

Theory Comput Syst (2015) 56:394–405
DOI 10.1007/s00224-014-9558-4

Sitting Closer to Friends than Enemies, Revisited

Marek Cygan · Marcin Pilipczuk · Michał
Pilipczuk · Jakub Onufry Wojtaszczyk

Published online: 1 July 2014

© The Author(s) 2014. This article is published with open access at Springerlink.com

Abstract Signed graphs, i.e., undirected graphs with edges labelled with a plus or minus sign, are commonly used to model relationships in social networks. Recently, Kermarrec and Thraves (2011) initiated the study of the problem of appropriately visualising the network: They asked whether any signed graph can be embedded into the metric space \mathbb{R}^l in such a manner that every vertex is closer to all its friends (neighbours via positive edges) than to all its enemies (neighbours via negative edges). Interestingly, embeddability into \mathbb{R}^1 can be expressed as a purely combinatorial problem. In this paper we pursue a deeper study of this case, answering several questions posed by Kermarrec and Thraves. First, we refine the approach of Kermarrec and Thraves for the case of complete signed graphs by showing that the problem is closely related to the recognition of proper interval graphs. Second, we prove that the general case, whose polynomial-time tractability remained open, is in fact NP -complete. Finally, we provide lower and upper bounds for the time complexity of the general case: we prove that the existence of a subexponential time (in the number of vertices and edges of the input signed graph) algorithm would violate the

A preliminary version of this paper has been presented at MFCS 2012 [5].

M. Cygan · M. Pilipczuk (✉)

Institute of Informatics, University of Warsaw, Warsaw, Poland

e-mail: malcin@mimuw.edu.pl

M. Cygan

e-mail: cygan@mimuw.edu.pl

M. Pilipczuk

Department of Informatics, University of Bergen, Bergen, Norway

e-mail: michal.pilipczuk@ii.uib.no

J. O. Wojtaszczyk

Google Inc., Warsaw, Poland

e-mail: onufry@google.com

Exponential Time Hypothesis, whereas a simple dynamic programming approach gives a running time single-exponential in the number of vertices.

Keywords Signed graphs · Relationships · Embedding into metric space

1 Introduction

Undirected graphs with edges labelled positively (by a $+$) and negatively (by a $-$), called *signed graphs*, in many applications serve as a very simple model of relationships between a group of people, e.g., in a social network. Sign labels can express in a simplified way mutual relations, like staying in a relationship, family bonds or conflicts, by classifying them either as *friendship* ($+$ edge), *hostility* ($-$ edge) or *ambivalence* (no edge). In particular, much effort has been put into properly understanding and representing the structure of the network, balancing it or naturally partitioning into clusters [1, 3, 6, 15–18, 21]. One of the problems is to visualize the model graph properly, i.e., in such a way that positive relations tend to make vertices be placed close to each other, while negative relations imply large distances between vertices.

In their recent work, Kermarrec and Thraves [14] formalized this problem as follows: Consider the metric space \mathbb{R}^d with the Euclidean metric denoted by d . Given a signed graph G , is it possible to embed the vertices of G in \mathbb{R}^d so that for any positive edge uu_1 and negative edge uu_2 it holds that $d(u, u_1) < d(u, u_2)$? This question has a natural interpretation: we would like to place a group of people so that every person is placed closer to his friends than to his enemies.

The work of Kermarrec and Thraves [14] concentrated on showing a number of examples and counterexamples for embeddability into spaces of small dimensions (1 and 2) and a deeper study of the 1-dimensional case. Interestingly enough, the case of the Euclidean line has an equivalent formulation in the language of pure combinatorics: Given a signed graph G , is it possible to order the vertices of G so that for any positive edge uw there is no negative edge uv with v laying between u and w ? The authors made algorithmic use of this combinatorial insight: Providing the given signed graph is complete (i.e., every pair of vertices is adjacent via a positive or negative edge) they show a polynomial-time algorithm that computes an embedding into a line or reports that no such embedding exists.

Kermarrec and Thraves also posed a number of open problems in the area, including the question of the complexity of determining the embeddability of an arbitrary (not necessarily complete) graph into the Euclidean line.

Our Results We focus on the problem of embedding a signed graph into a line. The reformulation of the 1-dimensional case of Kermarrec and Thraves turns out to be an interesting combinatorial problem, which allows classical methods of analysis and shows interesting links with the class of proper interval graphs.

We begin with refining the result of Kermarrec and Thraves for the case of complete graphs. We prove that a complete signed graph is embeddable into a line if and only if the graph formed by the positive edges is a proper interval graph.

