2D-TUCKER is PPAD-complete

Dömötör Pálvölgyi

Ecole Polytechnique Fédérale de Lausanne

What is TUCKER?

What is TUCKER?

TUCKER is a search problem, can be considered the discrete version of the BORSUK-ULAM problem.

What is TUCKER?

TUCKER is a search problem, can be considered the discrete version of the BORSUK-ULAM problem.

Theorem (Borsuk-Ulam)

If we continuously map the sphere to the plane, then there is a point whose image is the same as the image of the opposite point.

What is TUCKER?

TUCKER is a search problem, can be considered the discrete version of the BORSUK-ULAM problem.

Theorem (Borsuk-Ulam)

If we continuously map the sphere to the plane, then there is a point whose image is the same as the image of the opposite point.

But how hard is it to find such a point?

What is TUCKER?

TUCKER is a search problem, can be considered the discrete version of the BORSUK-ULAM problem.

Theorem (Borsuk-Ulam)

If we continuously map the sphere to the plane, then there is a point whose image is the same as the image of the opposite point.

But how hard is it to find such a point?

We show that it is as hard as some other, well-known problems.

What is TUCKER?

TUCKER is a search problem, can be considered the discrete version of the BORSUK-ULAM problem.

Theorem (Borsuk-Ulam)

If we continuously map the sphere to the plane, then there is a point whose image is the same as the image of the opposite point.

But how hard is it to find such a point?

We show that it is as hard as some other, well-known problems.

This means that TUCKER is **PPAD – complete**.

What is PPAD?

What is PPAD?

PPAD is a complexity class like the much more famous **NP**.

What is PPAD?

PPAD is a complexity class like the much more famous **NP**.

It is defined as the problems reducible to the following problem called END-OF-THE-LINE.

What is PPAD?

PPAD is a complexity class like the much more famous **NP**.

It is defined as the problems reducible to the following problem called END-OF-THE-LINE.

Suppose we are given a directed graph that consists of directed paths and cycles.

What is PPAD?

PPAD is a complexity class like the much more famous **NP**.

It is defined as the problems reducible to the following problem called END-OF-THE-LINE.

Suppose we are given a directed graph that consists of directed paths and cycles.

If somebody shows us a vertex which is the end of a path, another end must exist.

What is PPAD?

PPAD is a complexity class like the much more famous **NP**.

It is defined as the problems reducible to the following problem called END-OF-THE-LINE.

Suppose we are given a directed graph that consists of directed paths and cycles.

If somebody shows us a vertex which is the end of a path, another end must exist.

Suppose the vertices are the 0–1 strings of length n, 0^n has only one neighbor and the neighborhood of a vertex is decided by a Turing-machine running in polynomial time.

What is PPAD?

PPAD is a complexity class like the much more famous **NP**.

It is defined as the problems reducible to the following problem called END-OF-THE-LINE.

Suppose we are given a directed graph that consists of directed paths and cycles.

If somebody shows us a vertex which is the end of a path, another end must exist.

Suppose the vertices are the 0–1 strings of length n, 0^n has only one neighbor and the neighborhood of a vertex is decided by a Turing-machine running in polynomial time.

The problem for which the input is n and the description of the Turing-machine and the output is another path ending is called END-OF-THE-LINE.

What is complete?

What is complete?

From Wiktionary:

complete (comparative more complete, superlative most complete)

1. With everything included.

It was a complete shock when he turned up on my doorstep.

2. Finished; ended; concluded; completed; as, the edifice is complete.

3. (analysis, of a metric space) in which every Cauchy sequence converges

4. (algebra, of a lattice) in which every set with a lower bound has a greatest lower bound

What is complete?

From Wiktionary:

complete (comparative more complete, superlative most complete)

1. With everything included.

It was a complete shock when he turned up on my doorstep.

2. Finished; ended; concluded; completed; as, the edifice is complete.

3. (analysis, of a metric space) in which every Cauchy sequence converges

4. (algebra, of a lattice) in which every set with a lower bound has a greatest lower bound

Of course we mean none of these but the complexity theoretic meaning defined with reductions.

What is complete?

From Wiktionary:

complete (comparative more complete, superlative most complete)

1. With everything included.

It was a complete shock when he turned up on my doorstep.

2. Finished; ended; concluded; completed; as, the edifice is complete.

3. (analysis, of a metric space) in which every Cauchy sequence converges

4. (algebra, of a lattice) in which every set with a lower bound has a greatest lower bound

Of course we mean none of these but the complexity theoretic meaning defined with reductions.
PPAD – complete means the problem is in **PPAD** and every problem in **PPAD** is reducible to the problem.

Why PPAD?

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class.

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all.

