in the plane) with radius 1 such that, for every $i, j, 1 \leq i < j \leq n$, the disks d_i and d_j intersect at more than one points (i.e., their perimeters have two points in common) if and only if $\{v_i, v_j\} \in E(G)$.

The disk d_i representing vertex v_i can be algebraically described as the set $\{(x, y) \in \mathbb{R}^2 : (x - x_i)^2 + (y - y_i)^2 \leq 1\}$ for some real numbers x_i, y_i that determine the centre of the disk. We say that G is a *simplex K unit disk graph* if the real numbers x_i, y_i, $i = 1, 2, \dots n$ are under the constraints

$$x_i, y_i \geq 0, \text{ for every } i = 1, 2, \dots n, \text{ and}$$
$$\sum_{i=1}^{n} (x_i + y_i) = K, \quad \text{where } K > 0 \text{ is a given constant.}$$

We let SIM-K-UDG denote the class of all simplex K unit disk graphs with parameter $K > 0$.

The problem ϵ-RECOG(SIM-K-UDG) is defined as follows. Given an abstract undirected graph G, does it belong with tolerance ϵ to SIM-K-UDG?

Formulation of ϵ-RECOG(SIM-K-UDG).. Let us introduce variables X_i, Y_i representing the unknown quantities x_i, y_i, $i = 1, 2, \dots, n$. We require the vector $(X_1, Y_1, \dots, X_n, Y_n)$ to be in the $(2n-1)$-simplex with parameter K. Then we consider an ϵ-intersection $d_i \cap_\epsilon d_j \neq \emptyset$ to happen if:

$$\sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} < 2 + \epsilon,$$

and an ϵ-non-intersection $d_i \cap_\epsilon d_j = \emptyset$ to happen if:

$$\sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} \geq 2 - \epsilon.$$

The existence of a UDG-representation of G can then be expressed by the formula

$$\begin{aligned}
&(\exists X_1 Y_1 \dots X_n Y_n) \\
&\quad \left(\bigwedge_{\{i,j\} \in E} (X_i - X_j) \cdot (X_i - X_j) + (Y_i - Y_j) \cdot (Y_i - Y_j) < 4 + 2\epsilon + \epsilon^2 \right) \\
&\quad \wedge \left(\bigwedge_{\{i,j\} \notin E} (X_i - X_j) \cdot (X_i - X_j) + (Y_i - Y_j) \cdot (Y_i - Y_j) \geq 4 - 2\epsilon + \epsilon^2 \right).
\end{aligned}$$

Theorem 22. *There is an algorithm that runs in time $n^{O(K^2 \cdot \log n / \epsilon^2)}$ and either finds a vector $(X_1, Y_1, \dots, X_n, Y_n)$ that is a solution to ϵ-RECOG(SIM-K-UDG), or determines that there is no solution to 0-RECOG(SIM-K-UDG).*

PROOF. We set $x = (X_1, Y_1, \ldots, X_n, Y_n)$ and $F(x)$ to be the above formula that we constructed. Their combination makes an ϵ-ETR instance. Vector x is constrained over the convex set defined by the vertices of the $(2n-1)$-simplex, i.e. vectors $v_i \in \mathbb{R}^{2n}$, $i \in \{1, 2, \ldots 2n\}$ with their i-th element equal to K and the rest equal to 0. Therefore the cardinality of our convex set is $m = 2n$, and $\gamma = K$. By looking at the formula we can conclude that $a = 2$, $t = 7$, and $d = 2$. By Theorem 9 the result follows.

6. Discussion and Open Problems

It seems that ETR is a class which captures decision problems that are a lot harder than these in NP (under standard complexity assumptions) because either they do not have truth certificates of polynomial length or because the certificate cannot be checked in polynomial time. One can think of ETR and thus Function ETR (FETR) and Total Function ETR (TFETR) as being the analogues of NP, FNP and TFNP respectively in the Blum-Shub-Smale (BSS) model of computation [18], in which computing functions over real numbers is as costly as is computing functions over rational numbers in Turing machines.

In this paper we provide a general framework for approximation schemes, a framework designed for problems in a subclass of ETR (or more precisely, FETR). In particular, since some function problems in TFNP or, in general, FNP (whose corresponding decision problems are in NP), have polynomial or quasi-polynomial time approximation schemes (PTAS/QPTAS), we study harder problems in TFETR or FETR, and seek similar approximation schemes. In a beautiful turn of events, we show that PTASs and QPTASs exist for a wide class of problems in FETR. By extending the well-known Lipton-Markakis-Mehta (LMM) technique that yields the best possible algorithm (under standard complexity assumptions) for computing approximate Nash equilibria in bimatrix games, we provide a general framework that gives in a standardized way, approximation algorithms of the same quality as the state of the art for some problems, while for some other problems these algorithms are the first to achieve an efficient approximation. Interestingly, approximation techniques that work inside FNP, transcend it, and reach FETR.

