Approximation Algorithms and Hardness Results for Shortest Path Based Graph Orientations*

Dima Blokh[1**], Danny Segev[2**], and Roded Sharan[1]

[1] Blavatnik School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel.
Email: {blokhdmi,roded}@post.tau.ac.il
[2] Department of Statistics, University of Haifa, Haifa 31905, Israel.
Email: segevd@stat.haifa.ac.il

Abstract. The graph orientation problem calls for orienting the edges of an undirected graph so as to maximize the number of pre-specified source-target vertex pairs that admit a directed path from the source to the target. Most algorithmic approaches to this problem share a common preprocessing step, in which the input graph is reduced to a tree by repeatedly contracting its cycles. While this reduction is valid from an algorithmic perspective, the assignment of directions to the edges of the contracted cycles becomes arbitrary, and the connecting source-target paths may be arbitrarily long. In the context of biological networks, the connection of vertex pairs via shortest paths is highly motivated, leading to the following variant: Given an undirected graph and a collection of source-target vertex pairs, assign directions to the edges so as to maximize the number of pairs that are connected by a shortest (in the original graph) directed path. Here we study this variant, provide strong inapproximability results for it and propose an approximation algorithm for the problem, as well as for relaxations of it where the connecting paths need only be approximately shortest.

1 Introduction

Protein-protein interactions form the skeleton of signal transduction in the cell. While many of these interactions carry directed signaling information, current interaction measurement technologies, such as yeast two hybrid [5] and co-immunoprecipitation [7], reveal the presence of an interaction, but not its directionality. Identifying this directionality is fundamental to our understanding of how these protein networks function.

To tackle the arising orientation problem, previous work has relied on information from perturbation experiments [13], in which a gene is perturbed (cause) and as a result other genes change their expression levels (effects). The fundamental assumption is that, for an effect to take place, there must be a directed

* Due to space limitations, some proofs are omitted from this extended abstract. These will appear in the full version of this paper.

** These authors contributed equally to this work.

path in the network from the causal gene to the affected gene. This setting calls for an orientation, that is, an assignment of directions to the edges of the network, such that a maximum number of pairs admit a directed path from the cause (source of response) to the affected genes (targets of the response).

Recently, large scale networks for many organisms have become available, leading to increasing interest in orientation problems of this nature. Medvedovsky et al. [10], Gamzu et al. [6], and later on Elberfeld et al. [3], were the first to study the MAXIMUM GRAPH ORIENTATION problem (MGO), where the objective is to direct the edges of a given (undirected) network so as to maximize the number of vertex pairs that are connected by directed source-target paths, which are allowed to be of arbitrary length. They proved that MGO is NP-hard to approximate to within a factor better than 12/13 and provided an $\Omega(\log\log n/\log n)$ approximation algorithm for it. It was further shown that MGO, as well as several natural extensions, admit efficient integer programming formulations [10, 11].

The main caveat of these approaches is that they all employ a preprocessing step in which cycles in the input graph are contracted one after the other, ending up with a tree network. Such structural modifications do not affect the optimization criterion, since directed connectivity can be preserved when cycles are consistently oriented in advance, either in clockwise or counter-clockwise direction. However, in practice, this preprocessing step results in a large fraction of the edges being arbitrarily oriented and in arbitrarily long directed source-target paths.

Other approaches to the problem concentrated on short connecting paths, which are more plausible biologically [13]. Gitter et al. [8] focused on paths whose length is bounded by a parameter k, showing that while the resulting problem is NP-hard, it can still be approximated within factor $O(k/2^k)$. Vinayagam et al. [12] developed a Bayesian learning strategy to predict the directionality of each edge based on the shortest paths that contain it.

Problem definition and our contribution. In this paper, we study the latter biologically-motivated setting [8], in which the directed paths connecting each pair of source-target vertices are required to be shortest. Let $G = (V, E)$ be an undirected graph with a vertex set V of size n and an edge set E of size m. Denote by $\delta_G(s, t)$ the length (number of edges) of a shortest path between s and t. An *orientation* $\vec{G}$ of G is a directed graph on the same vertex set whose edge set contains a single directed instance of every undirected edge, but nothing more. We say that a pair of vertices (s, t) is *satisfied* by an orientation $\vec{G}$ when the latter contains a directed s-t path of length $\delta_G(s, t)$. The MAXIMUM SHORTEST-PATH ORIENTATION (MSPO) problem is defined as follows:

Input: An undirected graph G and a collection $P = \{(s_1, t_1), \ldots, (s_k, t_k)\}$ of source-target vertex pairs.