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE

BROUWER, SPERNER, EXCHANGE ECONOMY - by Papadimitriou

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE

BROUWER, SPERNER, EXCHANGE ECONOMY - by Papadimitriou

NASH - Daskalakis, Goldberg, Papadimitriou and Chen, Deng

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE

BROUWER, SPERNER, EXCHANGE ECONOMY - by Papadimitriou

NASH - Daskalakis, Goldberg, Papadimitriou and Chen, Deng

LEONTIEF EXCHANGE ECONOMY - Codenotti, Saberi, Varadarajan, Ye

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE

BROUWER, SPERNER, EXCHANGE ECONOMY - by Papadimitriou

NASH - Daskalakis, Goldberg, Papadimitriou and Chen, Deng

LEONTIEF EXCHANGE ECONOMY - Codenotti, Saberi, Varadarajan, Ye

ARROW-DEBREU EQUILIBRIA - Chen, Dai, Du, Teng

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE

BROUWER, SPERNER, EXCHANGE ECONOMY - by Papadimitriou

NASH - Daskalakis, Goldberg, Papadimitriou and Chen, Deng

LEONTIEF EXCHANGE ECONOMY - Codenotti, Saberi, Varadarajan, Ye

ARROW-DEBREU EQUILIBRIA - Chen, Dai, Du, Teng

PERSONALIZED EQUILIBRIA - Kintali et. al.

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE

BROUWER, SPERNER, EXCHANGE ECONOMY - by Papadimitriou

NASH - Daskalakis, Goldberg, Papadimitriou and Chen, Deng

LEONTIEF EXCHANGE ECONOMY - Codenotti, Saberi, Varadarajan, Ye

ARROW-DEBREU EQUILIBRIA - Chen, Dai, Du, Teng

PERSONALIZED EQUILIBRIA - Kintali et. al.

FISHER'S EQUILIBRIA - Chen, Teng

Why PPAD?

PPAD has several other variants that could be interesting, however, many famous problems are complete for this class. This means that if we can solve any of them fast, we can solve all. Unfortunately no one can solve any of these faster than exponential time, so we consider these problems hard.

PPAD – complete problems

END-OF-THE-LINE
BROUWER, SPERNER, EXCHANGE ECONOMY - by Papadimitriou
NASH - Daskalakis, Goldberg, Papadimitriou and Chen, Deng
LEONTIEF EXCHANGE ECONOMY - Codenotti, Saberi, Varadarajan, Ye
ARROW-DEBREU EQUILIBRIA - Chen, Dai, Du, Teng
PERSONALIZED EQUILIBRIA - Kintali et. al.
FISHER'S EQUILIBRIA - Chen, Teng

List maintained by Shiva Kintali.

Tucker's lemma

Tucker's lemma

Theorem (Tucker's lemma)

For any triangulation of the n-dimensional ball that is antipodally symmetric on the boundary and any coloring of the vertices with $+1, -1, +2, \ldots, +n, -n$ *that is also antipodal on the boundary, we have two vertices that are colored by opposite numbers.*

Tucker's lemma

Theorem (Tucker's lemma)

For any triangulation of the n-dimensional ball that is antipodally symmetric on the boundary and any coloring of the vertices with $+1, -1, +2, \ldots, +n, -n$ *that is also antipodal on the boundary, we have two vertices that are colored by opposite numbers.*

Definition (2D-TUCKER)

We are given the description of a deterministic Turing machine M *which for every input* $(u, v) \in \{0,1\}^{2n}$ *returns either* 1, -1, 2 *or* -2 *in time* $poly(n)$. *Furthermore, for all* i $M(0, i) = -M(2^n, 2^n - i)$ *and* $M(i, 0) = -M(2^n - i, 2^n)$. *The output (whose existence is guaranteed by Tucker's lemma) is* $(u, v) \in \{0,1\}^{2n}$ *and* $(u', v') \in \{0,1\}^{2n}$ *for which* $|u - u'| \leq 1$, $|v - v'| \leq 1$ *and* $M(u, v) = -M(u', v')$.

Tucker's lemma

Theorem (Tucker's lemma)

For any triangulation of the n-dimensional ball that is antipodally symmetric on the boundary and any coloring of the vertices with $+1, -1, +2, \ldots, +n, -n$ that is also antipodal on the boundary, we have two vertices that are colored by opposite numbers.

Definition (2D-TUCKER)

We are given the description of a deterministic Turing machine M which for every input $(u, v) \in \{0,1\}^{2n}$ returns either 1, -1, 2 or -2 in time poly(n). Furthermore, for all i $M(0, i) = -M(2^n, 2^n - i)$ and $M(i, 0) = -M(2^n - i, 2^n)$. The output (whose existence is guaranteed by Tucker's lemma) is $(u, v) \in \{0,1\}^{2n}$ and $(u', v') \in \{0,1\}^{2n}$ for which $|u - u'| \leq 1$, $|v - v'| \leq 1$ and $M(u, v) = -M(u', v')$.

Note that here instead of triangles we use squares.

TUCKER $\in$ PPAD

TUCKER $\in$ PPAD

Membership of TUCKER and BORSUK-ULAM was proved in 1994 by Papadimitriou in the paper where he defined **PPAD**.

TUCKER $\in$ PPAD

Membership of TUCKER and BORSUK-ULAM was proved in 1994 by Papadimitriou in the paper where he defined **PPAD**.

It easily follows from the known proofs for Tucker's lemma.