Using this theorem one can immediately transfer all the results from the well-studied area of proper interval graphs into our setting. Most importantly, as recognition of proper interval graphs can be performed in linear-time [4], we obtain a simpler algorithm for determining the embeddability of a complete graph into a line, with a linear runtime.

We next analyse the general case. We resolve the open problem posed in [14] negatively: it is NP -complete to resolve whether a given signed graph can be embedded into a line. This hardness result also answers other questions of Kermarrec and Thraves [14]. For example, we infer that it is NP -hard to decide the smallest dimension of a Euclidean space in which the graph can be embedded, as such an algorithm could be used to test embeddability into a line.

Furthermore, we are able to show a lower bound on the time complexity of resolving embeddability into a line, under a plausible complexity assumption. We prove that obtaining an algorithm running in subexponential time (in terms of the total number of vertices and edges of the input graph) would contradict the *Exponential Time Hypothesis* [11] (see Section 2 for an exact statement). We complete the picture of the complexity of the problem by showing a dynamic programming algorithm that runs in $O^*(2^n)$ time,¹ matching the aforementioned lower bound up to a constant in the base of the exponent (n denotes the number of vertices of the input graph).

Organisation of the Paper In Section 2 we recall widely known notions and facts that are of further use, and provide the details of the combinatorial reformulation of the problem by Kermarrec and Thraves [14]. Section 3 is devoted to refinements in the analysis of the case of the complete signed graphs, while Section 4 describes upper and lower bounds for the complexity of the general case. Finally, in Section 5 we gather conclusions and ideas for further work.

2 Preliminaries

Basic Definitions For a finite set V , by an *ordering* of V we mean a bijection $\pi : V \rightarrow \{1, 2, \dots, |V|\}$. We sometimes treat an ordering π as a linear order on V and for $u, v \in V$ we write $u \leq_\pi v$ to denote $\pi(u) \leq \pi(v)$. A *lexicographic ordering* imposed by π on pairs of elements from V is an ordering π' of $V \times V$ defined as follows: $(a, b) \leq_{\pi'} (c, d)$ if and only if $a <_\pi c$, or $a = c$ and $b \leq_\pi d$. If π is an ordering of vertices of a directed graph G , then we say that π is a *topological ordering* if and only if for every $(v, w) \in E(G)$ we have that $v \leq_\pi w$. A directed graph admits a topological ordering of vertices if and only if it is acyclic.

In a graph $G = (V, E)$ the *neighbourhood* of a vertex v , denoted $N(v)$, is the set of all its neighbours, i.e., $\{w : vw \in E\}$. The *closed neighbourhood* of v is defined as $N[v] = N(v) \cup \{v\}$.

¹The $O^*(\)$ notation suppresses factors that are polynomial in the input size.

A *signed graph* is a triple $G = (V, E^+, E^-)$, where $E^+, E^- \subseteq \binom{V}{2}$ and $E^+ \cap E^- = \emptyset$. We view a signed graph as an undirected simple graph with two possible labels on the edges: positive (+) and negative (-). We call the edges from E^+ positive, while those from E^- negative. The graph $G^+ = (V, E^+)$ is called the *positive part* of G , and $G^- = (V, E^-)$ — the *negative part*. A signed graph is called *complete* if $E^+ \cup E^- = \binom{V}{2}$, i.e., every pair of vertices is adjacent via a positive or negative edge.

Proper Interval Graphs Let $G = (V, E)$ be an undirected graph, \mathcal{I} be a family of size $|V|$ of intervals on real line with nonempty interiors and pairwise different endpoints and $\iota : V \rightarrow \mathcal{I}$ be any bijection. We say that \mathcal{I} is an *interval model* for G if for every $v, w \in V, v \neq w, vw \in E$ is equivalent to $\iota(v) \cap \iota(w) \neq \emptyset$. \mathcal{I} is a *proper interval model* if, additionally, none of the intervals is entirely contained in any other. Graphs having an interval model are called *interval graphs*, while if a proper interval model exists as well, we call them *proper interval graphs*.

Looges and Olariu showed the following combinatorial characterization of proper interval graphs. An ordering π of the vertex set of a graph $G = (V, E)$ is called an *umbrella ordering* if whenever $\pi(v_1) < \pi(v_2) < \pi(v_3)$, then $v_1v_3 \in E$ implies $v_1v_2 \in E$ and $v_2v_3 \in E$.

Theorem 2.1 ([19]) *A graph G is a proper interval graph if and only if G admits an umbrella ordering.*

Exponential Time Hypothesis [11]: The *Exponential Time Hypothesis* (ETH for short) asserts that there exists a constant $C > 0$ such that no algorithm solving the 3-CNF-SAT problem in $O(2^{Cn})$ exists, where n denotes the number of variables in the input formula.

