On the complexity of Sperner's Lemma

Katalin Friedl* Gábor Ivanyos† Miklos Santha ‡ Yves F. Verhoeven §

Abstract

We present several results on the complexity of various forms of Sperner's Lemma. In the black-box model of computing, we exhibit a deterministic algorithm for Sperner problems over pseudo-manifolds of arbitrary dimension. The query complexity of our algorithm is essentially linear in the separation number of the skeleton graph of the manifold and the size of its boundary. As a corollary we get an $O(\sqrt{n})$ deterministic query algorithm for the black-box version of the problem **2D-SPERNER**, a well studied member of Papadimitriou's complexity class PPAD. This upper bound matches the $\Omega(\sqrt{n})$ deterministic lower bound of Crescenzi and Silvestri. In another black-box result we prove for the same problem an $\Omega(\sqrt[4]{n})$ lower bound for its probabilistic, and an $\Omega(\sqrt[8]{n})$ lower bound for its quantum query complexity, showing that all these measures are polynomially related. Finally we explicit Sperner problems on a 2-dimensional pseudo-manifold and prove that they are complete respectively for the classes PPAD, PPADS and PPA. This is the first time that a 2-dimensional Sperner problem is proved to be complete for any of the polynomial parity argument classes.

1 Introduction

Papadimitriou defined in [23, 24] the complexity classes PPA, PPAD, and PSK in order to classify total search problems which have always a solution.The class PSK was renamed PPADS in [6]. These classes can be characterized by some underlying combinatorial principles. The class Polynomial Parity Argument (PPA) is the class of NP search problems, where the existence of the solution is guaranteed by the fact that in every finite graph the number of vertices with odd degree is even. The class PPAD is the directed version of PPA, and its basic search problem is the following: in a directed graph, given a source, find another source or a sink. In the class PPADS the basic search problem is more restricted than in PPAD: in a directed graph, given a source, find a sink.

These classes are in fact subfamilies of TFNP, the family of all total NP-search problems, introduced by Megiddo and Papadimitriou [20]. Other important subclasses of TFNP are Polynomial Pigeonhole Principle

*BUTE, H-1521 Budapest, P.O.Box 91., Hungary. Email: `friedl@cs.bme.hu`.The research was supported by the EU 5th framework program RESQ IST-2001-37559 and Centre of Excellence ICAI-CT-2000-70025, and OTKA grants T42559 and T46234.

†MTA SZTAKI, H-1518 Budapest, P.O. Box 63., Hungary. Email: `Gabor.Ivanyos@sztaki.hu`.The research was supported by the EU 5th framework program RESQ IST-2001-37559 and Centre of Excellence ICAI-CT-2000-70025, and OTKA grants T42706 and T42481.

‡CNRS–LRI, UMR 8623. Université Paris XI, 91405 Orsay, France. Email: `santha@lri.fr`.The research was supported by the EU 5th framework program RESQ IST-2001-37559, and by the ACI CR 2002-40 and ACI SI 2003-24 grants of the French Research Ministry.

§Univ. Paris XI, 91405 Orsay and ENST,75013 Paris, France. Email: `yves.verhoeven@normalesup.org`.The research was supported by the EU 5th framework program RESQ IST-2001-37559, and by the ACI CR 2002-40 and ACI SI 2003-24 grants of the French Research Ministry.

(PPP) and Polynomial Local Search (PLS). The elements of PPP are problems which by their combinatorial nature obey the pigeonhole principle and therefore have a solution. In a PLS problem, one is looking for a local optimum for a particular objective function, in some neighborhood structure. All these classes are interesting because they contain search problems not known to be solvable in polynomial time, but which are also somewhat easy in the sense that they can not be NP-hard unless NP = co-NP.

Another point that makes the parity argument classes interesting is that there are several natural problems from different branches of mathematics that belong to them. For example, in a graph with odd degrees, when a Hamiltonian path is given, a theorem of Smith [31] ensures that there is another Hamiltonian path. It turns out that finding this second path belongs to the class PPA [24]. A search problem coming from a modulo 2 version of Chevalley's theorem [24] from number theory is also in PPA. Complete problems in PPAD are the search versions of Brouwer's fixed point theorem, Kakutani's fixed point theorem, Borsuk-Ulam theorem, and Nash equilibrium (see [24]).

The classical Sperner's Lemma [28] states that in a triangle with a regular triangulation whose vertices are labelled with three colors, there is always a trichromatic triangle. This lemma is of special interest since some customary proofs for the above topological fixed point theorems rely on its combinatorial content. However, it is unknown whether the corresponding search problem, that Papadimitriou [24] calls **2D-SPERNER**, is complete in PPAD. Variants of Sperner's Lemma also give rise to other problems in the parity argument classes. Papadimitriou [24] has proven that a 3-dimensional analogue of **2D-SPERNER** is in fact complete in PPAD. In [15], Grigni described a non-oriented version of 3-dimensional Sperner's Lemma that is complete for the class PPA.

The study of query complexities of the black-box versions of several problems in TFNP is an active field of research. Several recent results point into the direction that quantum algorithms can give only a limited speedup over deterministic ones in this framework. The collision lower bound of Aaronson [1] and Shi [26] about PPP, and the recent result of Santha and Szegedy [25] on PLS imply that the respective deterministic and quantum complexities are polynomially related. As a consequence, if an efficient quantum algorithm exists for a problem in these classes, it must exploit its specific structure. In a related issue, Buresh-Oppenheim and Morioka [9] have obtained relative separation results among PLS and the polynomial parity argument classes.

2 Results

A *black-box problem* is a relation $R \subseteq S \times T$ where T is a finite set and $S \subseteq \Sigma^n$ for some finite set Σ. The oracle input is a function $x \in S$, hidden by a black-box, such that x_i, for $i \in \{1, \ldots, n\}$ can be accessed via a query parameterized by i. The output of the problem is some $y \in T$ such that $(x, y) \in R$. A special case is the *functional oracle problem* when the relation is given by a function $A : S \to T$, the (unique) output is then $A(x)$. We say that A is *total* if $S = \Sigma^n$.

In the query model of computation each query adds one to the complexity of the algorithm, but all other computations are free. The state of the computation is represented by three registers, the query register $i \in \{1, \ldots, n\}$, the answer register $a \in \Sigma$, and the work register z. The computation takes place in the vector space spanned by all basis states $|i\rangle|a\rangle|z\rangle$. In the *quantum query model* introduced by Beals, Buhrman, Cleve, Mosca and de Wolf [5] the state of the computation is a complex combination of all basis states which has unit length in the norm l_2. In the randomized model it is a non-negative real combination of unit length in the norm l_1, and in the deterministic model it is always one of the basis states.

The query operation O_x maps the basis state $|i\rangle|a\rangle|z\rangle$ into the state $|i\rangle|(a + x_i) \bmod |\Sigma|\rangle|z\rangle$ (here we identify Σ with the residue classes $\bmod|\Sigma|$). Non-query operations are independent of x. A *k-query algo-*

rithm is a sequence of $(k+1)$ operations $(U_0, U_1, \ldots, U_k)$ where U_i is unitary in the quantum and stochastic in the randomized model, and it is a permutation in the deterministic case. Initially the state of the computation is set to some fixed value $|0\rangle|0\rangle|0\rangle$, and then the sequence of operations $U_0, O_x, U_1, O_x, \ldots, U_{k-1}, O_x, U_k$ is applied. A quantum or randomized algorithm computes (with two-sided error) R if the observation of the appropriate last bits of the work register yield some $y \in T$ such that $(x, y) \in R$ with probability at least $2/3$. Then $\mathsf{QQC}(R)$ (resp. $\mathsf{RQC}(R)$) is the smallest k for which there exists a k-query quantum (resp. randomized) algorithm which computes R. In the case of deterministic algorithms of course exact computation is required, and the deterministic query complexity $\mathsf{DQC}(R)$ is defined then analogously. We have $\mathsf{DQC}(R) \geq \mathsf{RQC}(R) \geq \mathsf{QQC}(R)$.

Beals et al. [5] have shown that in the case of total functional oracle problems the deterministic and quantum complexities are polynomially related, and the gap is at most a degree 6 polynomial. No such relation is known for relations or for partial functional problems, in fact for several partial functional problems exponential quantum speedups are known [11, 27].

An *NP-search problem* is specified by a polynomial time relation $\mathcal{R}(x, y)$, such that for some polynomial $p(n)$, for every x and y such that $\mathcal{R}(x, y)$, we have $|y| \leq p(|x|)$. Given an input x to the problem, the task is to find a y such that $\mathcal{R}(x, y)$ if there is one, and else report failure. We call an NP-search problem *total* if for every x there exists a solution y. The class of total NP-search problems is called TFNP by Megiddo and Papadimitriou [20].

For two problems $\mathcal{R}_1$, $\mathcal{R}_2$ in TFNP, we say that $\mathcal{R}_1$ is *reducible to* $\mathcal{R}_2$ if there exist two functions f and g computable in polynomial time such that $f(x)$ is a legal input to $\mathcal{R}_2$ whenever x is an input to $\mathcal{R}_1$, and $\mathcal{R}_2(f(x), y)$ implies $\mathcal{R}_1(x, g(x, y))$.

The parity argument classes are defined via concrete problems, by closure under reduction. The **LEAF** problem is defined as follows. The input is a pair $(M, 0^k)$ where M is the description of a polynomial time Turing machine that on every input outputs a set of size at most 2, and k is a positive integer. Moreover, M is such that $M(0^k) = \{1^k\}$, and $0^k \in M(1^k)$. Such an input specifies an undirected graph $G_k = (V, E)$, where $V = \{0, 1\}^k$, and $\{u, v\}$ is in E if $u \in M(v)$, and $v \in M(u)$. The output of the problem is a leaf of G_k different from 0^k. The class PPA is the set of total search problems reducible to **LEAF**. In the search problems defining the classes PPADS and PPAD, the Turing machine defines a directed graph, where 0^k is always a source. The output in the case of PPADS is a sink, and in the case of PPAD a sink or source different from 0^k.

In this paper, we will give several results about various Sperner problems, both in the black-box and the NP-search framework. In Section 5, we will prove that the deterministic query complexity of **REGULAR 2-SPM**, the black-box version of **2D-SPERNER** is $O(\sqrt{n})$. This matches the deterministic $\Omega(\sqrt{n})$ lower bound of Crescenzi and Silvestri [10]. In fact, this result is the corollary of a general algorithm which we present in this section and that solves the Sperner problems over pseudo-manifolds of arbitrary dimension. The complexity analysis of the algorithm will be expressed in **Theorem 4** in two combinatorial parameters of the pseudo-manifold: the size of its boundary and the separation number of its skeleton graph. In Section 6, we show that quantum, probabilistic, and deterministic query complexities of **REGULAR 2-SPM** are polynomially related. More precisely, in **Theorem 8** we will prove that its randomized complexity is $\Omega(\sqrt[4]{n})$ and that its quantum complexity is $\Omega(\sqrt[8]{n})$. This result is analogous to the polynomial relations obtained for the respective query complexities of PPP and PLS. Finally, in Section 7, in **Theorem 9**, we show that there exists a Sperner problem on a 2-dimensional pseudo-manifold which is complete for the class PPAD. This is the first time that a 2-dimensional Sperner problem is proved to be complete for a parity argument class. We can generalize this completeness result for analogous problems in the classes PPADS and PPA.

3 Mathematical background

For an undirected graph $G = (V, E)$, and for a subset $V' \subseteq V$ of the vertices, we denote by $G[V']$ the induced subgraph of G by V'. A graph $G'' = (V'', E'')$ is a subgraph of G, in notation $G'' \subseteq G$, if $V'' \subseteq V$ and $E'' \subseteq E$. The ring $\mathbb{Z}/(2)$ denotes the ring with 2 elements.