3D-TUCKER is PPAD-complete

3D-TUCKER is PPAD-complete

The hardness of TUCKER and BORSUK-ULAM was also proved by Papadimitriou for dimensions 3 and higher.

3D-TUCKER is PPAD-complete

The hardness of TUCKER and BORSUK-ULAM was also proved by Papadimitriou for dimensions 3 and higher.

The reduction is from END-OF-THE-LINE.

3D-TUCKER is PPAD-complete

The hardness of TUCKER and BORSUK-ULAM was also proved by Papadimitriou for dimensions 3 and higher.

The reduction is from END-OF-THE-LINE.

Chen and Deng proved that 2D-SPERNER is **PPAD – complete**.

3D-TUCKER is PPAD-complete

The hardness of TUCKER and BORSUK-ULAM was also proved by Papadimitriou for dimensions 3 and higher.

The reduction is from END-OF-THE-LINE.

Chen and Deng proved that 2D-SPERNER is **PPAD – complete**.

We will use a method similar to theirs.

2D-TUCKER is PPAD-complete

2D-TUCKER is PPAD-complete

Membership was proved by Papadimitriou.

2D-TUCKER is PPAD-complete

Membership was proved by Papadimitriou.

Hardness will be a reduction to END-OF-THE-LINE.

2D-TUCKER is PPAD-complete

Membership was proved by Papadimitriou.

Hardness will be a reduction to END-OF-THE-LINE.

This means that we want to transform a graph into an appropriate coloring of the square.

2D-TUCKER is PPAD-complete

Membership was proved by Papadimitriou.

Hardness will be a reduction to END-OF-THE-LINE.

This means that we want to transform a graph into an appropriate coloring of the square.

The color of every vertex in the square will be determined by a finite number of queries to the Turing-machine that determines the graph.

2D-TUCKER is PPAD-complete

Membership was proved by Papadimitriou.

Hardness will be a reduction to END-OF-THE-LINE.

This means that we want to transform a graph into an appropriate coloring of the square.

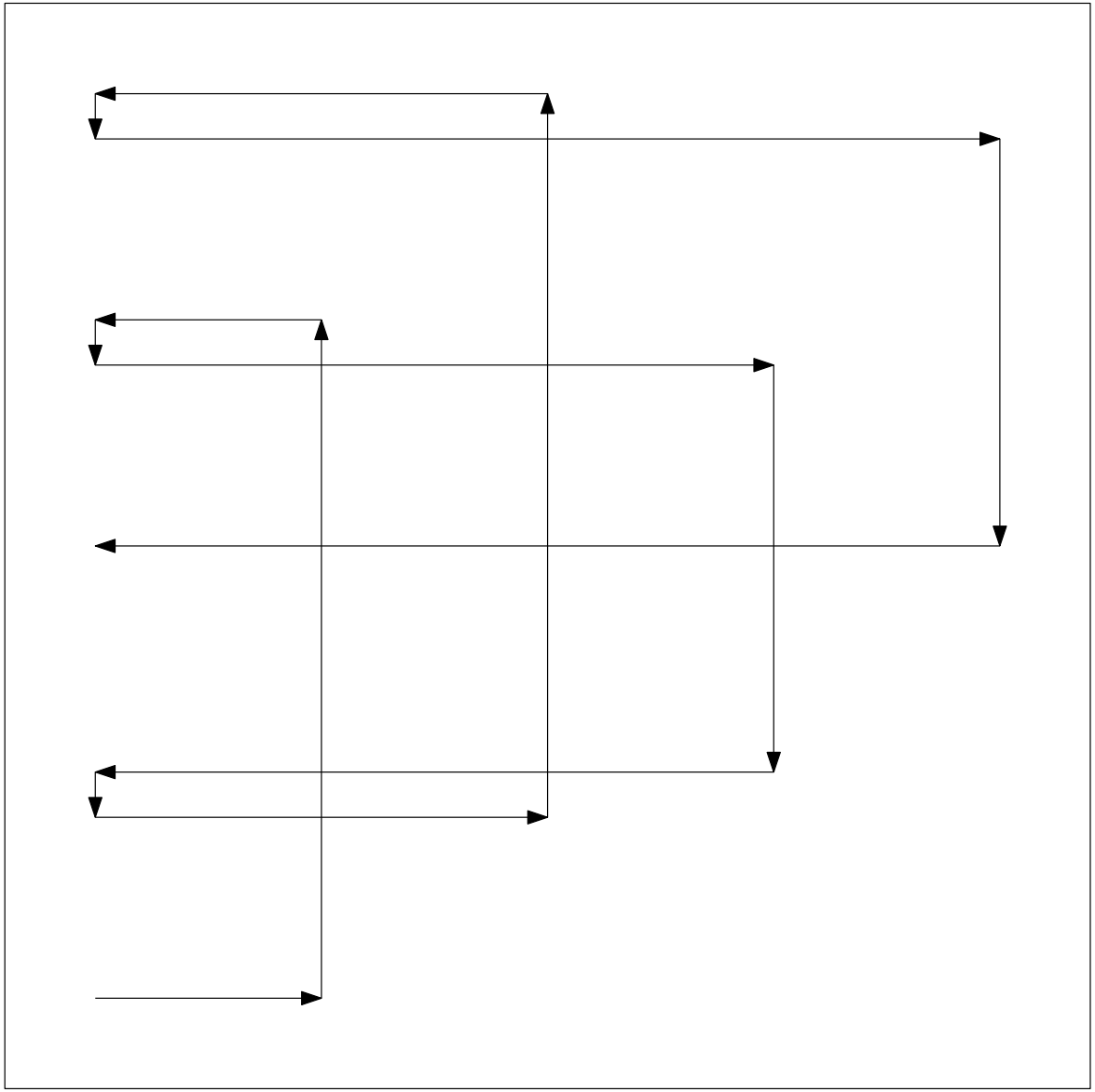
The color of every vertex in the square will be determined by a finite number of queries to the Turing-machine that determines the graph.