Combinatorial Problem Statement In [14], Kermarrec and Thraves work with the metric definition of the problem: Given a signed graph $G = (V, E^+, E^-)$ a feasible embedding of G in the Euclidean space \mathbb{R}^l is such a function $f : V \rightarrow \mathbb{R}^l$ that for all u_1, u_2, u , if $u_1u \in E^+$ and $u_2u \in E^-$, then $d(f(u_1), f(u)) < d(f(u_2), f(u))$ (recall that d stands for the Euclidean distance in \mathbb{R}^l). However, for the 1-dimensional case they have in essence proved the following result:

Theorem 2.2 (Lemmata 3 and 4 of [14], rephrased) *A signed graph $G = (V, E^+, E^-)$ has a feasible embedding in a line iff there is an ordering π of V such that for every $u \in V$:*

- (i) *there are no $u_1 <_{\pi} u_2 <_{\pi} u$ such that $u_1u \in E^+$ and $u_2u \in E^-$;*
- (ii) *there are no $u_1 >_{\pi} u_2 >_{\pi} u$ such that $u_1u \in E^+$ and $u_2u \in E^-$.*

We will jointly call conditions (i) and (ii) *the condition imposed on u* . Somewhat abusing the notation, the ordering π will also be called *an embedding of G into*

the line. Therefore, from now on we are working with the following combinatorial problem that is equivalent to the version considered by Kermarrec and Thraves:

LINE CLUSTER EMBEDDING
Input: A signed graph $G = (V, E^+, E^-)$.
Task: Does there exist an ordering π on V such that for every $u \in V$: (i) there are no $u_1 <_\pi u_2 <_\pi u$ such that $u_1u \in E^+$ and $u_2u \in E^-$; (ii) there are no $u_1 >_\pi u_2 >_\pi u$ such that $u_1u \in E^+$ and $u_2u \in E^-$.

3 The Complete Signed Graph Case

In their work, Kermarrec and Thraves [14] announced a polynomial-time algorithm solving the LINE CLUSTER EMBEDDING problem in the case where the input signed graph is complete. Their line of reasoning was essentially as follows: if a signed graph can be embedded into a line, then its positive part has to be chordal. However, for a connected chordal graph with at least 4 vertices that actually is embeddable into a line, every *perfect elimination ordering* of the graph is a feasible solution. Therefore, having checked that the graph is chordal and computed a perfect elimination ordering of every connected component, we can simply verify whether the obtained ordering is a correct line embedding.

We refine the approach of Kermarrec and Thraves by showing the following simple observation (see also Fig. 1 for an illustration).

Lemma 3.1 *For a signed complete graph $G = (V, E^+, E^-)$, an ordering π of V is a feasible embedding of G into a line if and only if π is an umbrella ordering of the positive part of G .*

Proof If π is not a feasible embedding of G into a line, then there exists $u_1, u_2, u \in V$ such that $u_1u \in E^+$, $u_2u \in E^-$, and u_2 lies between u_1 and u in the ordering π . Consequently, u_1u is an edge of the positive part, but u_2u is not, hence π is not an umbrella ordering of G^+ .

In the other direction, if π is not an umbrella ordering of G^+ , then there exist $v_1 <_\pi v_2 <_\pi v_3$ such that $v_1v_2 \in E^+$ but $v_1v_3 \notin E^+$ or $v_3v_2 \notin E^+$. As G is a

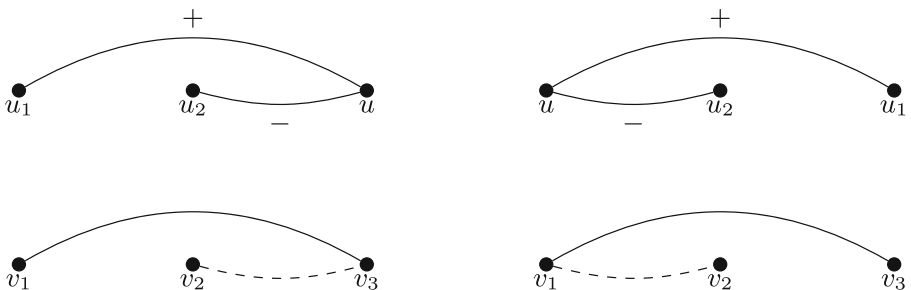


Fig. 1 A picture proof of Lemma 3.1. The *first row* presents forbidden situations in a feasible embedding of G into a line, whereas the *second row* presents forbidden situations in an umbrella ordering

complete graph, $v_1 v_3 \in E^-$ or $v_3 v_2 \in E^-$. In the first case, we have a violation of condition (ii) imposed on v_1 , and in the second case we have a violation of condition (i) imposed on v_3 . Consequently, π is not a feasible embedding of G into a line. \square