3.1 Simplicial complexes

Definition 1 (Simplicial complex). A *simplicial complex* K is a non-empty collection of subsets of a finite set U, such that whenever $S \in K$ then $S' \in K$ for every $S' \subseteq S$. An element S of K of cardinality $d + 1$ is called a d*-simplex*. A d'-simplex $S' \subseteq S$ is called a d'*-face of* S. We denote by K_d the set of d-simplices of K. An *elementary* d*-complex* is a simplicial complex that contains exactly one d-simplex and its subsets. The *dimension* of K, denoted by $\dim(K)$, is the largest d such that K contains a d-simplex.

The elements of K_0 are called the *vertices* of K, and the elements of K_1 are called the *edges* of K. The *skeleton graph* $G_K = (V_K, E_K)$ is the graph whose vertices are the vertices of K, and the edges are the edges of K.

Fact 1. *Let d be a positive integer. If S is an elementary d-complex, then G_S is the complete graph.*

Without loss of generality, we suppose that U consists of integers, and we identify $\{u\}$ with u, for $u \in U$. A geometrical realization $\tilde{K}$ of K can be constructed as follows: let $b_1, \ldots, b_{|U|} \in \mathbb{R}^{|U|}$ be linearly independent vectors. For a simplex $S \in K$ the set $\tilde{S}$ is the convex hull of $\{b_u \mid u \in S\}$. The geometrical realization $\tilde{K}$ is $\bigcup_{S \in K} \tilde{S}$.

Definition 2 (Oriented Simplex). For every positive integer n, we define an equivalence relation $\equiv_n$ over $\mathbb{Z}^n$, by $a \equiv_n b$ if there exists an even permutation σ such that $\sigma \cdot a = b$. For every $a \in \mathbb{Z}^n$ we denote by $[a]_{\equiv_n}$ the equivalence class of a for $\equiv_n$. The two equivalence classes of the orderings of the 0-faces of a simplex are called its *orientations*. An *oriented simplex* is a pair formed of a simplex and one of its orientations.

For an oriented d-simplex $(S, [\tau]_{\equiv_{d+1}})$, where τ is an ordering of the 0-faces of S, and a permutation σ over $\{1, \ldots, d + 1\}$, we denote by $\sigma \cdot (S, [\tau]_{\equiv_{d+1}})$ the oriented d-simplex $(S, [\sigma \cdot \tau]_{\equiv_{d+1}})$. For every integer d, and every simplicial complex K whose simplices have been oriented, we denote by K_d the set of oriented d-simplices of K. From now on, S may denote an oriented or a non-oriented simplex. When S is an oriented simplex, $\bar{S}$ will denote the same simplex with the opposite orientation. We also define $S^{(i)}$ to be S if i is even, and to be $\bar{S}$ if i is odd. We will often specify an oriented simplex by an ordering of its 0-faces.

Definition 3. Let $S = (v_0, \ldots, v_d)$ be an oriented d-simplex. For every $0 \leq i \leq d$, for every $(d-1)$-face $\{v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_d\}$ of S, the *induced orientation* is $(v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_d)^{(i)}$.

Definition 4. Let K be a simplicial complex whose simplices have been oriented, and let R be a ring. We define $C_d(K; R)$ as the submodule of the free R-module over the d-simplices of K with both possible orientations, whose elements are of the form $\sum_{S \in K_d} (c_S \cdot S + c_{\bar{S}} \cdot \bar{S})$, with $c_S \in R$, satisfying the relation $c_S = -c_{\bar{S}}$. The elements of $C_d(K; R)$ are called d*-chains*. For every oriented simplex S of K, we denote by $\langle S \rangle$ the element $S - \bar{S}$ of $C_d(K; R)$.

Let S be an oriented d-simplex $(v_0, v_1, \ldots v_d)$ of K. The *algebraic boundary* of $\langle S \rangle$, denoted by $\partial_d \langle S \rangle$, is the $(d-1)$-chain of $C_{d-1}(K; R)$

$$\partial_d \langle S \rangle = \sum_{i=0}^{d} (-1)^i \langle (v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_d) \rangle.$$

Since $\partial_d \langle S \rangle = -\partial_d \langle \bar{S} \rangle$, the operator ∂_d has been correctly defined on a basis of $C_d(K;R)$ and can therefore be uniquely extended into a homomorphism $\partial_d : C_d(K;R) \to C_{d-1}(K;R)$. The next Lemma follows immediately from the definition of ∂_d.

Lemma 1. *Let S be an oriented d-simplex of a simplicial complex K. Denote by F_S the set of $(d-1)$-faces of S, and for every $S' \in F_S$ by $\tau^S_{S'}$ the induced orientation on S'. Then $\partial_d \langle S \rangle = \sum_{S' \in F_S} \langle (S', \tau^S_{S'}) \rangle$.*

Following an early version of a paper of Bloch [8], in the next definition we generalize the notion of pseudo-manifold, without the usual requirements of connectivity and pure dimensionality.

Definition 5. A simplicial complex $\mathcal{M}$ is a *pseudo d-manifold*, for a positive integer d, if

(i) $\mathcal{M}$ is a union of elementary d-complexes,

(ii) every $(d-1)$-simplex in $\mathcal{M}$ is a $(d-1)$-face of at most two d-simplices of $\mathcal{M}$.

The *boundary of* $\mathcal{M}$ is the set of elementary $(d-1)$-complexes in $\mathcal{M}$ that belong exactly to one d-simplex of $\mathcal{M}$. We denote it by $\partial\, \mathcal{M}$.

A pseudo d-manifold $\mathcal{M}$ is said to be *orientable* if it is possible to assign an orientation to each d-simplex of $\mathcal{M}$, such that for all $(d-1)$-simplex of $\mathcal{M}$ that is not on its boundary the orientations induced by the two d-simplices to which it belongs are opposite. Such a choice of orientations for all the d-simplices of $\mathcal{M}$ makes $\mathcal{M}$ *oriented*.

If the d-simplices of $\mathcal{M}$ are oriented, then there is a natural orientation of the $(d-1)$-simplices of $\partial\, \mathcal{M}$, where each $(d-1)$-simplex has the orientation induced by the oriented d-simplex of which it is a $(d-1)$-face. Notice that if $\mathcal{M}$ is a pseudo d-manifold, then $\partial\, M$ need not be a pseudo $(d-1)$-manifold.

From now, all the simplicial complexes will be pseudo-manifolds. Observe that if $R = \mathbb{Z}/(2)$, then for any oriented d-simplex S, we have $\langle S \rangle = \langle \bar{S} \rangle$.

Definition 6. Given a simplicial complex K of dimension d, the *standard d-chain $\widehat{K}$ of K* will be defined depending on whether K is oriented as follows:

- if K is non-oriented, then $\widehat{K} = \sum_{S \in K_d} \langle (S, \tau_S) \rangle \in C_d(K, \mathbb{Z}/(2))$, for an arbitrary choice of orientations τ_S of the d-simplices S in K,
- if K is oriented, then $\widehat{K} = \sum_{S \in K_d} \langle (S, \tau_S) \rangle \in C_d(K, \mathbb{Z})$ where τ_S is the orientation of S in K.

Fact 2. *Let d be an integer, and let $\mathcal{M}$ be a pseudo d-manifold. Then,*

1. if $\mathcal{M}$ is not oriented, then the equality $\widehat{\partial\, \mathcal{M}} = \partial_d \widehat{\mathcal{M}}$ holds in $C_{d-1}(\partial\, \mathcal{M}, \mathbb{Z}/(2))$,

2. and if $\mathcal{M}$ is oriented, then the equality $\widehat{\partial\, \mathcal{M}} = \partial_d \widehat{\mathcal{M}}$ holds in $C_{d-1}(\partial\, \mathcal{M}, \mathbb{Z})$.

Proof. For every d-simplex S in $\mathcal{M}$, denote by F_S the set of $(d-1)$-faces of S, and for every $S' \in F_S$ denote by $\tau^S_{S'}$ the induced orientation of S' from S. From Lemma 1, we can write

$$\partial_d \widehat{\mathcal{M}} = \sum_{S \in \mathcal{M}_d} \partial_d \langle (S, \tau_S) \rangle = \sum_{S \in \mathcal{M}_d} \sum_{S' \in F_S} \langle (S', \tau^S_{S'}) \rangle.$$

From the definition of a pseudo d-manifold, we know that in the last sum each $(d-1)$-simplex that is not on the boundary of $\mathcal{M}$ appears exactly twice. If $\mathcal{M}$ is not oriented, then as the base ring is $\mathbb{Z}/(2)$, the only $(d-1)$-simplices that remain in the sum are those that are in $\partial\, \mathcal{M}$. If $\mathcal{M}$ is oriented, then from the definition of the orientability, it follows that each $(d-1)$-simplex that appears in two d-simplices of $\mathcal{M}$ appears in the sum once with each orientation. As for any oriented simplex S the equality $\langle S \rangle + \langle \bar{S} \rangle = 0$ holds, the only terms that do not cancel are the oriented $(d-1)$-simplices of the boundary. These $(d-1)$-simplices appear with the correct orientation. $\square$

3.2 Triangulated surfaces

Definition 7 (Surface). A *surface* $\mathcal{S}$ is a pseudo 2-manifold whose skeleton graph is connected, and such that for every two 2-simplices T and T' that contain a vertex v, there exists a sequence $T = T_0, T_1, \ldots, T_k = T'$ of 2-simplices of $\mathcal{S}$ such that $T_i \cap T_{i+1}$ is a 1-simplex of $\mathcal{S}$ containing v, for $1 \leq i < k$. An *oriented surface* $\mathcal{S}$ is a surface equipped with an orientation in the sense of Definition 5.

Notice that our definition of surface coincides with the usual definition of triangulated surface.

Definition 8. Let $G = (V, E)$ be a graph. A *rotation system* for G is a set $\Pi = \{\pi_v \mid v \in V\}$ of permutations such that for every $v \in V$ the permutation π_v is a cyclic permutation of the neighbors of v in G.

Fact 3. *Let $\mathcal{S}$ be an oriented surface with empty boundary, and let v be a vertex of $\mathcal{S}$. For every vertex v' such that $\{v, v'\}$ is an edge, let v'' be the (unique) vertex in V such that (v', v, v'') is an oriented 2-simplex of $\mathcal{S}$. Set $\pi_v(v') = v''$. The map π_v is a cyclic permutation.*

Definition 9. Let G be the skeleton graph of an oriented surface $\mathcal{S}$ with empty boundary. The rotation system defined in Fact 3 is called the *rotation system of $\mathcal{S}$*.

Definition 10. Let $(G_n)_{n\in\mathbb{N}} = (V_n, E_n)_{n\in\mathbb{N}}$ be a family of undirected graphs where $|V_n| = n$, and $\Pi_n = \{\pi_v \mid v \in V_n\}$ be a rotation system for G_n. The rotation system Π_n is said to be *efficiently computable* if there exists a Turing machine M such that

(i) on input n and pair (v, v'), with $\{v, v'\} \in E_n$, computes the vertices v'' and v''' such that $\pi_v(v') = v''$ and $\pi_v^{-1}(v') = v'''$ using time polynomial in $\log n$,

(ii) on input n and triple (v, v', v''), with $\{v, v'\}$ and $\{v, v''\}$ in E_n, computes the smallest non-negative integer i such that $\pi_v^i(v') = v''$ using time polynomial in $\log n$. Later, we will refer to the integer i by $\log_{v'}^{\pi_v}(v'')$.