Finding two neighbors with opposite colors will give a path end.

2D-TUCKER is PPAD-complete

-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	2
-1	2	2	2	2	2	2	2	2	2	2	2
-1	2	1	1	1	1	1	1	1	1	1	1
-1	2	1									1
-1	2	1									1
-1	2	1		2	2	2	2				1
-1	2	1		2	-1	-1	-1				1
-1	2	1	1	2	-1	-2	-2				1
-1	2	2	2	2	-1	-2					1
-1	-1	-1	-1	-1	-1	-2					1
-2	-2	-2	-2	-2	-2	-2					1
-2	1	1	1	1	1	1	1	1	1	1	1

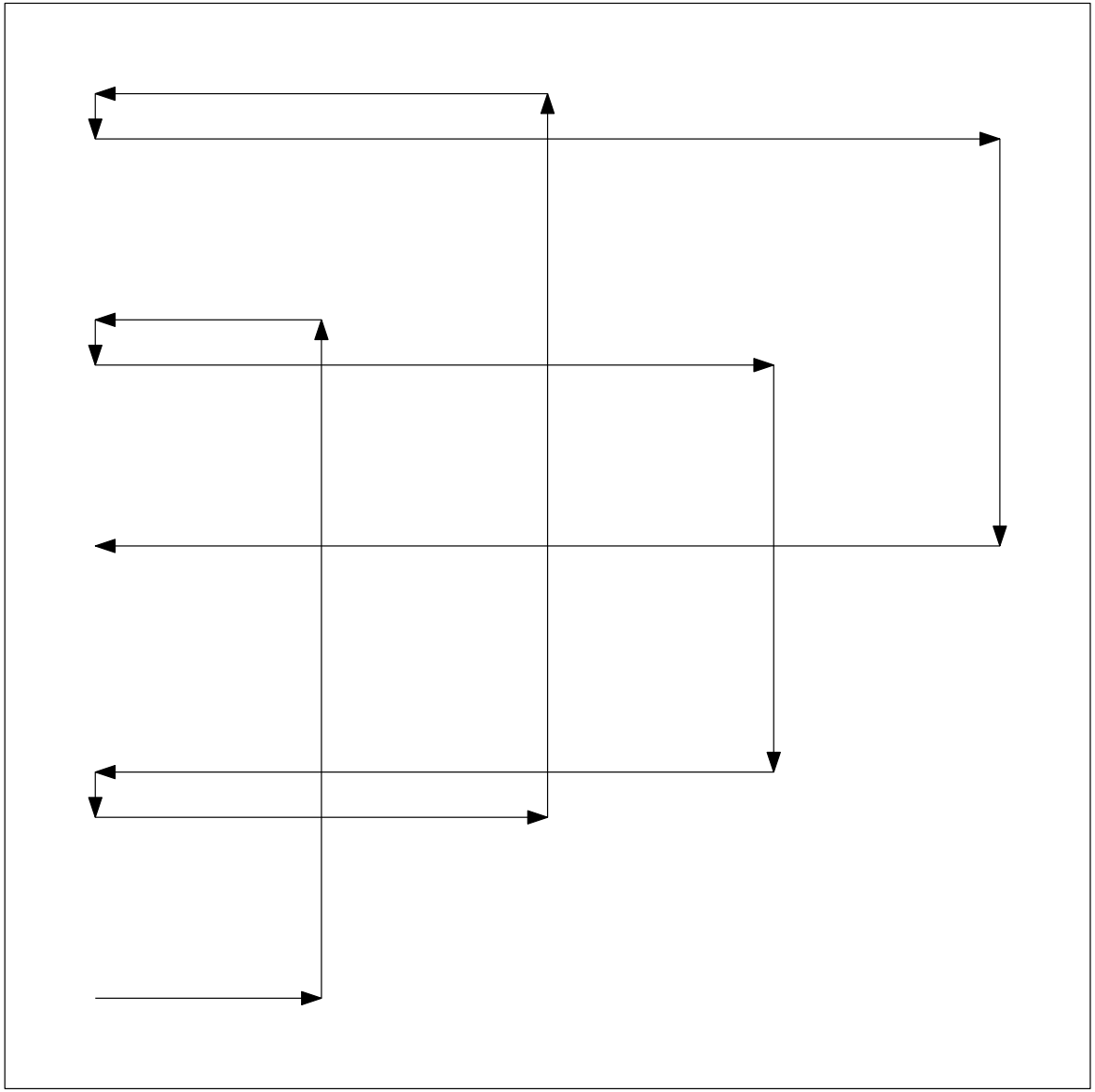
2D-TUCKER is PPAD-complete



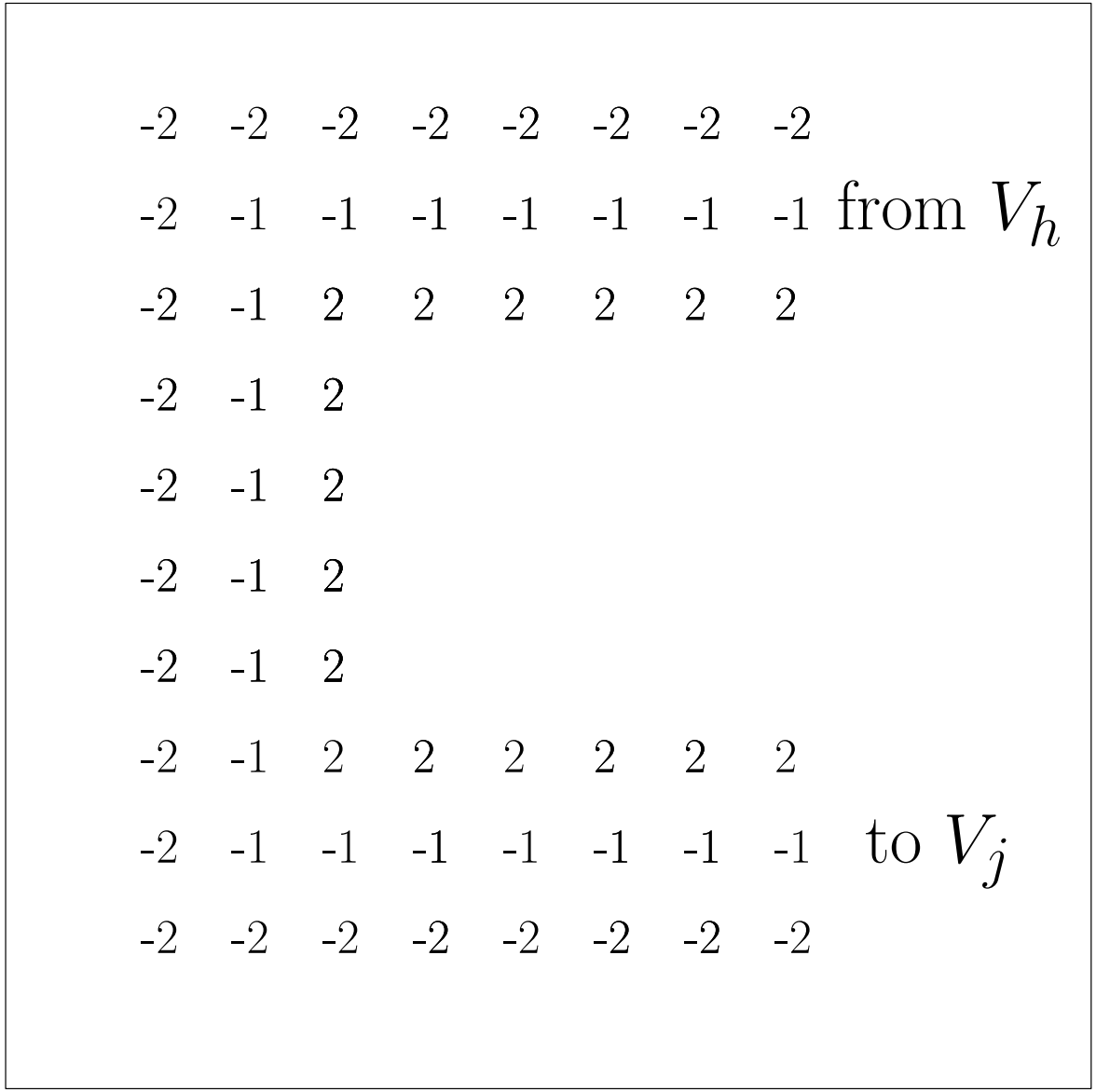
2D-TUCKER is PPAD-complete

	-2	-2	-2	-2	-2	-2	-2	-2
V_j	-1	-1	-1	-1	-1	-1	-1	-2
	2	2	2	2	2	2	-1	-2
						2	-1	-2
						2	-1	-2
						2	-1	-2
						2	-1	-2
	2	2	2	2	2	2	-1	-2
V_i	-1	-1	-1	-1	-1	-1	-1	-2
	-2	-2	-2	-2	-2	-2	-2	-2