Corollary 3.2 *A complete signed graph $G = (V, E^+, E^-)$ is embeddible in \mathbb{R}^1 if and only if $G^+ = (V, E^+)$ is a proper interval graph.*

Recall that proper interval graphs are a subclass of chordal graphs; therefore, the result nicely fits into the picture of Kermarrec and Thraves. Moreover, the theory of proper interval graphs is well-studied, so many results from that area can be immediately translated to our setting. For instance, many NP-complete problems become solvable in polynomial time on proper interval graphs (e.g., [2, 10, 13, 20]), and the linear-time algorithm of Cornil et al. [4] for recognizing proper interval graphs immediately solves the LINE CLUSTER EMBEDDING problem in linear time in case of a complete signed graph.

Corollary 3.3 *Assuming the input graph is complete and given as the set of positive edges, LINE CLUSTER EMBEDDING can be solved in $O(|V| + |E^+|)$ time complexity. Moreover, the algorithm can produce a feasible ordering of the vertices in the same time, if such an ordering exists.*

4 The General Case

4.1 NP-Completeness of the General Case

In [14] Kermarrec and Thraves asked whether the LINE CLUSTER EMBEDDING problem is also polynomial-time solvable in the case where the input is not restricted to complete graphs. In this section we show that this is unlikely: in fact, the problem becomes NP-complete.

In the proof we use an auxiliary problem, called ACYCLIC PARTITION.

ACYCLIC PARTITION

Input: A directed graph $D = (V, A)$.

Task: Is it possible to partition V into two sets V_1 and V_2 , so that both $D[V_1]$ and $D[V_2]$ are directed acyclic graphs (DAGs)?

ACYCLIC PARTITION has been introduced and proven to be NP-complete by Eppstein and Mumford [7] and, independently, by Guillemot et al. [9]. In both these papers it was used as a pivot problem for proving NP-completeness of other problems. For the sake of completeness we would like to revisit the NP-hardness proof, because we need explicit bounds on the size of the directed graph obtained in the reduction for further applications. We begin with an instance of the SET SPLITTING problem, which is known to be NP-complete [8].

SET SPLITTING

Input: A set system (\mathcal{F}, U) , where $\mathcal{F} \subseteq 2^U$.

Task: Does there exist a subset $X \subseteq U$ such that each set in \mathcal{F} contains both an element from X and an element from $U \setminus X$?

Lemma 4.1 *There exists a polynomial-time algorithm that given an instance (\mathcal{F}, U) of SET SPLITTING outputs an equivalent instance $G = (V, A)$ of ACYCLIC PARTITION, for which $|V| = |U| + \sum_{F \in \mathcal{F}} |F|$ and $|A| = 3 \sum_{F \in \mathcal{F}} |F|$.*

Proof We construct the directed graph $D = (V, A)$ as follows. For every set $F \in \mathcal{F}$ and every $u \in F$ we build a vertex c_u^F and connect all the vertices corresponding to the same set F into a directed cycle in any order. For every element $u \in U$ we build a vertex d_u and for every vertex of the form c_u^F we introduce two arcs: (d_u, c_u^F) and (c_u^F, d_u) . This concludes the construction; it is easy to verify the claimed sizes of V and A .

Let us formally prove that the instances are equivalent. Let X be any solution to the (\mathcal{F}, U) instance of SET SPLITTING. Let $V_1 = \{d_u : u \in X\} \cup \{c_u^F : u \in U \setminus X\}$ and $V_2 = \{d_u : u \in U \setminus X\} \cup \{c_u^F : u \in X\}$. As X splits every set $F \in \mathcal{F}$, none of the cycles formed by vertices c_u^F for fixed F is entirely contained in either V_1 or V_2 . Also, for every element u the vertex d_u becomes isolated in the corresponding graph $D[V_i]$, as all his neighbours belong to V_{3-i} . Therefore, both $D[V_1]$ and $D[V_2]$ are collections of isolated vertices and directed paths and (V_1, V_2) is a solution to the ACYCLIC PARTITION instance.

In the other direction, let (V_1, V_2) be a solution to the instance of ACYCLIC PARTITION. Let $X = \{u : d_u \in V_1\} \subseteq U$; we claim that X is a solution to the instance of SET SPLITTING. Note that, due to the 2-cycles on the vertices d_u and c_u^F , we have that for every $u \in X$, all vertices c_u^F belong to V_2 , whereas for every $u \notin X$, all vertices c_u^F belong to V_1 .