Lemma 2. *If m is a non zero integer that is equal to 7 modulo 12, then the complete graph K_m over m vertices is the skeleton graph of an orientable surface $\mathcal{S}_m$ of empty boundary. Moreover, the rotation system of $\mathcal{S}_m$ can be efficiently computed.*

The construction of the surface $\mathcal{S}_m$ can be found in [21]. The surface is completely specified by giving an appropriate rotation system for K_m. There are actually several such rotation systems [12], but in order to uniquely define $\mathcal{S}_m$, we focus on the one given in [21]. The proof of the efficient computability of the rotation system is straightforward based on the constructions in [21, 12], although it requires a tedious case study. We omit the details.

In the following definition, we will formalize the notion of "regular subdivision" of a surface, which consists in substituting every 2-simplex with a regular subdivision of it, as shown on Figure 1.

We will make use of the free Abelian monoid $\mathbb{N}[V]$ over the set of vertices V of a surface $\mathcal{S}$: the elements are those of the form $\sum_{v\in V} c_v \cdot v$, where c_v is a non-negative integer, and v is a vertex of $\mathcal{S}$. For any subset $V' \subseteq V$ and positive integer r let $\mathbb{N}_r[V']$ denote those elements $\sum_{v\in V'} c_v \cdot v$ of $\mathbb{N}[V]$ such that $\sum_{v\in V'} c_v = r$. If $s = \sum_{v\in V} s_v \cdot v$ and $t = \sum_{v\in V} t_v \cdot v$ are two elements of $\mathbb{N}[V]$, we denote by $d(s, t)$ the distance $\sum_{v\in V} |s_v - t_v|$.

Definition 11. Let $\mathcal{S}$ be a surface, and r be a positive integer. Let $\mathcal{S}^{(r)}$ be the simplicial complex whose maximal simplices are of the form $\{s_1, s_2, s_3\} \subseteq \mathbb{N}_r[\{a, b, c\}]$, for a 2-simplex $\{a, b, c\}$ in $\mathcal{S}$, and such that $d(s_1, s_2) = d(s_2, s_3) = d(s_1, s_3) = 2$. We call $\mathcal{S}^{(r)}$ the *regular r-subdivision* of $\mathcal{S}$.

The subdivision $\mathcal{S}^{(r)}$ of a surface $\mathcal{S}$ is again a surface with geometric realization homeomorphic to that of $\mathcal{S}$. Moreover, if $\mathcal{S}$ is oriented, then $\mathcal{S}^{(r)}$ inherits the orientation of $\mathcal{S}$.

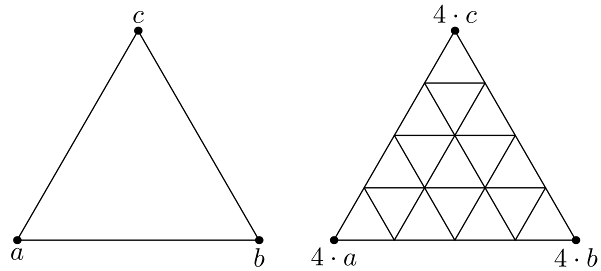


Figure 1: An elementary 2-complex and its regular 4-subdivision.

4 Sperner Problems

We state now a very general form of Sperner's Lemma due to Fan [13]. The exact formulation of the statement we reproduce here was given by Taylor in [30].

Definition 12. Let K be a simplicial complex. A *labelling of* K is a mapping ℓ of the vertices of K into the set $\{0, \ldots, \dim(K)\}$. If a simplex S of K is labelled with all possible labels, then we say that S *is fully labelled.*

A labelling ℓ naturally maps every oriented d-simplex $S = (v_0, \ldots, v_d)$ to the equivalence class $\ell(S) = [(\ell(v_0), \ldots, \ell(v_d))]_{\equiv_{d+1}}$.

Definition 13. Given a labelling ℓ of a simplicial complex K, and an integer $0 \leq d \leq \dim(K)$, we define the *d-dimensional flow* $N_d[\langle S \rangle]$ by

$$N_d[\langle S \rangle] = \begin{cases} 1 & \text{if } \ell(S) = [(0, 1, 2 \ldots, d)]_{\equiv_{d+1}}, \\ -1 & \text{if } \ell(S) = [(1, 0, 2, \ldots, d)]_{\equiv_{d+1}}, \\ 0 & \text{otherwise,} \end{cases}$$

and then extend it by linearity into a homomorphism $N_d : C_d(K; R) \to \mathbb{Z}$.

Theorem 1 (Sperner's Lemma [28, 13, 30]). *Let K be a simplicial complex of dimension d, let ℓ be a labelling of K, and let R be a ring. For an element C of $C_d(K; R)$, we have*

$$N_d[C] = (-1)^d N_{d-1}[\partial_d C].$$

Using Fact 2, Theorem 1 can be translated in terms of pseudo-manifolds.

Theorem 2 (Sperner's Lemma on pseudo-manifolds). *Let d be an integer, let $\mathcal{M}$ be a pseudo d-manifold, and let ℓ be a labelling of $\mathcal{M}$. Then we have*

$$N_d[\widehat{\mathcal{M}}] = (-1)^d N_{d-1}[\widehat{\partial \mathcal{M}}]$$

where $\begin{cases} \widehat{\mathcal{M}} \in C_d(\mathcal{M}, \mathbb{Z}/(2)),\ \widehat{\partial \mathcal{M}} \in C_{d-1}(\partial \mathcal{M}, \mathbb{Z}/(2)), & \textit{if } \mathcal{M} \textit{ is not oriented,} \\ \widehat{\mathcal{M}} \in C_d(\mathcal{M}, \mathbb{Z}),\ \widehat{\partial \mathcal{M}} \in C_{d-1}(\partial \mathcal{M}, \mathbb{Z}), & \textit{if } \mathcal{M} \textit{ is oriented.} \end{cases}$

This version of Sperner's lemma can be viewed, from a physicist's point of view, as a result equivalent to a global conservation law of a flow. If there is a source for the flow and the space is bounded then there must be a sink for that flow. More concretely, the lines of flow can be drawn over d-simplices, that goes from one d-simplex to another if they share a $(d-1)$-face that has all possible labels in $\{0,\ldots,d-1\}$. The sources and sinks of the flow are the fully labelled d-simplices. The lemma basically says that if the amount of flow entering the manifold at the boundary is larger than the exiting flow, then there must exist sinks inside.

The local conservation is stated by the fact that if there is an ingoing edge, there will not be two outgoing edges, and conversely. Formally, we have the following.

Fact 4. *Let (S,τ_S) be an oriented d-simplex. Then at most two of its oriented $(d-1)$-faces have a non-zero image by N_{d-1}. Moreover, if there are exactly two $(d-1)$-faces $(S',\tau^S_{S'})$ and $(S'',\tau^S_{S''})$ that have non-zero image by N_{d-1}, then $N_d[\langle(S,\tau_S)\rangle]=0$ and $N_{d-1}[\langle(S',\tau^S_{S'})\rangle]=-N_{d-1}[\langle(S'',\tau^S_{S''})\rangle]$.*

This gives a relation between the problem of finding fully labelled d-simplices and the natural complete problems for the parity argument classes. We can consider an oriented d-simplex (S,τ_S) with $N_d[\langle(S,\tau_S)\rangle]=1$ as a source for the flow, and $(S',\tau_{S'})$ with $N_d[\langle(S',\tau_{S'})\rangle]=-1$ as a sink.

We now state the algorithmical Sperner problems we will consider, starting with the black-box problems.

Sperner on Pseudo d-Manifolds (d-SPM)

Input: a pseudo d-manifold $\mathcal{M}$, and $S\in\mathcal{M}_d$.
Oracle input: a labelling $\ell:\mathcal{M}_0\to\{0,1,\ldots,d\}$.
Promise: one of the two conditions holds:
a) $N_{d-1}[\widehat{\partial\mathcal{M}}]=1$ in $C_{d-1}(\partial\mathcal{M},\mathbb{Z}/(2))$,
b) $N_{d-1}[\widehat{\partial\mathcal{M}}]=0$ in $C_{d-1}(\partial\mathcal{M},\mathbb{Z}/(2))$ and $N_d[\langle S\rangle]=1$ in $C_d(\mathcal{M},\mathbb{Z}/(2))$.
Output: $S'\in\mathcal{M}_d$ such that $N_d[\langle S'\rangle]=1$, and $S\neq S'$ for the second condition.

We will deal in particular with the following important special case of 2-**SPM**. Let $V_m=\{(i,j)\in\mathbb{N}^2\mid 0\le i+j\le m\}$. Observe that $|V_m|=\binom{m+2}{2}$

Regular Sperner (REGULAR 2-SPM)

Input: $n=\binom{m+2}{2}$ for some integer m.
Oracle input: a labelling $\ell:V_m\to\{0,1,2\}$.
Promise: for $0\le k\le m$, we have $\ell(0,k)\neq 1$, $\ell(k,0)\neq 0$, and $\ell(k,m-k)\neq 2$.
Output: p,p' and $p''\in V$, such that $p'=p+(\varepsilon,0)$, $p''=p+(0,\varepsilon)$ for some $\varepsilon\in\{-1,1\}$, and $\{\ell(p),\ell(p'),\ell(p'')\}=\{0,1,2\}$.

In fact, **REGULAR 2-SPM** on input $n=\binom{m+2}{2}$ is the instance of d-**SPM** on the regular m-subdivision of an elementary 2-simplex.

Oriented Sperner on Pseudo d-Manifolds (d-OSPM)

Input: an oriented pseudo d-manifold $\mathcal{M}$, and $S\in\mathcal{M}_d$.
Oracle input: a labelling $\ell:\mathcal{M}_0\to\{0,1,\ldots,d\}$.
Promise: one of the two conditions holds:
a) $(-1)^dN_{d-1}[\widehat{\partial\mathcal{M}}]<0$ in $C_{d-1}(\partial\mathcal{M},\mathbb{Z})$,
b) $(-1)^dN_{d-1}[\widehat{\partial\mathcal{M}}]=0$ in $C_{d-1}(\partial\mathcal{M},\mathbb{Z})$ and $N_d[\langle S\rangle]=1$ in $C_d(\mathcal{M},\mathbb{Z})$.
Output: $S'\in\mathcal{M}_d$ such that $N_d[\langle S'\rangle]=-1$.

Theorem 2 states that each of the previous problems has always a solution. The solution is not necessarily unique as it can be easily checked on simple instances. Thus the problems are not functional oracle problems.

The NP-search problem for which we prove completeness in Section 7 is the following. The surface $\mathcal{S}_m$ is the one given by Lemma 2. Its skeleton graph is K_m. The surface $\mathcal{S}_m^{(4)}$ is the regular 4-subdivision of $\mathcal{S}_m$.

Oriented Sperner Problem for the Surface $\mathcal{S}_m^{(4)}$ (OSPS)

Input: an integer m equal to 7 modulo 12, the description of a Turing machine M that on input vertex v of $\mathcal{S}_m^{(4)}$ outputs a label $\ell(v)$ in $\{0,1,2\}$ using time polynomial in $\log m$, and an oriented 2-simplex (T,τ_T) of $\mathcal{S}_m^{(4)}$, such that $N_2[\langle (T,\tau_T)\rangle] = 1$ in $C_2(\mathcal{S}_m^{(4)},\mathbb{Z})$.

Output: $(T',\tau_{T'})$ of $\mathcal{S}_m^{(4)}$, with $T' \neq T$, such that $N_2[\langle (T',\tau_{T'})\rangle] \neq 0$.