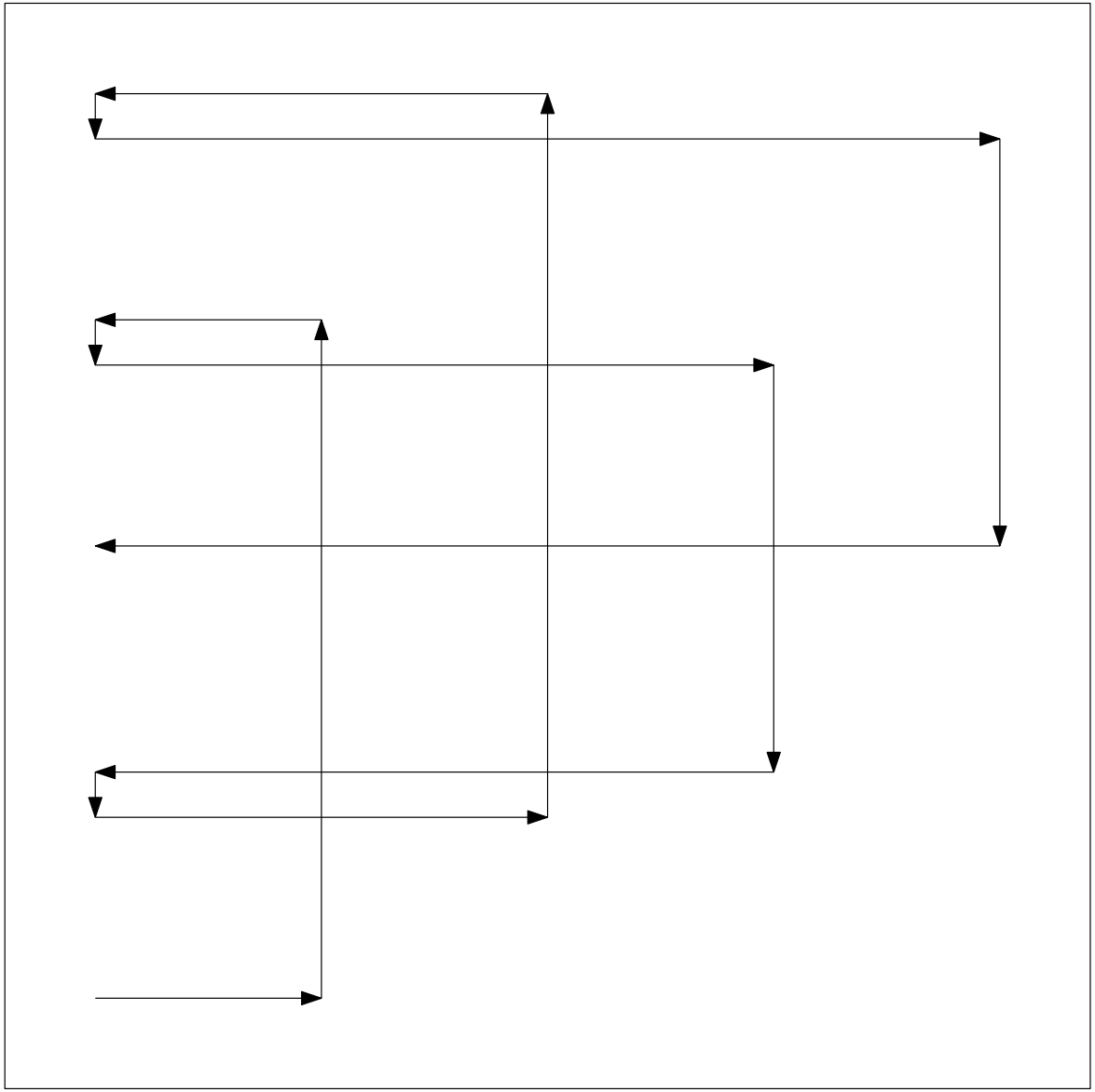
2D-TUCKER is PPAD-complete



2D-TUCKER is PPAD-complete



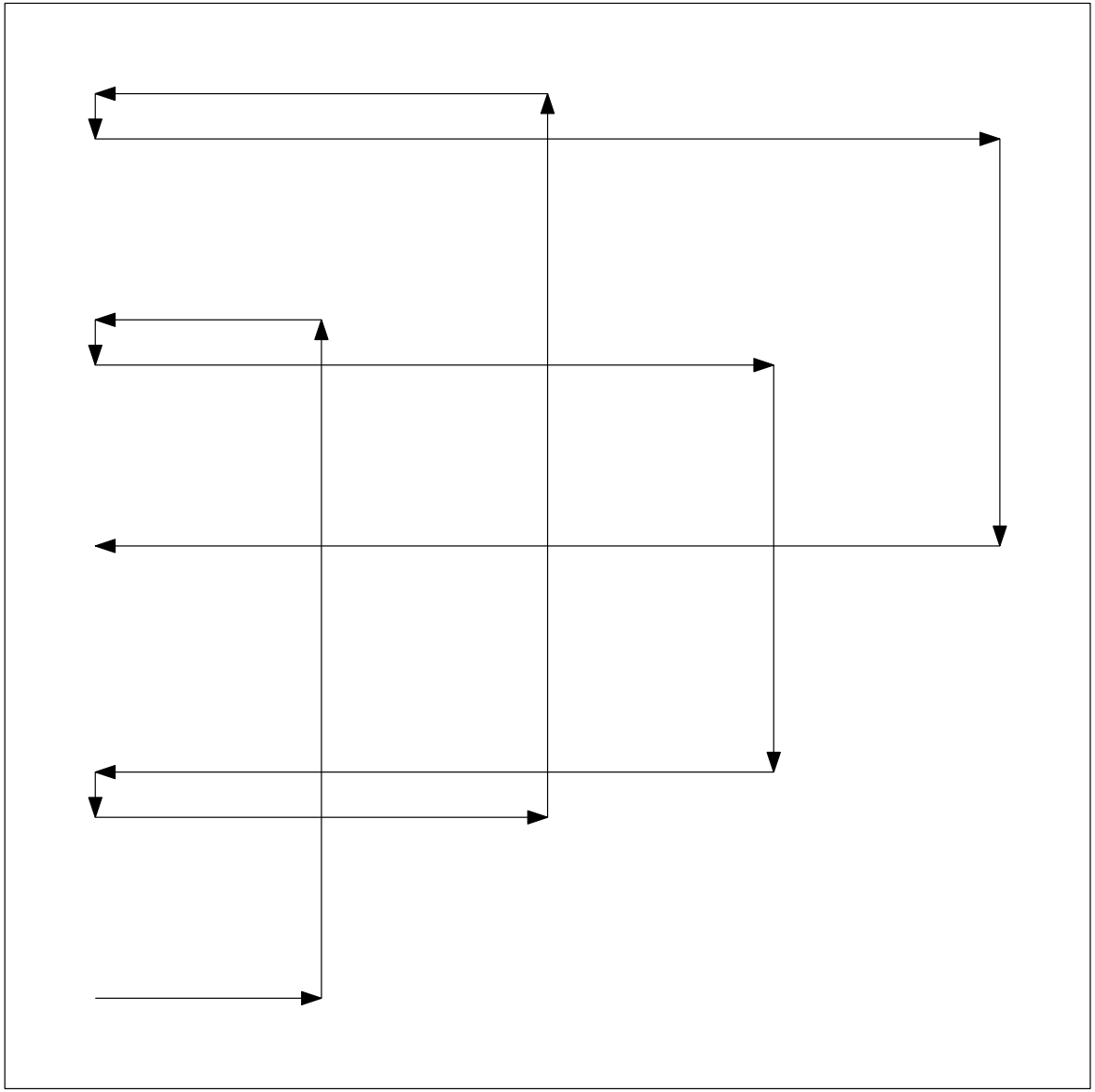
2D-TUCKER is PPAD-complete



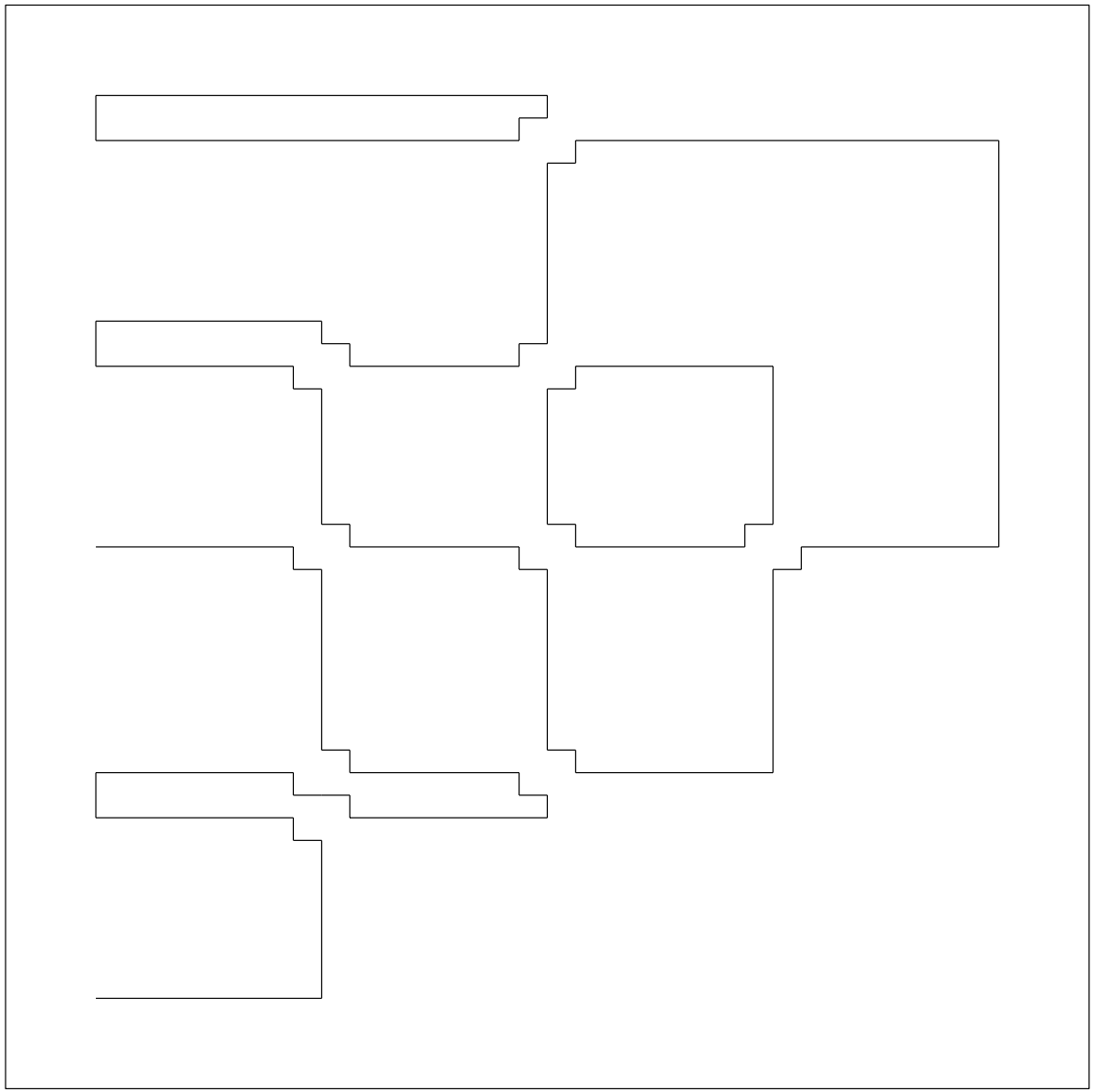
2D-TUCKER is PPAD-complete