Take any $F \in \mathcal{F}$. As the cycle formed by vertices c_u^F is not entirely contained in any of the graphs $D[V_1], D[V_2]$, there exist some u_1 such that $c_{u_1}^F \in V_1$ and u_2 such that $c_{u_2}^F \in V_2$. As the cycles formed by pairs $\{d_{u_1}, c_{u_1}^F\}$ and $\{d_{u_2}, c_{u_2}^F\}$ are also not entirely contained in $D[V_1]$ nor in $D[V_2]$, $d_{u_1} \in V_2$ and $d_{u_2} \in V_1$. Consequently, $u_1 \in U \setminus X$, $u_2 \in X$ and F is split. \square

Lemma 4.2 *There exists a polynomial-time algorithm that given an instance $D = (V, A)$ of ACYCLIC PARTITION outputs an equivalent instance $H = (V', E^+, E^-)$ of LINE CLUSTER EMBEDDING, such that $|V'| = |V| + |A| + 1$, $|E^+| = 2|A|$ and $|E^-| = |A| + |V|$.*

Proof We construct the graph H as follows: The set of vertices, V' , consists of:

- a *special* vertex s ;
- for every $e \in A$, a *checker* vertex c_e ;
- for every $v \in V$, an *alignment* vertex a_v .

We construct the edges of the signed graph as follows:

- for every $e \in A$, we introduce a positive edge sc_e ;
- for every $v \in V$, we introduce a negative edge sa_v ;
- for every arc $(v, w) \in A$, we introduce a positive edge $c_{(v,w)}a_v$ and a negative edge $c_{(v,w)}a_w$.

This concludes the construction; it is easy to verify the claimed sizes of V' , E^+ , E^- .

Let us prove equivalence of the instances. Let π , an ordering of V' , be a solution of the LINE CLUSTER EMBEDDING instance (V', E^+, E^-) . As the special vertex s is adjacent via positive edges to all the checker vertices, and via negative edges to all the other, alignment, vertices, in the ordering π the checker vertices together with the special vertex have to form an interval, i.e., a set of consecutive elements with respect to π . Let V_1 be the set of those $v \in V$ for which a_v is to the left of this interval, whereas V_2 is the set of those $v \in V$ for which a_v is to the right of this interval. Formally, $V_1 = \{v \in V : a_v \leq_{\pi} s\}$ and $V_2 = \{v \in V : a_v \geq_{\pi} s\}$. We claim that (V_1, V_2) is a feasible solution of the ACYCLIC PARTITION instance (V, A) . Consider any arc (v, w) such that $v, w \in V_1$. As $a_v \leq_{\pi} c_{(v,w)}$, $a_w \leq_{\pi} c_{(v,w)}$, $c_{(v,w)}a_v \in E^+$ and $c_{(v,w)}a_w \in E^-$, then it follows that $a_w \leq_{\pi} a_v$. Thus, π has to induce a reverse topological ordering on the vertices of $D[V_1]$ and, therefore, $D[V_1]$ has to be acyclic. Symmetrically, $D[V_2]$ has to be acyclic as well, which concludes the proof of (V_1, V_2) being a feasible solution.

Now take any solution (V_1, V_2) of ACYCLIC PARTITION instance (V, A) . Let π_1 be any topological ordering of $D[V_1]$ and π_2 be any topological ordering of $D[V_2]$, by which we mean that if (u, v) is an arc of $D[V_1]$, $\pi_1(u) < \pi_1(v)$, and the same holds for π_2 . Let us construct an ordering π of V' as follows:

- first, place all the vertices a_v for $v \in V_1$ in the reverse order induced by π_1 ;
- then, place all the checker vertices $c_{(v,w)}$ for which $v \in V_1$ and $w \in V_2$, in any order;
- then, place all the checker vertices $c_{(v,w)}$ for which $v, w \in V_1$, in reverse lexicographic order imposed by π_1 on pairs (v, w) ;
- then, place the special vertex s ;
- then, place all the checker vertices $c_{(v,w)}$ for which $v, w \in V_2$, in lexicographic order imposed by π_2 on pairs (v, w) ;
- then, place all the checker vertices $c_{(v,w)}$ for which $v \in V_2$ and $w \in V_1$, in any order;
- finally, place all the vertices a_v for $v \in V_2$ in the order induced by π_2 .

We claim that such π is a feasible solution to LINE CLUSTER EMBEDDING instance (V', E^+, E^-) .