Again, Theorem 2 assures that there is always a solution. Observe that **OSPS** is in fact not a promise problem, since the input requirements can be syntactically enforced. To see this, we first provide a syntactical way to force the Turing machine to always give a correct output. One can assume, for instance, that every output value not in $\{0,1,2\}$ is interpreted as 0. We ensure syntactically that $N_2[\langle (T,\tau_T)\rangle] = 1$ with the help of an arbitrary polynomial time computable total order $<$ on the vertices of $\mathcal{S}_m^{(4)}$. Let $s_1 < s_2 < s_3$ be the 0-faces of T. The label of s_3 is fixed to 2. The 0-face s_1 will get label 0 and s_2 label 1 if the orientation specified by (s_1,s_2,s_3) is the same as τ_T, and the labels are exchanged in the opposite case.

5 Black-box algorithms for pseudo d-manifolds

The purpose of this section is to give a black-box algorithm for d-**SPM** and d-**OSPM**. To solve these problems, we adopt a divide and conquer approach. This kind of approach was successfully used in [19, 18] and [25], to study the query complexity of the oracle version of the Local Search problem.

In our algorithms the division of the pseudo d-manifold $\mathcal{M}$ will be done according to the combinatorial properties of its skeleton graph. The particular parameter we will need is its *iterated separation number* that we introduce now for general graphs.

Definition 14. Let $G = (V,E)$ be a graph. If A and C are subsets of V such that $V = A \cup C$, and that there is no edge between $A \setminus C$ and $C \setminus A$, then (A,C) is said to be a *separation* of the graph G, in notation $(A,C) \prec G$. The set $A \cap C$ is called a *separator* of the graph G.

The *iterated separation number* is defined by induction on the size of the graph G by

$$s(G) = \min_{(A,C)\prec G} \{|A\cap C| + \max(s(G[A\setminus C]), s(G[C\setminus A]))\}\,.$$

A pair $(A,C) \prec G$ such that $s(G) = |A\cap C| + \max(s(G[A\setminus C]), s(G[C\setminus A]))$ is called a *best separation* of G.

The iterated separation number of a graph is equal to the *value of the separation game* on the graph G, which was introduced in [19]. In that article, that value was defined as the gain of a player in a certain game. Notice, also, that the iterated separation number is at most $\log|V|$ times the *separation number* as defined in [25].

Before giving the algorithms, and their analyzes, we still need a few observations.

Lemma 3. *Let $\mathcal{A}$ and $\mathcal{B}$ be two pseudo d-manifolds, such that $\mathcal{A}\cup\mathcal{B}$ is also a pseudo d-manifold. Let ℓ be a labelling of $\mathcal{A}\cup\mathcal{B}$. If $\mathcal{A}$ and $\mathcal{B}$ have no d-simplex in their intersection, then*

$$N_d[\widehat{\mathcal{A}\cup\mathcal{B}}] = N_d[\widehat{\mathcal{A}}] + N_d[\widehat{\mathcal{B}}].$$

Proof. By definition, we have $N_d[\widehat{\mathcal{A} \cup \mathcal{B}}] = \sum_{S \in (\mathcal{A} \cup \mathcal{B})_d} N_d[\langle S \rangle] = \sum_{S \in \mathcal{A}_d} N_d[\langle S \rangle] + \sum_{S \in \mathcal{B}_d} N_d[\langle S \rangle] = N_d[\widehat{\mathcal{A}}] + N_d[\widehat{\mathcal{B}}]$. $\square$

Lemma 4. *Let $\mathcal{M}$ be a pseudo d-manifold, and $\mathcal{M}'$ be a union of elementary d-complexes such that $\mathcal{M}' \subseteq \mathcal{M}$. Then $\mathcal{M}'$ is a pseudo d-manifold.*

Proof. A $(d-1)$-face of a d-simplex in $\mathcal{M}'$ can not belong to more d-simplices in $\mathcal{M}'$ than in $\mathcal{M}$. $\square$

Theorem 3. *Let $\mathcal{M}$ be a pseudo d-manifold, H a subset of $\mathcal{M}_0$, and ℓ be a labelling of the vertices of $\mathcal{M}$. Let $(A, C) \prec G_{\mathcal{M}}[\mathcal{M}_0 \setminus H]$, $B = H \cup (A \cap C)$, and $M' = A \setminus C$ and $M'' = C \setminus A$. Denote by $\mathcal{B}$ the set of elementary d-complexes of $\mathcal{M}$ whose vertices are all in B, and by $\mathcal{M}'$ (*resp. *$\mathcal{M}''$) the set of elementary d-complexes of which at least one of the vertices belongs to M' (*resp. *M''). Denote also by $\mathcal{B}'$ the set of elementary $(d-1)$-complexes of $\mathcal{M}$ whose vertices are all in B. Then,*

- *(i) $\mathcal{B}$, $\mathcal{M}'$, $\mathcal{M}''$ and $\mathcal{M}' \cup \mathcal{M}''$ are pseudo d-manifolds,*
- *(ii) if $H \neq \mathcal{M}_0$ then $\mathcal{B}$, $\mathcal{M}'$ and $\mathcal{M}''$ are proper subsets of $\mathcal{M}$,*
- *(iii) $N_d[\widehat{\mathcal{M}}] = N_d[\widehat{\mathcal{B}}] + N_d[\widehat{\mathcal{M}'}] + N_d[\widehat{\mathcal{M}''}]$,*
- *(iv) the inclusions $\partial \mathcal{M}' \subseteq \partial \mathcal{M} \cup \mathcal{B}'$ and $\partial \mathcal{M}'' \subseteq \partial \mathcal{M} \cup \mathcal{B}'$ hold,*

Proof. Clearly, the complexes $\mathcal{B}$, $\mathcal{M}'$, $\mathcal{M}''$ and $\mathcal{M}' \cup \mathcal{M}''$ are pseudo d-manifolds, according to the previous lemma.

For *(ii)*, assume $H \neq \mathcal{M}_0$. Let x be a vertex in $A \setminus C$ and S be a d-simplex that contains it. By Fact 1, all the points of S are neighbors of x in the skeleton graph. The neighbors of x are all in A, since there is no edge between $A \setminus C$ and $C \setminus A$. Therefore, S is in $\mathcal{M}'$ but not in $\mathcal{B} \cup \mathcal{M}''$. By symmetry of $\mathcal{M}'$ and $\mathcal{M}''$, there is also a d-simplex which is not in $\mathcal{M}'$.

Let us now turn to prove *(iii)*. Let S be an elementary d-complex in $\mathcal{M}$. As B separates $G_{\mathcal{M}}[\mathcal{M}_0 \setminus H]$ into the two components M' and M'', it is not possible that S contains elements from both M' and from M'', as from Fact 1 we know that G_S is a complete graph. So, either all the vertices of S are in B, or in $B \cup M'$, or they are in $B \cup M''$. This proves that S belongs to $\mathcal{B} \cup \mathcal{M}' \cup \mathcal{M}''$. Therefore $\mathcal{M} \subseteq \mathcal{B} \cup \mathcal{M}' \cup \mathcal{M}''$. The converse inclusion clearly holds, which implies that it is in fact an equality. Moreover, from their definitions, the simplicial complexes $\mathcal{B}$, $\mathcal{M}'$ and $\mathcal{M}''$ have no d-simplex in common. Then using the first point, two applications of Lemma 3 allow us to deduce the announced equality in *(iii)*.

For *(iv)*, let S be a $(d-1)$-simplex in $\partial \mathcal{M}'$. It is not in $\partial \mathcal{M}$ if and only if it belongs to two d-simplices of $\mathcal{M}$. We will prove that if S belongs to two d-simplices of $\mathcal{M}$, then its 0-faces must all lie in B. Assume that S is a $(d-1)$-face common to two d-simplices T_1 and T_2. We can assume without loss of generality that T_1 belongs to $\mathcal{M}'$. But T_2 can not be in $\mathcal{M}'$ as S is in the boundary of $\mathcal{M}'$. So, either T_2 is in $\mathcal{B}$, or it is in $\mathcal{M}''$. In the first case, it immediately follows that S has all its 0-faces in B, as it is a face of a d-simplex whose 0-faces all lie in B. In the second case, again, the only possibility is that S has all its 0-faces in B, as else a vertex in M' and a vertex in M'' would be neighbors. This proves the third point for $\partial \mathcal{M}'$. The proof is the same for $\partial \mathcal{M}''$. $\square$

We are now ready to state Algorithm 1 and Algorithm 2, which solve d-**SPM** and d-**OSPM**, when the labels of the 0-faces of $\partial \mathcal{M}$ are also known. The final algorithms will first query these labels, and then call these procedures.

Algorithm 1 Main routine for solving d-**SPM**.

Input: A pseudo d-manifold $\mathcal{M}$, $S \in \mathcal{M}_d$, a set $H \supseteq (\partial\mathcal{M})_0$ together with the labels of its elements.

Let $(A, C) \prec G_{\mathcal{M}}[\mathcal{M}_0 \setminus H]$ be a best separation, and $B = H \cup (A \cap C)$.
Let the complexes $\mathcal{B}$, $\mathcal{M}'$ and $\mathcal{M}''$ be defined as in Theorem 3.
Query the labels of the vertices in $A \cap C$.
if $\mathcal{B}$ contains a fully labelled elementary d-complex **then**
  Return the corresponding oriented d-simplex.
end if
Evaluate $N_{d-1}[\widehat{\partial\mathcal{B}}]$, $N_{d-1}[\widehat{\partial\mathcal{M}'}]$ and $N_{d-1}[\widehat{\partial\mathcal{M}''}]$.
if $N_{d-1}[\widehat{\partial K}] = 1$ for $K \in \{\mathcal{B}, \mathcal{M}', \mathcal{M}''\}$ **then**
  Iterate the algorithm on K, any d-simplex $S \in K$, and B with the labels of its elements.
else
  Iterate the algorithm on $K \in \{\mathcal{B}, \mathcal{M}', \mathcal{M}''\}$ containing S, S and B with the labels of its elements.
end if

Algorithm 2 Main routine for solving d-**OSPM**.

Input: A pseudo d-manifold $\mathcal{M}$, $S \in \mathcal{M}_d$, a set $H \supseteq (\partial\mathcal{M})_0$ together with the labels of its elements.

Let $(A, C) \prec G_{\mathcal{M}}[\mathcal{M}_0 \setminus H]$ be a best separation, and $B = H \cup (A \cap C)$.
Let the complexes $\mathcal{B}$, $\mathcal{M}'$ and $\mathcal{M}''$ be defined as in Theorem 3.
Query the labels of the vertices in $A \cap C$.
if $\mathcal{B}$ contains a fully labelled elementary d-complex **then**
  Return the corresponding oriented d-simplex.
end if
Evaluate $N_{d-1}[\widehat{\partial\mathcal{B}}]$, $N_{d-1}[\widehat{\partial\mathcal{M}'}]$ and $N_{d-1}[\widehat{\partial\mathcal{M}''}]$.
if $(-1)^d N_{d-1}[\widehat{\partial K}] < 0$ on $K \in \{\mathcal{B}, \mathcal{M}', \mathcal{M}''\}$ **then**
  Iterate the algorithm on K, any d-simplex $S \in K$, and B with the labels of its elements.
else
  Iterate the algorithm on $K \in \{\mathcal{B}, \mathcal{M}', \mathcal{M}''\}$ containing S, S and B with the labels of its elements.
end if

We next give the result which states the correctness of our algorithms and specifies their complexities.