For a given constrained ϵ-ETR instance whose variables' domain is the convex hull of ℓ vectors, we presented an algorithm which runs in time $\min\{\ell^{O(kn)}, k^{O(\ell n)}\}$, for k indicated in Theorem 5, that either computes a solution or responds that a solution to the exact instance does not exist. This algorithm is a QPTAS or PTAS for many well-known problems. However, our algorithm, being an extension of the LMM algorithm, for some problems does not have better running time than the state of the art algorithms that are tailored to these problems. The most important open problem is to make the quantity k depend logarithmically on crucial parameters, such as the number of variables n and the degree of the polynomials d, instead of polynomially. This would generalize many algorithms, such as the PTAS for computing an ϵ-Nash equilibrium in anonymous games [40] and the best algorithm for computing an ϵ-Nash equilibrium in general multi-player normal form games [4].

Acknowledgements

P. Spirakis wishes to dedicate this paper to the memory of his late father in law Mathematician and Professor Dimitrios Chrysofakis, who was among the first in Greece to work on tensor analysis. This work was completed while A. Deligkas was affiliated with the Department of Computer Science and the Leverhulme Research Centre for Functional Materials Design at the University of Liverpool, Liverpool, UK. The author was supported by Leverhulme Trust and the Leverhulme Research Centre for Functional Materials Design. J. Fearnley was supported by the EPSRC grant EP/P020909/1. This work was completed while T. Melissourgos was affiliated with the Department of Computer Science at the University of Liverpool, Liverpool, UK. P. Spirakis was supported by the EPSRC Grant EP/P02002X/1, the Leverhulme Research Centre, and the ERC Project ALGAME.

[1] A. Deligkas, J. Fearnley, T. Melissourgos, P. G. Spirakis, Approximating the existential theory of the reals, in: G. Christodoulou, T. Harks (Eds.), Web and Internet Economics, Springer International Publishing, Cham, 2018, pp. 126–139.

[2] R. J. Lipton, E. Markakis, A. Mehta, Playing large games using simple strategies, in: Proc. of EC, ACM, 2003, pp. 36–41.

[3] I. Althöfer, On sparse approximations to randomized strategies and convex combinations, Linear Algebra and its Applications 199 (1994) 339 – 355, special Issue Honoring Ingram Olkin. doi:https://doi.org/10.1016/0024-3795(94)90357-3.
URL http://www.sciencedirect.com/science/article/pii/0024379594903573

[4] Y. Babichenko, S. Barman, R. Peretz, Empirical distribution of equilibrium play and its testing application, Mathematics of Operations Research 42 (1) (2016) 15–29.

[5] S. Barman, Approximating Nash equilibria and dense bipartite subgraphs via an approximate version of Caratheodory's theorem, in: Proc. of STOC, 2015, pp. 361–369.

[6] S. Barman, K. Ligett, G. Piliouras, Approximating Nash equilibria in tree polymatrix games, in: Proc. of SAGT, Springer, 2015, pp. 285–296.

[7] A. Deligkas, J. Fearnley, P. Spirakis, Lipschitz continuity and approximate equilibria, in: Proc. of SAGT, Springer, 2016, pp. 15–26.

[8] A. Deligkas, J. Fearnley, R. Savani, Computing constrained approximate equilibria in polymatrix games, in: Proc. of SAGT, 2017, pp. 93–105.

[9] J. Canny, Some algebraic and geometric computations in PSPACE, in: Proc. of STOC, ACM, New York, NY, USA, 1988, pp. 460–467. doi:10.1145/62212.62257.
URL http://doi.acm.org/10.1145/62212.62257

[10] V. Bilò, M. Mavronicolas, The complexity of decision problems about Nash equilibria in win-lose games, in: Proc. of SAGT, 2012, pp. 37–48.

[11] V. Bilò, M. Mavronicolas, Complexity of rational and irrational Nash equilibria, Theory of Computing Systems 54 (3) (2014) 491–527.

[12] V. Bilò, M. Mavronicolas, A catalog of EXISTS-R-complete decision problems about Nash equilibria in multi-player games, in: 33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016, February 17-20, 2016, Orléans, France, 2016, pp. 17:1–17:13. doi:10.4230/LIPIcs.STACS.2016.17.
URL https://doi.org/10.4230/LIPIcs.STACS.2016.17

[13] V. Bilò, M. Mavronicolas, Existential-R-complete decision problems about symmetric Nash equilibria in symmetric multi-player games, in: 34th Symposium on Theoretical Aspects of Computer Science, STACS 2017, March 8-11, 2017, Hannover, Germany, 2017, pp. 13:1–13:14. doi:10.4230/LIPIcs.STACS.2017.13.
URL https://doi.org/10.4230/LIPIcs.STACS.2017.13

[14] J. Garg, R. Mehta, V. V. Vazirani, S. Yazdanbod, ETR-completeness for decision versions of multi-player (symmetric) Nash equilibria, ACM Transactions on Economics and Computation (TEAC) 6 (1) (2018) 1.

[15] M. Schaefer, D. Stefankovic, Fixed points, Nash equilibria, and the existential theory of the reals, Theory Comput. Syst. 60 (2) (2017) 172–193. doi:10.1007/s00224-015-9662-0.
URL https://doi.org/10.1007/s00224-015-9662-0

[16] M. Braverman, Y. Kun-Ko, O. Weinstein, Approximating the best Nash equilibrium in $n^{o(\log n)}$-time breaks the exponential time hypothesis, in: Proc. of SODA, 2015, pp. 970–982.