Note that the positive neighbours of the special vertex s form an interval, therefore the condition imposed on this vertex is satisfied. Now consider a checker vertex $c_{(v,w)}$. If v, w belong to different sets V_1, V_2 , then the only negative neighbour of $c_{(v,w)}$ is the first or the last of his closed neighbourhood with respect to π , thus satisfying the condition imposed on $c_{(v,w)}$. In case when $v, w \in V_1$ or $v, w \in V_2$ this is also true, due to π_1, π_2 being topological orderings of $D[V_1], D[V_2]$ respectively.

Now take any vertex a_v , by symmetry assume $v \in V_1$. We need to prove that the condition imposed on a_v is satisfied as well. The neighbours of v consist of:

1. positive neighbours $c_{(v,v')}$, such that $v' \in V_2$;
2. positive neighbours $c_{(v,v')}$, such that $v' \in V_1$;
3. negative neighbours $c_{(v',v)}$, such that $v' \in V_1$;
4. negative neighbours $c_{(v',v)}$, such that $v' \in V_2$.

We now verify that by the construction of π the neighbours of a_v lie in this very order with respect to π . Clearly, the order in which we placed the checkers in π ensures that the neighbours from (1) are placed before the neighbours from (2) and that the neighbours from (3) are placed before the neighbours from (4). Thus the only non-trivial check is whether the vertices from (2) lie before the vertices from (3). Assume otherwise, that there are some v'_1, v'_2 such that $(v, v'_1) \in A$, $(v'_2, v) \in A$, but $c_{(v, v'_1)} >_{\pi} c_{(v'_2, v)}$. Then $v'_2 <_{\pi_1} v$ as π_1 is a topological ordering of $D[V_1]$, so the pair (v'_2, v) is lexicographically smaller than the pair (v, v'_1) . Hence $c_{(v, v'_1)} >_{\pi} c_{(v'_2, v)}$, a contradiction with the construction of π .

We have verified that for all the vertices the conditions imposed on them are satisfied, so the instances are equivalent. \square

The *NP*-completeness of the SET SPLITTING problem [8], together with Lemmata 4.1, 4.2 and a trivial observation that LINE CLUSTER EMBEDDING is in *NP*, gives us the following theorem.

Theorem 4.3 *The LINE CLUSTER EMBEDDING problem is NP-complete.*

As mentioned before, the question of finding the smallest dimension of the Euclidean space, into which the given graph can be embedded, clearly generalizes testing embeddability into a line. Therefore, we have the following corollary.

Corollary 4.4 *It is NP-hard to decide the smallest dimension of the Euclidean space, into which a given signed graph can be embedded.*

4.2 Lower Bound on the Complexity

In this subsection we observe that the presented chain of reductions enables us also to establish a lower bound on the complexity of solving LINE CLUSTER EMBEDDING under ETH. Firstly, let us complete the chain of the reductions.

Lemma 4.5 *There exists a polynomial-time algorithm that given an instance φ of 3-CNF-SAT with n variables and m clauses, outputs an equivalent instance (U, \mathcal{F}) of SET SPLITTING with $|U| = 2n + 1$ and $\sum_{F \in \mathcal{F}} |F| = 2n + 4m$.*

Proof We construct the instance (U, \mathcal{F}) as follows. The universe U consists of one special element s and two literals $x, \neg x$ for every variable x of φ . The family \mathcal{F} includes (a) for every variable x , a set $F_x = \{x, \neg x\}$; (b) for every clause C , a set F_C consisting of s and all the literals in C . It is easy to check the claimed sizes of U, \mathcal{F} . We claim that the instance of SET SPLITTING (U, \mathcal{F}) is equivalent to the instance φ of 3-CNF-SAT.

Assume that ψ is a boolean evaluation of variables of φ that satisfies φ . We construct a set $X \subseteq U$ as follows: X consists of all the literals that are true in ψ . Now,

every set F_x is split, as exactly one of the literals is true and one is false, whereas every set F_C is split as well, as it contains a true literal, which belongs to X , and the special element s , which does not.

Now assume that $X \subseteq U$ is a solution to the SET SPLITTING instance (U, \mathcal{F}) . As taking $U \setminus X$ instead of X also yields a solution, without losing generality we can assume that $s \notin X$. Every set F_x is split by X ; therefore, exactly one literal of every variable belongs to X and exactly one does not. Let ψ be a boolean evaluation of variables of φ such that it satisfies all the literals belonging to X . Observe that ψ satisfies φ : for every clause C the set F_C has to be split, so, as $s \notin X$, one of the literals of C belongs to X and, thus, is satisfied by ψ . \square

Note that by pipelining Lemmata 4.5, 4.1 and 4.2, we obtain a reduction from 3-CNF-SAT to LINE CLUSTER EMBEDDING, where the output instance has a number of vertices and edges bounded linearly in the number of variables and clauses of the input formula. As the Exponential Time Hypothesis also excludes a possibility of having an algorithm for 3-CNF-SAT with running time subexponential in the total number of variables and clauses of the formula [12], we obtain the following.