Lemma 5. *If $\mathcal{M}$ and S satisfy the promises of the respective Sperner problems, then Algorithms 1 and 2 return a solution and use at most $s(G_{\mathcal{M}}[\mathcal{M}_0 \setminus H])$ queries.*

Proof. We will prove the two claims for Algorithm 1 by induction, the proofs for Algorithm 2 are similar.

We start by proving the correctness. First observe that there is always enough information for the evaluations of the flows. Indeed by *(iv)* of Theorem 3, all the 0-faces of $\partial\mathcal{B}$, $\partial\mathcal{M}'$ and $\partial\mathcal{M}''$ are in $\partial\mathcal{M} \cup B$. The labels of the 0-faces of $\mathcal{M}$ are given as an input, and the labels of B are queried right before the flow evaluations.

Let us now consider an input that satisfies the promise of d-**SPM**, and where the number of d-simplices in $\mathcal{M}$ is n. If $n = 1$, then $\mathcal{M} = \mathcal{B}$ is an elementary d-complex, and by the promise, it is fully labelled. Therefore, the output of the algorithm is correct. When $n > 1$, we will prove that the recursive call will be made on an input which also satisfies the promise, and where the number of d-simplices in the pseudo d-manifold is less than n.

From Theorem 1 and *(iii)* of Theorem 3, we have $N_{d-1}[\widehat{\partial\,\mathcal{M}}] = N_{d-1}[\widehat{\partial\,\mathcal{B}}]+N_{d-1}[\widehat{\partial\,\mathcal{M}'}]+N_{d-1}[\widehat{\partial\,\mathcal{M}''}]$. In case *a)* of the promise, this sum is equal to 1, and therefore there exists a $K \in \{\mathcal{B}, \mathcal{M}', \mathcal{M}''\}$ for which $N_{d-1}[\widehat{\partial\,K}] = 1$. In case *b)* of the promise, either there exists a $K \in \{\mathcal{B}, \mathcal{M}', \mathcal{M}''\}$ for which $N_{d-1}[\widehat{\partial\,K}] = 1$, or there exists a $K \in \{\mathcal{B}, \mathcal{M}', \mathcal{M}''\}$ for which $N_{d-1}[\widehat{\partial\,K}] = 0$ and $S \in K$. In both cases, the number of d-simplices in K is less than the number of d-simplices of $\mathcal{M}$ because of *(ii)* of Theorem 3.

Let us now prove the bound on the complexity. For every pseudo d-manifold $\mathcal{M}$, denote by $T(\mathcal{M}, H)$ the number of queries made by the algorithm on $\mathcal{M}$ with the set H of labels. Each recursive call ends in three possible ways:

1) it stops after the first test if $\mathcal{M}$ is a fully labelled elementary d-complex,
2) it iterates on $\mathcal{B}$,
3) or it iterates on $\mathcal{M}'$ or on $\mathcal{M}''$.

In the first case, no queries are made, as all vertices of $\mathcal{M}$ are on its boundary. In the second case, $|A \cap C|$ queries are made, as in further iterations all labels will be known. In the third case, the number of queries is at most $|A \cap C| + \max(T(\mathcal{M}', B), T(\mathcal{M}'', B))$. Thus, we get $T(\mathcal{M}, H) \le |A \cap C| + \max(T(\mathcal{M}', B), T(\mathcal{M}'', B))$.

We now prove that $T(\mathcal{M}, H) \le s(G_{\mathcal{M}}[\mathcal{M}_0 \setminus H])$ for every pseudo d-manifold $\mathcal{M}$ and set of vertices $H \subseteq \mathcal{M}_0$. If $\mathcal{M}$ is an elementary d-complex, then the algorithm does not make any query, and therefore $T(\mathcal{M}, H) = 0$, and the statement is trivial.

Let now $\mathcal{M}$ be a pseudo d-manifold that is not an elementary d-complex. We have $G_{\mathcal{M}'}[\mathcal{M}'_0 \setminus B] \subseteq G_{\mathcal{M}}[M'] = G_{\mathcal{M}}[A \setminus C]$ and $G_{\mathcal{M}''}[\mathcal{M}''_0 \setminus B] \subseteq G_{\mathcal{M}}[M''] = G_{\mathcal{M}}[C \setminus A]$, for a best separation $(A, C) \prec G_{\mathcal{M}}[\mathcal{M}_0 \setminus H]$. Using the induction hypothesis, we get $T(\mathcal{M}', B) \le s(G_{\mathcal{M}'}[\mathcal{M}'_0 \setminus B])$ and $T(\mathcal{M}'', B) \le s(G_{\mathcal{M}''}[\mathcal{M}''_0 \setminus B])$. Since $s(G') \le s(G)$ if G' is a subgraph of G, we get $T(\mathcal{M}', B) \le s(G_{\mathcal{M}}[A \setminus C])$ and $T(\mathcal{M}'', B) \le s(G_{\mathcal{M}}[C \setminus A])$. As (A, C) is a best separation of $G_{\mathcal{M}}[\mathcal{M}_0 \setminus H]$, this proves the inequality $T(\mathcal{M}, H) \le s(G_{\mathcal{M}}[\mathcal{M}_0 \setminus H])$. □

Theorem 4. $\mathsf{DQC}(d\text{-}\mathbf{SPM}) = O(s(G_{\mathcal{M}}[\mathcal{M}_0 \setminus (\partial\,\mathcal{M})_0]) + |(\partial\mathcal{M})_0|)$ *and* $\mathsf{DQC}(d\text{-}\mathbf{OSPM}) = O(s(G_{\mathcal{M}}[\mathcal{M}_0 \setminus (\partial\,\mathcal{M})_0]) + |(\partial\mathcal{M})_0|)$.

Proof. The algorithms consist in querying the labels of the vertices of $\partial\mathcal{M}$ and then running respectively Algorithm 1 or Algorithm 2 with the initial choice $H = (\partial\,\mathcal{M})_0$. □

To bound the complexity of our algorithms we need an upper-bound on the iterated separator number of the skeleton graph. The following theorem gives, for any graph, an upper bound on the size of a balancing separator, whose deletion leaves the graph with two roughly equal size components. The bound depends on the genus and the number of vertices of the graph.

Theorem 5 (Gilbert, Hutchinson, Tarjan [14]). *A graph of genus g with n vertices has a set of at most $6\sqrt{g \cdot n} + 2\sqrt{2n} + 1$ vertices whose removal leaves no component with more than $2n/3$ vertices.*

In [14], there is an algorithm to find a separator with the required properties which uses an embedding of the graph in a surface of genus g. However, already finding the genus of a graph is an NP-complete problem. Thus, this approach does not yield an effective way to generate a balanced separator. In a recent work, Kelner [16] constructed an efficient algorithm to generate a balanced separator using a different approach.

For our purposes we can immediately derive an upper bound on the iterated separation number.

Corollary 1. *For graphs $G = (V, E)$ of size n and genus g we have $s(G) \le \lambda(6\sqrt{g \cdot n} + 2\sqrt{2n}) + \log_{3/2} n$, where λ is solution of $\lambda = 1 + \lambda\sqrt{2/3}$.*

Proof. Let us prove this fact by induction over n. It obviously holds for $n = 1$. Assume now that $n > 1$. Theorem 5 shows that there exist three pairwise disjoint sets S_1, S_2 and S_3 such that $V = S_1 \cup S_2 \cup S_3$, $|S_2| \leq 6\sqrt{g \cdot n}+2\sqrt{2n}+1$ and $|S_1|, |S_3| \leq 2n/3$. If we let $A = S_1 \cup S_2$ and $C = S_2 \cup S_3$, then $(A, C) \prec G$ and $A \cap C = S_2$. The construction implies that $|A \setminus C|, |C \setminus A| \leq 2n/3$. Using the induction hypothesis, we get

$$\begin{aligned} s(G) &\leq |A \cap C| + \max(s(G[A \setminus C]), s(G[C \setminus A])) \\ &\leq 6\sqrt{g \cdot n} + 2\sqrt{2n} + 1 + \lambda(6\sqrt{g \cdot 2n/3} + 2\sqrt{2 \cdot 2n/3}) + \log_{3/2}(2n/3) \\ &\leq \lambda(6\sqrt{g \cdot n} + 2\sqrt{2n}) + \log_{3/2} n. \end{aligned}$$

□

In general, there is no immediate relationship between the genus of a pseudo d-manifold and the genus of its skeleton graph. However, if the pseudo d-manifold $\mathcal{M}$ is a triangulated oriented surface, then the genus of the graph is equal to the genus of $\mathcal{M}$.

Used in conjunction with Corollary 1, Theorem 4 gives an effective upper bound for pseudo d-manifolds.

Corollary 2. *Let $\mathcal{M}$ be a pseudo d-manifold such that $G_{\mathcal{M}}$ is of size n and of genus g. Then,*

$$\mathsf{DQC}(d\text{-}\mathbf{SPM}) = O(\sqrt{(g+1)n} + |(\partial\mathcal{M})_0|) \textit{ and } \mathsf{DQC}(d\text{-}\mathbf{OSPM}) = O(\sqrt{(g+1)n} + |(\partial\mathcal{M})_0|).$$

Since the skeleton graph of the underlying pseudo 2-manifold of **REGULAR** 2-**SPM** is planar, it has genus 0. Thus we get:

Theorem 6. $\mathsf{DQC}(\mathbf{REGULAR}\ 2\text{-}\mathbf{SPM}) = O(\sqrt{n})$.

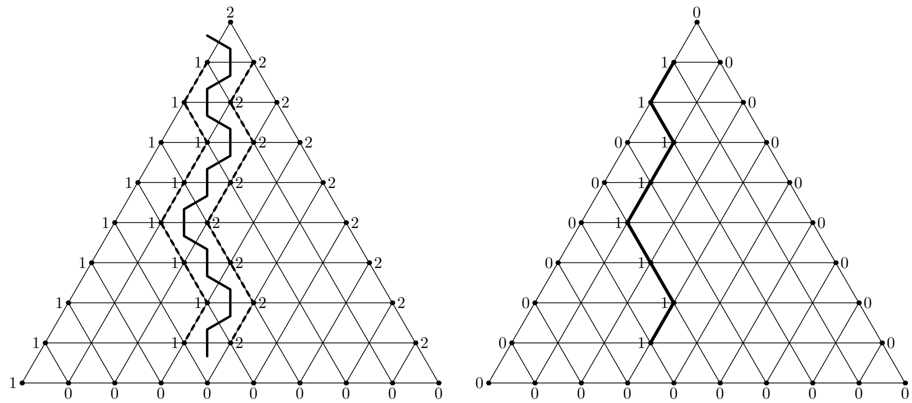
In the next section, we show nontrivial lower bounds on the randomized and the quantum query complexity of the **REGULAR** 2-**SPM** problem. Observe that for some general instances of the 2-**SPM** over the same pseudo 2-manifold we can easily derive exact lower bounds from the known complexity of Grover's search problem [7]. For example, if a labelling is 2 everywhere, except on two consecutive vertices on the boundary where it takes respectively the values 0 and 1, then finding a fully labelled 2-simplex is of the same complexity as finding a distinguished element on the boundary.

6 Lower bounds for REGULAR 2-SPM

We denote by **UNIQUE-SPERNER** all those instances of **REGULAR** 2-**SPM** for which there exists a unique fully labelled triangle. There exist several equivalent adversary methods for proving quantum lower bounds in the query model [29]. Here, we will use the weighted adversary method [2, 4, 17].