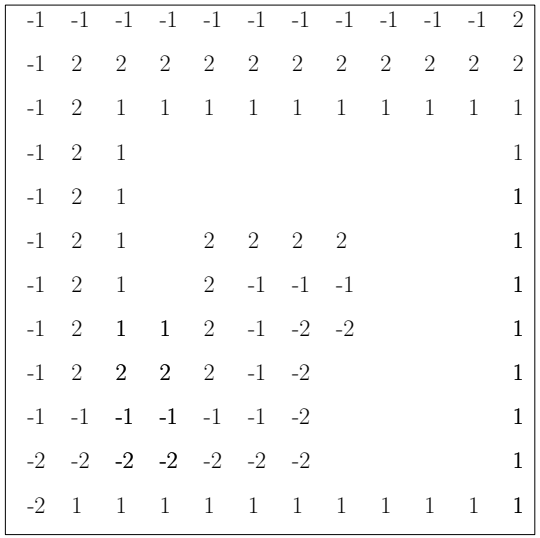
				-2	-1	2	2	2			
				-2	-1	-1	-1	2			
				-2	-2	-2	-1	2			
						-2	-1	2			
2	2	2	2	2		-2	-1	2	2	2	2
-1	-1	-1	-1	2		-2	-1	-1	-1	-1	-1
-2	-2	-2	-1	2		-2	-2	-2	-2	-2	-2
		-2	-1	2							
		-2	-1	2	2	2					