Theorem 4.6 *Unless ETH fails, there is a constant $\delta > 0$ such that there is no algorithm that given a (V, E^+, E^-) instance of LINE CLUSTER EMBEDDING problem, solves it in $O(2^{\delta(|V|+|E^+|+|E^-|)})$ time.*

4.3 A Single-Exponential Algorithm for LINE CLUSTER EMBEDDING

Note that the trivial brute-force algorithm for LINE CLUSTER EMBEDDING checks all possible orderings, working in $O^*(n!)$ time. To complete the picture of the complexity of LINE CLUSTER EMBEDDING, we show that a simple dynamic programming approach can give single-exponential time complexity. This matches the lower bound obtained from under Exponential Time Hypothesis (up to a constant in the base of the exponent).

Before we proceed with the description of the algorithm, let us state a combinatorial observation that will be its main ingredient. Let (V, E^+, E^-) be the given LINE CLUSTER EMBEDDING instance. For $X \subseteq V$ and $v \notin X$ we will say that v is *good* for the set X iff

- no vertex $w \in X$ that is adjacent to v via a negative edge is simultaneously adjacent to some vertex from $V \setminus (X \cup \{v\})$ via a positive edge;
- no vertex $w \in V \setminus (X \cup \{v\})$ that is adjacent to v via a negative edge is simultaneously adjacent to some vertex from X via a positive edge.

Lemma 4.7 *An ordering π is a feasible solution of (V, E^+, E^-) if and only if every vertex $v \in V$ is good for the set $\{u : u <_{\pi} v\}$.*

Proof One direction is obvious: if π is a feasible solution, then every vertex v has to be good for the set $\{u : u <_{\pi} v\}$. If v would not be good for $\{u : u <_{\pi} v\}$, there

would exist a vertex w certifying that v is not good, and the condition imposed upon w would be not satisfied.

Now assume that every vertex $v \in V$ is good for $\{u : u <_{\pi} v\}$ and take an arbitrary vertex $v \in V$. If there were vertices $u_1 <_{\pi} u_2 <_{\pi} v$ such that $u_1 v \in E^+$ while $u_2 v \in E^-$, then u_2 would not be good for the set $\{u : u <_{\pi} u_2\}$, a contradiction. Similarly, if there were vertices $u_1 >_{\pi} u_2 >_{\pi} v$ such that $u_1 v \in E^+$ while $u_2 v \in E^-$, then u_2 would not be good for the set $\{u : u <_{\pi} u_2\}$, a contradiction as well. Therefore, the condition imposed on v is satisfied for an arbitrary choice of v . \square

Theorem 4.8 LINE CLUSTER EMBEDDING can be solved in $O^*(2^n)$ time and space complexity. Moreover, the algorithm can also output a feasible ordering of the vertices, if it exists.

Proof Let (V, E^+, E^-) be the given LINE CLUSTER EMBEDDING instance. Let $W = \{(v, X) : v \text{ is good for } X\}$. Let us construct a directed graph $D = (W, F)$, where $((v, X), (v', X')) \in F$ if and only if $X' = X \cup \{v\}$. As recognizing being good is clearly a polynomial time operation, the graph D can be constructed in $O^*(2^n)$ time and has that many vertices and edges. Observe that by Lemma 4.7 there is a feasible ordering π if and only if some sink $(v, V \setminus \{v\})$ is reachable from some source (u, \emptyset) ; indeed, such a path corresponds to introducing the vertices of V one by one in such a manner that each of them is good for the respective prefix. Reachability of any sink from any source can be, however, tested in time linear in the size of the graph using a breadth-first search. The search can also reconstruct the path in the same complexity, thus constructing the feasible solution. \square

5 Conclusions

In this paper we addressed a number of problems raised by Kermarrec and Thraves in [14] for embeddability of a signed graph into a line. We refined their study of the case of a complete signed graph by showing relation with proper interval graphs. Moreover, we have proven NP -hardness of the general case and shown an almost complete picture of its complexity.

Although the general case of the problem appears to be hard, real-life social networks have a certain structure. Is it possible to develop faster, maybe even polynomial-time algorithms for classes of graphs reflecting this structure?