Theorem 7. *Let Σ be a finite set, let $n \geq 1$ be an integer, and let $S \subseteq \Sigma^n$ and S' be sets. Let $f : S \to S'$. Let Γ be an arbitrary $S \times S$ nonnegative symmetric matrix that satisfies $\Gamma[x, y] = 0$ whenever $f(x) = f(y)$. For $1 \leq k \leq n$, let Γ_k be the matrix*

$$\Gamma_k[x, y] = \begin{cases} 0 & \textit{if } x_k = y_k, \\ \Gamma[x, y] & \textit{otherwise.} \end{cases}$$



(i) *(ii)*

Figure 2: In the coordinates system of the Figure, the point $(0,0)$ is the highest corner of the triangles, the x coordinates increase by going down and left, and the y coordinates increase by going down and right. On sub-figure *(i)*, the labelling C_b corresponds to the binary sequence $b = 0100110$. On sub-figure *(ii)*, the labelling O_b corresponds to the same sequence b. The unmarked vertices are all labelled 0.

For all $S \times S$ matrix M and $x \in S$, let $\sigma(M,x) = \sum_{y \in S} M[x,y]$. Then

$$\mathsf{QQC}(f) = \Omega\left(\min_{\Gamma[x,y]\neq 0, x_k \neq y_k} \sqrt{\frac{\sigma(\Gamma,x)\sigma(\Gamma,y)}{\sigma(\Gamma_k,x)\sigma(\Gamma_k,y)}}\right),$$

$$\mathsf{RQC}(f) = \Omega\left(\min_{\Gamma[x,y]\neq 0, x_k \neq y_k} \max\left(\frac{\sigma(\Gamma,x)}{\sigma(\Gamma_k,x)}, \frac{\sigma(\Gamma,y)}{\sigma(\Gamma_k,y)}\right)\right).$$

For the lower bound we will consider specific instances of **REGULAR** 2-**SPM**. For that, we need a few definitions. For any binary sequence b, let $|b|$ denote the length of the sequence b, and for $i = 0, 1$ let $w_i(b)$ be the number of bits i in b. For $0 \leq t \leq |b|$, let $b^t = b_1 \ldots b_t$ denote the prefix of length t of b.

The instances of **REGULAR** 2-**SPM** we will consider are those whose oracle inputs C_b are induced by binary sequences $b = b_1 \ldots b_{m-2}$ of length $m-2$ as follows:

$$C_b(i,j) = \begin{cases} 1 & \text{if } j = 0 \text{ and } i \neq 0, \\ 2 & \text{if } i = 0 \text{ and } j \neq m, \\ 0 & \text{if } i + j = m \text{ and } j \neq 0, \\ 1 & \text{if there exists } 0 \leq t \leq m-2 \text{ such that } (i,j) = (w_0(b^t)+1, w_1(b^t)), \\ 2 & \text{if there exists } 0 \leq t \leq m-2 \text{ such that } (i,j) = (w_0(b^t), w_1(b^t)+1), \\ 0 & \text{otherwise.} \end{cases}$$

Notice that the first and fourth (*resp.* second and fifth) conditions can be simultaneously satisfied, but the labelling definition is consistent. Also observe that, for any b, there is a unique fully labelled triangle, whose coordinates are $\{(w_0(b)+1, w_1(b)), (w_0(b), w_1(b)+1), (w_0(b)+1, w_1(b)+1)\}$. Therefore C_b is an instance of **UNIQUE-SPERNER**. We illustrate an instance of C_b in Figure 2.

It turns out that technically it will be easier to prove the lower bound for a problem which is closely related to the above instances of **REGULAR** 2-**SPM**, that we call **SNAKE**. Recall that $V_m = \{(i,j) \in$

$\mathbb{N}^2 \mid 0 \leq i{+}j \leq m\}$. For every binary sequence $b = b_1 \dots b_{m-2}$, we denote by O_b the function $V_m \to \{0,1\}$ defined for $p \in V_m$ by

$$O_b(p) = \begin{cases} 1 & \text{if there exists } 0 \leq t \leq m-2 \text{ such that } (i,j) = (w_0(b^t)+1, w_1(b^t)), \\ 0 & \text{otherwise.} \end{cases}$$

See again Figure 2 for an example.

SNAKE

Input:	$n = \binom{m}{2}$ for some integer m.
Oracle input:	a function $f : V_m \to \{0,1\}$.
Promise:	there exists a binary sequence $b = b_1 \dots b_{m-2}$ such that $f = O_b$.
Output:	$(w_0(b), w_1(b))$.

We recall here the definition of [25] of c-query reducibility between black-box problems, which we will use to prove our lower bound.

Definition 15. For an integer $c > 0$, a functional oracle problem $A : S_1 \to T_1$ with $S_1 \subseteq \Sigma_1^n$ is *c-query reducible* to a functional oracle problem $B : S_2 \to T_2$ with $S_2 \subseteq \Sigma_2^{n'}$ if the following two conditions hold:

(i) $\exists \alpha : S_1 \to S_2, \quad \exists \beta : T_2 \to T_1$, such that $\forall x \in S_1,\ A(x) = \beta(B(\alpha(x)))$,

(ii) $\exists \gamma_1, \dots, \gamma_c : \{1, \dots, n'\} \to \{1, \dots, n\}$ and $\gamma : \{1, \dots, n'\} \times \Sigma_1^c \to \Sigma_2$ such that $\forall x \in S_1$, $k \in \{1, \dots, n'\}, \quad \alpha(x)(k) = \gamma(k, x_{\gamma_1(k)}, \dots, x_{\gamma_c(k)})$.

Lemma 6 ([25]). *If A is c-query reducible to B then $\mathsf{QQC}(B) \geq \mathsf{QQC}(A)/2c$, and $\mathsf{RQC}(B) \geq \mathsf{RQC}(A)/c$.*

Lemma 7. **SNAKE** *is 3-query reducible to* **UNIQUE-SPERNER**.

Proof. We define the oracle transformations as $\alpha(O_b) = C_b$, and

$$\beta(\{(i_1,j_1),(i_2,j_2),(i_3,j_3)\}) = (\min\{i_1,i_2,i_3\}, \min\{j_1,j_2,j_3\}).$$

Obviously, α and β satisfy the first condition of the definition.

We now turn to the simulation of an oracle for **UNIQUE-SPERNER** by an oracle for **SNAKE**. If the query concerns a point (i,j) on the boundary, the answer is independent from the oracle, and is given according to the definition of C_b, for any b. Otherwise, the simulator will query the point and its left and right neighbors in the sense of the Figure 2, from which the value of C_b can be easily determined. Formally, for such a point (i,j), let the functions γ, γ_1, γ_2 and γ_3 be defined as

$$\gamma((i,j),a_1,a_2,a_3) = \begin{cases} 1 & \text{if } (a_1,a_2,a_3) = (0,1,0), \\ 2 & \text{if } (a_1,a_2,a_3) = (1,0,0), \\ 0 & \text{otherwise,} \end{cases}$$

and $\gamma_1(i,j) = (i+1, j-1)$, $\gamma_2(i,j) = (i,j)$, $\gamma_3(i,j) = (i-1, j+1)$. ∎

Lemma 8. *The query complexity of* **SNAKE** *f satisfies*

$$\mathsf{RQC}(\textbf{SNAKE}) = \Omega(\sqrt[4]{n}),$$
$$\mathsf{QQC}(\textbf{SNAKE}) = \Omega(\sqrt[8]{n}).$$

Proof. We give now the definition of the adversary matrix Γ which will be a $2^{m-2} \times 2^{m-2}$ symmetric matrix, whose rows and columns will be indexed by the labellings O_b, when $b \in \{0,1\}^{m-2}$. For the sake of simplicity, we will only use binary sequences to denote rows and columns, instead of the induced labellings. For two binary sequences b and b', we denote by $b \wedge b'$ their longest common prefix. Then let

$$\Gamma[b,b'] = \begin{cases} 0 & \text{if } w_0(b) = w_0(b'), \\ 2^{|b \wedge b'|} & \text{otherwise.} \end{cases}$$

For a given binary sequence b, there are $2^{m-2-(d+1)}$ sequences that have longest common prefix of length d with b. Out of them, $\binom{m-3-d}{w_{b_{d+1}}(b)-w_{b_{d+1}}(b^d)}$ will give the same output as b. Therefore,

$$\begin{aligned} \sigma(\Gamma, b) &= \sum_{d=0}^{m-4} 2^d \left[2^{m-3-d} - \binom{m-3-d}{w_{b_{d+1}}(b) - w_{b_{d+1}}(b^d)} \right] \\ &\geq (m-3)2^{m-3} - \sum_{d=0}^{m-4} 2^d \binom{m-3-d}{\lfloor (m-3-d)/2 \rfloor} \\ &\geq (m-3)2^{m-3} - \left(\sum_{d=0}^{m-4} \frac{2^{m-3}}{\sqrt{m-3-d}} \right) \\ &\geq (m-3)2^{m-3} - O(\sqrt{m}2^m) = \Omega(m2^m). \end{aligned}$$

We now turn to bound from above $\sigma(\Gamma_p, b)$ and $\sigma(\Gamma_p, b')$ when $\Gamma[b,b'] \neq 0$ and $O_b(p) \neq O_{b'}(p)$. Let us now fix a point $p = (i,j)$ with $1 \leq i+j \leq m-1$, and two sequences b and b', such that $O_b(p) \neq O_{b'}(p)$. We assume that $O_b(p) = 0$ and $O_{b'}(p) = 1$. We trivially upper bound $\sigma(\Gamma_p, b')$ by $\sigma(\Gamma, b') = O(m2^m)$.

We will now upper-bound $\sigma(\Gamma_p, b)$. Set $h = i+j$. If a sequence b'' is such that $O_{b''}(p) = 1$, then the length of its longest common prefix with b is at most $h-2$. We regroup these sequences according to the value $|b \wedge b''|$. The number of sequences b'' for which $|b \wedge b''| = d$ and $O_{b''}(p) = 1$ is at most $\binom{h-1-(d+1)}{\lfloor h-1-(d+1))/2 \rfloor} 2^{m-h-1}$. Therefore we can bound $\sigma(\Gamma_p, b)$ as

$$\begin{aligned} \sigma(\Gamma_p, b) &\leq \sum_{d=0}^{h-2} 2^d \cdot \binom{h-d-2}{\lfloor (h-d-2)/2 \rfloor} 2^{m-h-1} \\ &\leq \sum_{d=0}^{h-2} \left[2^{-(h-d-2)} \cdot \binom{h-d-2}{\lfloor (h-d-2)/2 \rfloor} \right] 2^{m-3} = O(\sqrt{m}2^m). \end{aligned}$$

By Theorem 7 we conclude that

$$\begin{aligned} \mathsf{RQC}(\textbf{SNAKE}) &= \Omega\left(\max\left(\frac{m2^m}{m2^m}, \frac{m2^m}{\sqrt{m}2^m} \right) \right) = \Omega(\sqrt[4]{n}), \\ \mathsf{QQC}(\textbf{SNAKE}) &= \Omega\left(\frac{m2^m}{\sqrt{m2^m \cdot \sqrt{m}2^m}} \right) = \Omega(\sqrt[8]{n}). \end{aligned}$$

□

Theorem 8. *The query complexity of* **REGULAR** 2-**SPM** *satisfies*

$$\begin{aligned} \mathsf{RQC}(\textbf{REGULAR } 2\textbf{-SPM}) &= \Omega(\sqrt[4]{n}), \\ \mathsf{QQC}(\textbf{REGULAR } 2\textbf{-SPM}) &= \Omega(\sqrt[8]{n}). \end{aligned}$$

Proof. By Lemma 6 and 7, the lower bounds of Lemma 8 for **SNAKE** also apply to **REGULAR** 2-**SPM**. □

7 Completeness results

Let m be a positive integer equal to 7 modulo 12. We will work with the regular 4-subdivision $\mathcal{S}_m^{(4)}$ of $\mathcal{S}_m$.