[17] A. Deligkas, J. Fearnley, R. Savani, Inapproximability results for constrained approximate Nash equilibria, Information and Computation (2018). doi:https://doi.org/10.1016/j.ic.2018.06.001.
URL http://www.sciencedirect.com/science/article/pii/S0890540118301123

[18] L. Blum, M. Shub, S. Smale, On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines, Bulletin of the American Mathematical Society (N.S.) 21 (1) (1989) 1–46.
URL https://projecteuclid.org:443/euclid.bams/1183555121

[19] E. De Klerk, M. Laurent, P. A. Parrilo, A PTAS for the minimization of polynomials of fixed degree over the simplex, Theoretical Computer Science 361 (2-3) (2006) 210–225.

[20] T. S. Motzkin, E. G. Straus, Maxima for graphs and a new proof of a theorem of Turán, Canadian Journal of Mathematics 17 (1965) 533–540.
URL http://math.ca/10.4153/CJM-1965-053-6#

[21] R. M. Karp, Reducibility among combinatorial problems, in: Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA, 1972, pp. 85–103. doi:10.1007/978-1-4684-2001-2_9.
URL https://doi.org/10.1007/978-1-4684-2001-2_9

[22] I. M. Bomze, E. De Klerk, Solving standard quadratic optimization problems via linear, semidefinite and copositive programming, Journal of Global Optimization 24 (2) (2002) 163–185.

[23] W. Hoeffding, Probability inequalities for sums of bounded random variables, Journal of the American Statistical Association 58 (301) (1963) 13–30.
URL `http://www.jstor.org/stable/2282952`

[24] E. De Klerk, The complexity of optimizing over a simplex, hypercube or sphere: a short survey, Central European Journal of Operations Research 16 (2) (2008) 111–125.

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

[26] V. Conitzer, T. Sandholm, New complexity results about Nash equilibria, Games and Economic Behavior 63 (2) (2008) 621 – 641.

[27] P. Austrin, M. Braverman, E. Chlamtac, Inapproximability of NP-complete variants of Nash equilibrium, Theory of Computing 9 (2013) 117–142.

[28] E. Hazan, R. Krauthgamer, How hard is it to approximate the best Nash equilibrium?, SIAM J. Comput. 40 (1) (2011) 79–91.

[29] L. S. Shapley, Stochastic games, Proceedings of the national academy of sciences 39 (10) (1953) 1095–1100.

[30] K. A. Hansen, M. Koucký, N. Lauritzen, P. B. Miltersen, E. P. Tsigaridas, Exact algorithms for solving stochastic games: extended abstract, in: Proc. of STOC, 2011, pp. 205–214.

[31] F. W. Simmons, F. E. Su, Consensus-halving via theorems of Borsuk-Ulam and Tucker, Mathematical social sciences 45 (1) (2003) 15–25. `doi:10.1016/S0165-4896(02)00087-2`.

[32] A. Deligkas, J. Fearnley, T. Melissourgos, P. G. Spirakis, Computing exact solutions of consensus halving and the Borsuk-Ulam theorem, in: 46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece., 2019, pp. 138:1–138:14.
`doi:10.4230/LIPIcs.ICALP.2019.138`.
URL `https://doi.org/10.4230/LIPIcs.ICALP.2019.138`

[33] A. Deligkas, J. Fearnley, T. Melissourgos, P. G. Spirakis, Computing exact solutions of consensus halving and the Borsuk-Ulam theorem, J. Comput. Syst. Sci. 117 (2021) 75–98. doi:10.1016/j.jcss.2020.10.006.
URL https://doi.org/10.1016/j.jcss.2020.10.006

[34] E. De Klerk, M. Laurent, P. A. Parrilo, A PTAS for the minimization of polynomials of fixed degree over the simplex, Theoretical Computer Science 361 (2-3) (2006) 210–225.

[35] E. de Klerk, M. Laurent, Z. Sun, An alternative proof of a PTAS for fixed-degree polynomial optimization over the simplex, Mathematical Programming 151 (2) (2015) 433–457.

[36] L. Faybusovich, Global optimization of homogeneous polynomials on the simplex and on the sphere, in: Frontiers in global optimization, Springer, 2004, pp. 109–121.

[37] C. J. Hillar, L.-H. Lim, Most tensor problems are NP-hard, Journal of the ACM (JACM) 60 (6) (2013) 45.

[38] J. Cardinal, Computational geometry column 62, ACM SIGACT News 46 (4) (2015) 69–78.

[39] J. Matoušek, Intersection graphs of segments and $\exists\mathbb{R}$, CoRR abs/1406.2636 (2014). arXiv:1406.2636.
URL http://arxiv.org/abs/1406.2636

[40] C. Daskalakis, C. H. Papadimitriou, On oblivious PTAS's for Nash equilibrium, in: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009, 2009, pp. 75–84. doi:10.1145/1536414.1536427.
URL https://doi.org/10.1145/1536414.1536427