Acknowledgments The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement n. 267959 (Michał Pilipczuk), n. 279352 (Marek Cygan), Foundation for Polish Science (Marek Cygan and Marcin Pilipczuk), and Polish National Science Centre grant N206567140 (Marcin Pilipczuk). We thank Sylvain Guillemot for pointing us to [7] and [9].

Open Access This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

1. Antal, T., Krapivsky, P.L., Redner, S.: Dynamics of social balance on networks. *Phys. Rev. E* **72**(3), 036121 (2005)
2. Belmonte, R., Vatshelle, M.: Graph classes with structured neighborhoods and algorithmic applications. In: Kolman, P., Kratochvíl, J. (eds.), WG, volume 6986 of Lecture Notes in Computer Science, pp. 47–58. Springer (2011)
3. Cartwright, D., Harary, F.: Structural balance: a generalization of heider's theory. *Psychol Rev* **63**(5), 277–293 (1956)
4. Corneil, D.G., Kim, H., Natarajan, S., Olariu, S., Sprague, A.P.: Simple linear time recognition of unit interval graphs. *Inf. Process. Lett.* **55**(2), 99–104 (1995)
5. Cygan, M., Pilipczuk, M., Pilipczuk, M., Wojtaszczyk, J.O.: Sitting closer to friends than enemies, revisited. In: Rovan, B., Sassone, V., Widmayer, P. (eds.), MFCS, volume 7464 of Lecture Notes in Computer Science, pp. 296–307. Springer (2012)
6. Davis, J.A.: Clustering and structural balance in graphs. *Hum. Relat.* **20**(2), 181 (1967)
7. Eppstein, D., Mumford, E.: Self-overlapping curves revisited. In: Mathieu, C. (ed.), SODA, pp. 160–169. SIAM (2009)
8. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman, New York (1979)
9. Guillemot, S., Jansson, J., Sung, W.-K.: Computing a smallest multilabeled phylogenetic tree from rooted triplets. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **8**(4), 1141–1147 (2011)
10. Ibarra, L.: A simple algorithm to find hamiltonian cycles in proper interval graphs. *Inf. Process. Lett.* **109**(18), 1105–1108 (2009)
11. Impagliazzo, R., Paturi, R.: On the complexity of k-SAT. *J. Comput. Syst. Sci.* **62**(2), 367–375 (2001)
12. Impagliazzo, R., Paturi, R., Zane, F.: Which problems have strongly exponential complexity. *J. Comput. Syst. Sci.* **63**(4), 512–530 (2001)
13. Ioannidou, K., Mertzios, G.B., Nikolopoulos, S.D.: The longest path problem is polynomial on interval graphs. In: Královic, R., Niwinski, D. (eds.), MFCS, volume 5734 of Lecture Notes in Computer Science, pp. 403–414. Springer (2009)
14. Kermarrec, A.-M., Thraves, C.: Can everybody sit closer to their friends than their enemies? In: Murlak, F., Sankowski, P. (eds.) MFCS, volume 6907 of Lecture Notes in Computer Science, pp. 388–399. Springer (2011)
15. Kunegis, J., Schmidt, S., Lommatzsch, A., Lerner, J., Luca, E.W.D.e., Albayrak, S.: Spectral analysis of signed graphs for clustering, prediction and visualization. In: SDM, p. 559. SIAM (2010)
16. Leskovec, J., Huttenlocher, D.P., Kleinberg, J.M.: Governance in social media: A case study of the wikipedia promotion process. In: Cohen, W.W., Gosling, S. (eds.) ICWSM. The AAAI Press (2010)
17. Leskovec, J., Huttenlocher, D.P., Kleinberg, J.M.: Predicting positive and negative links in online social networks. In: Rappa, M., Jones, P., Freire, J., Chakrabarti, S. (eds.), WWW, pp. 641–650. ACM (2010)
18. Leskovec, J., Huttenlocher, D.P., Kleinberg, J.M.: Signed networks in social media. In: Mynatt, E.D., Schoner, D., Fitzpatrick, G., Hudson, S.E., Edwards, W.K., Rodden, T. (eds.), CHI, pp. 1361–1370. ACM (2010)
19. Looges, P.J., Olariu, S.: Optimal greedy algorithms for indifference graphs. *Comput. Math. Appl.* **25**(7), 15–25 (1993)
20. George, B.: Mertzios. A polynomial algorithm for the k-cluster problem on the interval graphs. *Electron. Notes Discret. Math.* **26**, 111–118 (2006)
21. Szell, M., Lambiotte, R., Thurner, S.: Multirelational organization of large-scale social networks in an online world. *PNAS* **107**(31), 13636–13641 (2010)