Theorem 9. *The problem* **OSPS** *is PPAD-complete.*

Proof. To see membership in PPAD, we reduce **OSPS** to the natural complete problem for PPAD. Let V be the set of oriented 2-simplices in $\mathcal{S}_m^{(4)}$, and E be the set

$$E = \{((S, \tau_S), (S', \tau_{S'})) \in V^2 \mid S \cap S' \text{ is a 1-simplex in } \mathcal{S}_m^{(4)} \text{ and } N_1[\langle (S \cap S', \tau^S_{S\cap S'})\rangle] > 0\}.$$

We call the directed graph $G = (V, E)$ the N_1-flow-graph. The reduction follows the flow argument based on Fact 4, of Section 4. All that we should provide is a polynomial time Turing machine that gives the possible predecessor and successor of a vertex v in G. This can be done as follows.

Given the oriented 2-simplex $v = (S, \tau_S)$ in $\mathcal{S}_m^{(4)}$, the machine determines the 1-faces of v for which N_1 is non-zero, with their induced orientations. A 1-face for which its value is 1 (*resp.* -1) is common with only one other 2-simplex, which is its successor (*resp.* its predecessor). Given a 1-face, recovering the other 2-simplex $(S', \tau_{S'})$ to which it belongs, can be done efficiently using the rotation system.

We turn to the proof of completeness. Let k be any positive integer. Let $G = (V, E)$ be a graph which is specified by an instance of the natural complete problem for PPAD (see Section 2). It is an oriented graph over $V = \{0, 1\}^k$, such that each vertex has indegree at most one, and outdegree at most one. Moreover, 0^k is a source in G. Let us denote by M the polynomial time Turing machine that, given a vertex $v \in V$, outputs its predecessor and its successor, if they exist. From G we make an instance of **OSPS** such that a solution can be efficiently turned into a source or a sink of the graph G different from 0^k.

Let m be the smallest integer greater than 2^k that is equal to 7 modulo 12. We assume that V is included in the set of vertices of $\mathcal{S}_m$. We denote by $\Pi = \{\pi_v \mid v \text{ vertex of } \mathcal{S}_m\}$ the rotation system for $\mathcal{S}_m$.

Informally, we give a labelling such that the N_1-flow-graph imitates the graph G as follows: if (a, b) is an edge of G, then there will be a small path on the N_1-flow-graph going along the edges near the (a, b) side of the triangle "above" (a, b) (that is the triangle $\{a, b, \pi_a^{-1}(b)\}$). If moreover (b, c) is an edge in G then there will be a path around b in the direction given by the rotation system, leading to the triangle above (b, c). To manage the latter, we need a tool for deciding whether, for a 0-simplex $d \notin \{a, b, c\}$, the 1-simplex $\{b, d\}$ is "between" $\{a, b\}$ and $\{b, c\}$ according to the rotation π_b. This tool is provided by the function $\log_a^{\pi_b}$ defined in Definition 10: the 1-simplex $\{b, d\}$ is between $\{a, b\}$ and $\{b, c\}$ if $0 < \log_a^{\pi_b}(d) < \log_a^{\pi_b}(c)$. The function $\log_a^{\pi_b}$ is efficiently computable by Lemma 2.

We design a Turing machine M' that for every vertex v in $\mathcal{S}_m^{(4)}$ outputs a label $\ell(v)$ in $\{0, 1, 2\}$, using M as a subroutine. Let $(a, b, c) = (S, \tau_S)$ be an oriented 2-simplex in $\mathcal{S}_m$, and let i_a, i_b and i_c be three non-negative integers such that $i_a + i_b + i_c = 4$. Denote by σ the permutation $\binom{a,b,c}{b,c,a}$. Observe that the definition of the rotation system implies that for every $v \in \{a, b, c\}$ the equality $\pi_v(\sigma^{-1}(v)) = \sigma(v)$ holds.

On input $z = i_a \cdot a + i_b \cdot b + i_c \cdot c$ the Turing machine M' outputs

$$\ell(z) = \begin{cases} 0 & \text{if } \exists v, v' \in S,\ i_v + i_{v'} = 4,\ (v, v') \in E, \quad (1) \\ 0 & \text{if } \exists v \in S,\ i_v = 4,\ \exists w \notin S,\ (v, w) \in E \text{ or } (w, v) \in E, \quad (2) \\ 1 & \text{if } \exists v \in S,\ (i_v, i_{\sigma(v)}) \in \{(2,1), (1,2)\},\ (v, \sigma(v)) \in E, \quad (3) \\ 1 & \text{if } \exists v \in S,\ \exists v' \in \{\sigma^{-1}(v), \sigma(v)\},\ (i_v, i_{v'}) = (3,1), \\ & \quad \exists w, w' \in V,\ (w, v), (v, w') \in E \text{ and } \log_w^{\pi_v}(v') < \log_w^{\pi_v}(w'), \quad (4) \\ 2 & \text{otherwise.} \quad (5) \end{cases}$$

Finding the case in which z falls can be done in time polynomial in k, as the Turing machine M, on input $v \in \{a, b, c\}$, outputs the neighbors of v, and the rotation system Π can be efficiently computed.

Using these rules, we describe (see Figure 3) the possible cases for an oriented 2-simplex (a, b, c) in $\mathcal{S}_m$ (we assume that the rotation system is clockwise, and hence the orientation is counter-clockwise):

Case 1: (a, b), (b, c), $(c, a) \in E$.

Case 2: (a, b), $(b, c) \in E$, but $(c, a) \notin E$. The value of $\ell(3 \cdot a + c)$ is 2 if a is a source in G, and 1 otherwise. Similarly, the value of $\ell(3 \cdot c + a)$ is 2 if c is a sink in G, and 1 otherwise.

Case 3: $(a, b) \in E$, but (b, c) and (c, a) are not in E. The value of $\ell(4 \cdot c)$ is 2 if c is isolated in G, and otherwise 0. The value of $\ell(a + 3 \cdot c) = \ell(b + 3 \cdot c)$ is 1 if $\log_w^{\pi_c}(b) < \log_w^{\pi_c}(a) < \log_w^{\pi_c}(w')$, and otherwise 2. The value of $\ell(3 \cdot a + c)$ is 2 if a is a source in G, and 1 otherwise. The value of $\ell(3 \cdot b + c)$ is 2 if b is a sink in G, and 1 otherwise.

Case 4: (a, b), (b, c) and (c, a) are not in E. Let v be in $\{a, b, c\}$. We do not enumerate all the possible subcases, but only state the essential relations between the labels:

(i) $\ell(3 \cdot v + \sigma(v)) = 1 \iff \ell(3 \cdot v + \sigma^{-1}(v)) = 1$, as both $3 \cdot v + \sigma^{-1}(v)$ and $3 \cdot v + \sigma(v)$ simultaneously fall in one of the cases (1), (4) and (5) in the definition of ℓ.

(ii) $\ell(3 \cdot v + \sigma(v)) = 0 \iff \ell(2 \cdot v + 2 \cdot \sigma(v)) = 0$, as if $\ell(3 \cdot v + \sigma(v)) = 0$ or $\ell(2 \cdot v + 2 \cdot \sigma(v)) = 0$ then case (1) in the definition of ℓ must apply,

(iii) $\ell(3 \cdot v + \sigma^{-1}(v)) = 0 \iff \ell(2 \cdot v + 2 \cdot \sigma^{-1}(v)) = 0$, for the same reasons as previously.

These are the only possible cases, up to renaming the vertices a, b and c, but preserving the orientation of the 2-simplex (a, b, c).

We have to prove that this labelling scheme ℓ is correctly defined. It is easy to check that it is correctly defined on $4 \cdot v$, where v is a vertex of V: if v is an isolated vertex in G, then in every face to which it belongs only the case (5) in the definition of ℓ applies, and therefore $\ell(4 \cdot v) = 2$. If v is not isolated, then case (2) in the definition of ℓ applies, and therefore $\ell(4 \cdot v) = 0$.

So, finally, proving that the labelling has been correctly defined amounts to proving that the label $\ell(z)$ of a vertex $z = i_a \cdot a + i_b \cdot b$, $0 < i_a, i_b < 4$ with $i_a + i_b = 4$, that we have defined is the same for the two oriented 2-simplices $(a, b, \pi_a^{-1}(b))$ and $(a, \pi_a(b), b)$. We study the different cases:

- $(i_a, i_b) = (3, 1)$ or $(1, 3)$: if (a, b) or (b, a) is in E, then case (1) in the definition of ℓ applies to z, and $\ell(z) = 0$. Otherwise, either case (4) applies and therefore $\ell(z) = 1$, or case (5) applies and therefore $\ell(z) = 2$.

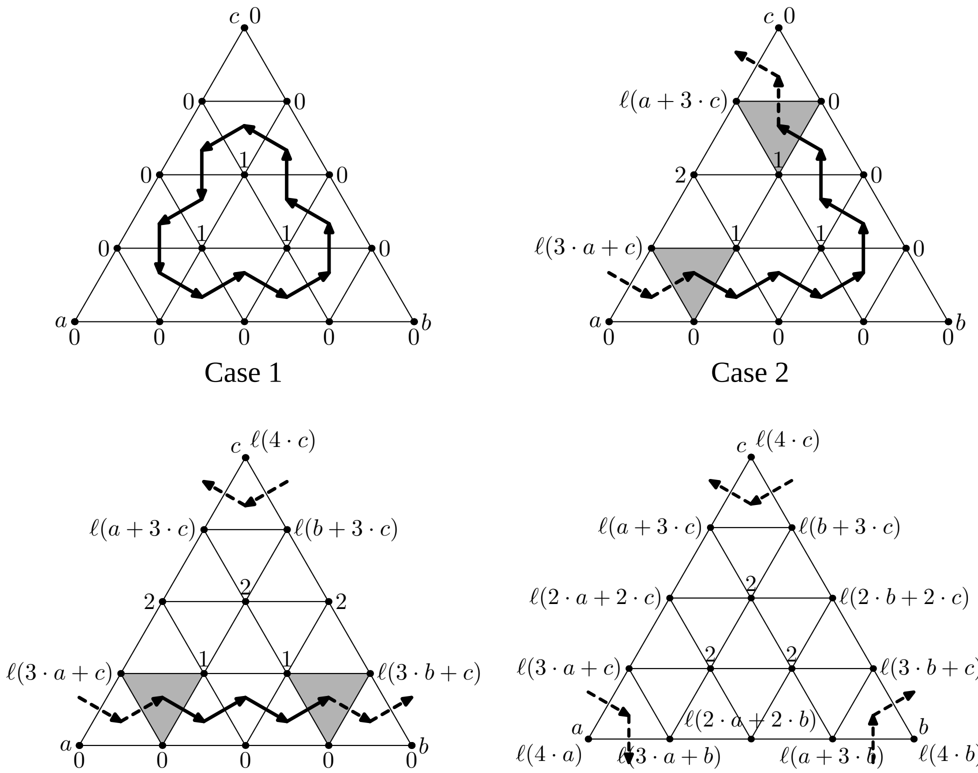


Figure 3: The different possible cases in the labelling of an oriented 2-simplex (a, b, c) of $\mathcal{S}_m^{(4)}$. The possibly fully labelled 2-simplices are grayed.

- $(i_a, i_b) = (2, 2)$: if (a, b) or (b, a) is in E, then case (1) in the definition of ℓ applies to z, and $\ell(z) = 0$. Otherwise, case (5) applies.