		-2	-1	-1	-1	2					
		-2	-2	-2	-1	2					
				-2	-1	2					

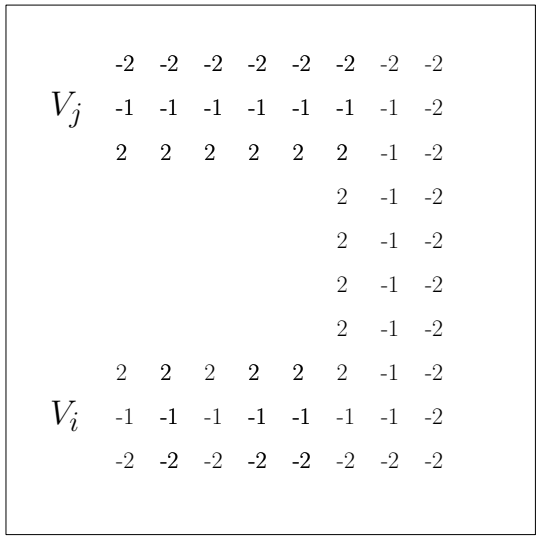
2D-TUCKER is PPAD-complete

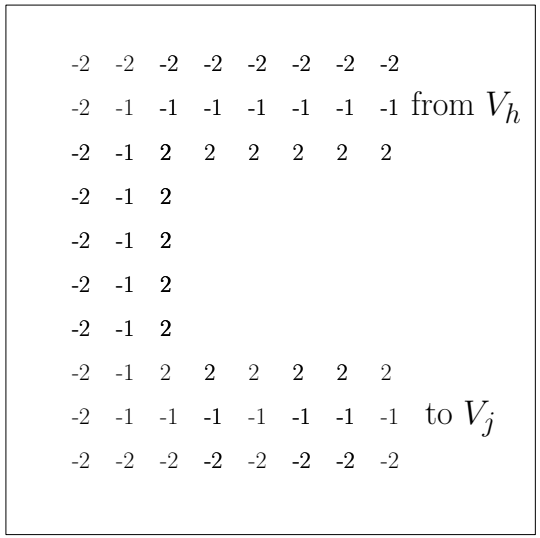


2D-TUCKER is PPAD-complete









Thank you for your attention!