Let (a', b', c') be an oriented 2-simplex in the subdivision of an oriented 2-simplex (a, b, c) in $\mathcal{S}_m$, such that $N_2[\langle(a', b', c')\rangle] \neq 0$. We prove that there exists a unique $v = v(a', b', c') \in \{a, b, c\}$ such that $(a', b', c') = (3 \cdot v + \sigma(v), 2 \cdot v + \sigma^{-1}(v) + \sigma(v), 3 \cdot v + \sigma^{-1}(v))$, and v is a source in G if $N_2[\langle(a', b', c')\rangle] = 1$, and a sink if $N_2[\langle(a', b', c')\rangle] = -1$. Also, given (a', b', c'), one can efficiently retrieve $v(a', b', c')$. The proof is done for the different cases of Figure 3.

In Case 1 there is no such simplex $\{a', b', c'\}$.

Let us examine Case 2. The possible values for $\ell(3 \cdot a + c)$ and $\ell(a + 3 \cdot c)$ are 1 and 2. Therefore, the only possibilities for (a', b', c') are $(b + 3 \cdot c, a + b + 2 \cdot c, a + 3 \cdot c)$ when $\ell(a + 3 \cdot c) = 2$, and $(3 \cdot a + c, 2 \cdot a + b + c, 3 \cdot a + b)$ when $\ell(3 \cdot a + c) = 2$. These values correspond respectively to the case when c is a sink and $N_2[\langle(a', b', c')\rangle] = -1$, and to the case when a is a source and $N_2[\langle(a', b', c')\rangle] = 1$.

Let us turn to Case 3. In this case, we always have $\ell(a + 3 \cdot c) = \ell(b + 3 \cdot c)$. So, the only possibilities for (a', b', c') are $(3 \cdot b + a, a + 2 \cdot b + c, 3 \cdot b + c)$ when $\ell(3 \cdot b + c) = 2$, and $(3 \cdot a + c, 2 \cdot a + b + c, 3 \cdot a + b)$ when $\ell(3 \cdot a + c) = 2$. These values correspond respectively to the case when b is a sink and $N_2[\langle(a', b', c')\rangle] = -1$, and to the case when a is a source and $N_2[\langle(a', b', c')\rangle] = 1$.

We finish the case study by proving that in Case 4, there can be no oriented 2-simplex (a', b', c') such that $N_2[\langle(a', b', c')\rangle] \neq 0$. All the 2-simplices that have twice the label 2 can immediately be discarded.

By symmetry between a, b and c, we can assume without loss of generality that a', b' and c' should be in $\{i_a \cdot a + i_b \cdot b + i_c \cdot c \in \mathbb{N}_4[\{a,b,c\}] \mid i_a \geq 2\}$. Assume that (a', b', c') is a fully labelled triangle. The possibilities are:

- $(a', b', c') = (3 \cdot a + c, 4 \cdot a, 3 \cdot a + b)$: $\ell(4 \cdot a) \in \{0, 2\}$, so $\ell(3 \cdot a + c) = 1$ or $\ell(3 \cdot a + b) = 1$, and therefore relation *(i)* implies $\ell(3 \cdot a + c) = \ell(3 \cdot a + b)$, which is impossible,
- $(a', b', c') = (3 \cdot a + c, 3 \cdot a + b, 2 \cdot a + b + c)$: similar to the previous case,
- $(a', b', c') = (2 \cdot a + 2 \cdot c, 3 \cdot a + 2 \cdot c, 2 \cdot a + b + c)$: $\ell(2 \cdot a + 2 \cdot c) \in \{0, 2\}$ and $\ell(2 \cdot a + b + c) = 2$, so $\ell(2 \cdot a + 2 \cdot c) = 0$ and therefore relation *(ii)* implies $\ell(3 \cdot a + 2 \cdot c) = 0$, which is impossible,
- $(a', b', c') = (3 \cdot a + b, 2 \cdot a + 2 \cdot b, 2 \cdot a + b + c)$: similar to the previous case, using relation *(iii)*.

Our next step is showing that the map $(a', b', c') \mapsto v(a', b', c')$ is a bijection between oriented 2-simplices (a', b', c') such that $N_2[\langle (a', b', c') \rangle] = 1$ and sources of G. It is onto, as if v is a source in G, v' is the successor of v and $v'' = \pi_v^{-1}(v')$ then $v = v(a', b', c')$, where $(a', b', c') = (3 \cdot v + v'', 2 \cdot v + v' + v'', 3 \cdot v + v'')$. The case study also shows that if (a', b', c') is an oriented 2-simplex of $\mathcal{S}_m^{(4)}$ such that $N_2[\langle (a', b', c') \rangle] = 1$ and $v = v(a', b', c')$ then $(a', b', c') = (3 \cdot v + v'', 2 \cdot v + v' + v'', 3 \cdot v + v'')$ as oriented simplices, where v' is the successor of v in G and $v'' = \pi_v^{-1}(v')$. Therefore the map is injective as well. A similar bijection exists between oriented 2-simplices (a', b', c') such that $N_2[\langle (a', b', c') \rangle] = -1$ and sinks of G.

Let $(a_0, b_0, c_0) = (0^k, 1^k, \pi_{0^k}^{-1}(1^k))$. The oriented 2-simplex (T, τ_T), which is part of the input for **OSPS**, is $(3 \cdot a_0 + c, 2 \cdot a_0 + b_0 + c_0, 3 \cdot a_0 + b_0)$. We conclude that if we can find an oriented 2-simplex $(a', b', c') \neq (T, \tau_T)$ such that $N_2[\langle (a', b', c') \rangle] \neq 0$ then we can efficiently retrieve a source or sink v of G with $v = v(a', b', c')$ different from 0^k. $\square$

An instance of the **Strict Oriented Sperner Problem for the Surface** $\mathcal{S}_m^{(4)}$ **(SOSPS)** has the same input as an instance of **OSPS**. The output is an oriented 2-simplex $(T', \tau_{T'})$ of $\mathcal{S}_m^{(4)}$ such that $N_2[\langle (T', \tau_{T'}) \rangle] = -1$. The argument of Theorem 9 immediately implies that **SOSPS** is PPADS-complete.

We can construct a non-orientable surface $\mathcal{N}_m$ from the regular 12-subdivision of $\mathcal{S}_m^{(12)}$ of $\mathcal{S}_m$ by adding some further topological constructions, namely adding so-called *crosscaps*. The **Sperner Problem for the Surface** $\mathcal{N}_m$ **(SPS)** is the non-oriented analogue of **OSPS**. We can show that **SPS** is PPA-complete. The proof goes along the lines of the proof of Theorem 9. As we have a bigger and more complicated surface, the description of the labelling in the reduction is more complicated as well.

Acknowledgements

We would like to thank Éric Colin de Verdière for the modifications he suggested. The last two authors acknowledge the hospitality of the Isaac Newton Institute, Cambridge, where this work was completed.

References

[1] S. Aaronson. Quantum lower bound for the collision problem. In *Proc. of the 34th STOC*, pp. 635–642, 2002.
[2] S. Aaronson. Lower bounds for local search by quantum arguments. In *Proc. of 36th STOC*, pp. 465–474, 2004.
[3] A. Ambainis. Quantum lower bounds by quantum arguments. *J. Comput. Syst. Sci.*, 64(4):750–767, 2002.

[4] A. Ambainis. Polynomial degree vs. quantum query complexity. In *Proc. of 44th FOCS*, pp. 230–239, 2003.

[5] R. Beals, H. Buhrman, R. Cleve, M. Mosca and R. de Wolf. Quantum lower bounds by polynomials, *J. of the ACM* (48):4, pp. 778–797, 2001.

[6] P. Beame, S. Cook, J. Edmonds, R. Impagliazzo and T. Pitassi. The relative complexity of NP search problems. *J. Comput. System Sci.*, 57(1):3–19, 1998.

[7] C. Bennett, E. Bernstein, G. Brassard and U. Vazirani. Strength and weaknesses of quantum computing. *SIAM J. on Computing*, 26(5):1510–1523, 1997.

[8] E. Bloch. Mod 2 degree and a generalized no retraction Theorem. To appear in *Mathematische Nachrichten*.

[9] J. Buresh-Oppenheim and T. Morioka. Relativized NP search problems and propositional proof systems. In *Proc. of 19th Conference on Computational Complexity*, pp. 54–67, 2004.

[10] P. Crescenzi and R. Silvestri. Sperner's lemma and robust machines. *Comput. Complexity*, 7(2):163–173, 1998.

[11] D. Deutsch and R. Jozsa. Rapid solution of problems by quantum computation. *Proc. of the Royal Society A*, volume 439,1985.

[12] L. Goddyn, R. Bruce Richter and Jozef Širáň. Triangular embeddings of complete graphs from graceful labellings of paths. Preprint, 2004.

[13] K. Fan. Simplicial maps from an orientable n-pseudomanifold into S^m with the octahedral triangulation. *J. Combinatorial Theory*, 2:588–602, 1967.

[14] J. Gilbert, J. Hutchinson and R. Tarjan. A separator theorem for graphs of bounded genus. *J. Algorithms*, 5(3):391–407, 1984.

[15] M. Grigni. A Sperner lemma complete for PPA. *Inform. Process. Lett.*, 77(5-6):255–259, 2001.

[16] J. Kelner. Spectral partitioning, eigenvalue bounds, and circle packings for graphs of bounded genus. In *Proc. of 36th STOC*, pp. 455–464, 2004.

[17] S. Laplante and F. Magniez. Lower bounds for randomized and quantum query complexity using kolmogorov arguments. In *Proc of 19th Conference on Computational Complexity*, pp. 294–304, 2004.

[18] D. Llewellyn and C. Tovey. Dividing and conquering the square. *Discrete Appl. Math.*, 43(2):131–153, 1993.

[19] D. Llewellyn, C. Tovey and M. Trick. Local optimization on graphs. *Discrete Appl. Math.*, 23(2):157–178, 1989.

[20] N. Megiddo and C. Papadimitriou. On total functions, existence theorems and computational complexity. *Theoret. Comput. Sci.*, 81:317–324, 1991.

[21] B. Mohar and C. Thomassen. *Graphs on surfaces*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, 2001.

[22] J. Munkres. *Elements of algebraic topology*. Addison-Wesley Publishing Company, Menlo Park, CA, 1984.

[23] C. Papadimitriou. On graph-theoretic lemmata and complexity classes. *In Proc. of 31st FOCS*, pp. 794–801, 1990.

[24] C. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *J. Comput. System Sci.*, 48(3):498–532, 1994.

[25] M. Santha and M. Szegedy. Quantum and classical query complexities of local search are polynomially related. In *Proc. of 36th STOC*, pp. 494–501, 2004.

[26] Y. Shi. Quantum lower bounds for the collision and the element distinctness problems. In *Proc. of 43rd FOCS*, pp. 513–519, 2002.

[27] D. Simon. On the power of quantum computation. *SIAM J. on Computing* (26):5, pp. 1474–1783, 1997.

[28] E. Sperner. Neuer Beweis für die Invarianz der Dimensionzahl und des Gebietes. *Abh. Math. Sem. Hamburg Univ.* 6:265–272, 1928.

[29] R. Špalek and M. Szegedy. All quantum adversary methods are equivalent. `http://xxx.lanl.gov/abs/quant-ph/0409116`.

[30] L. Taylor. Sperner's Lemma, Brouwer's Fixed Point Theorem, The Fundamental Theorem of Algebra. `http://www.cs.csubak.edu/˜larry/sperner.pdf`.

[31] A. Thomason. Hamilton cycles and uniquely edge colourable graphs. *Ann. Discrete Math.* 3: 259–268, 1978.