Objective: Compute an orientation of G that satisfies a maximum number of pairs.

Our contribution is three-fold: (i) We relate the hardness of approximating MSPO to that of the Independent Set problem through a combinatorial construction called the "single-pair gadget", which may be interesting in its own right. Consequently, we show that this problem is NP-hard to approximate within factors $O(k^{1-\epsilon})$ and $O(m^{1/3-\epsilon})$, for any fixed $\epsilon > 0$ (Section 2). (ii) On the positive side, we adapt the approximation algorithm of [3], which was initially suggested for MGO in mixed graphs, and attain a performance guarantee of $\Omega(1/\max\{n,k\}^{1/\sqrt{2}})$ (Section 3.1). (iii) Last, we show that significantly better upper bounds can be obtained when one is willing to settle for bi-criteria approximations, where the strict requirement of connecting pairs only via shortest paths is relaxed and, instead, approximately-shortest paths are allowed. Here, we make use of random embeddings to compute $\tilde{O}(\log n)$-approximate shortest paths connecting an $\Omega(1/\log n)$ fraction of all pairs, with constant probability. Additionally, we show that by using $(1+\epsilon)$-approximate shortest paths one can satisfy an $\tilde{\Omega}(1/\sqrt{k})$ fraction of the pairs (Section 3.2).

2 Hardness of Approximation

In this section we provide a reduction from Independent Set showing that it is NP-hard to approximate MSPO to within factors $\Omega(1/k^{1-\epsilon})$ and $\Omega(1/m^{1/3-\epsilon})$ of optimum for any fixed $\epsilon > 0$. To this end, we first construct a *single-pair gadget*, which shows that there are MSPO instances in which even optimal orientations satisfy only one out of k source-target pairs. This construction will serve as the main building block of our hardness reduction. The single-pair gadget is also interesting in its own right, as it creates a strong separation between our definition of satisfying a given pair via a shortest path and the one studied by Medvedovsky et al. [10], in which pairs could be satisfied via any directed path, a setting where a logarithmic fraction of all pairs can always be satisfied.

2.1 The single-pair gadget

For convenience, we describe the single-pair gadget using an edge-weighted mixed graph, in which some of the edges are pre-directed. Later on, we show how to remove these extra constraints. In what follows, given any integer k, we show how to create an MSPO instance (G, P) with k pairs, $O(k^2)$ vertices and $O(k^2)$ edges, such that the following properties are satisfied: (1) For every pair in P there is some orientation that satisfies it, and (2) Any orientation of G satisfies at most one pair in P. To this end, we will argue that, in the instance described below, there is a unique shortest path connecting any given source-target pair. Moreover, these will be contradicting paths, in the sense that when one sets the direction of any such path from source to target all other paths can no longer be similarly directed (due to overlapping edges that need to be oriented in opposite directions).

Our construction is schematically drawn in Figure 1. In detail, the graph vertices are partitioned into k layers, $\mathcal{V}_1, \ldots, \mathcal{V}_k$, where $\mathcal{V}_i$ contains $2k-i$ vertices, $\{v_{i,1}, \ldots, v_{i,2k-i}\}$. There are three types of edges:

– Cross edges, E_{cross}: For every $1 \leq i \leq k-1$ and $i < j \leq k$, we have a pair of directed edges $(v_{j,i}, v_{i,2j-i-1})$ and $(v_{i,2j-i-2}, v_{j,i+1})$. The weight of these edges is 1.
– Contradiction edges, E_{cont}: For every $1 \leq i \leq k-1$ and $i < j \leq k$, we have an undirected edge $(v_{i,2j-i-2}, v_{i,2j-i-1})$. The weight of these edges is 0.
– Direction edges, E_{dir}: For every $1 \leq i \leq k-1$ and $i < j \leq k+1$, we have a directed edge $(v_{i,2j-i-1}, v_{i,2j-i})$. The weight of these edges is 2.

Finally, the collection of pairs is $P = \{(s_i, t_i) : 1 \leq i \leq k\}$, where $s_i = v_{i,1}$ and $t_i = v_{i,2k-i}$.

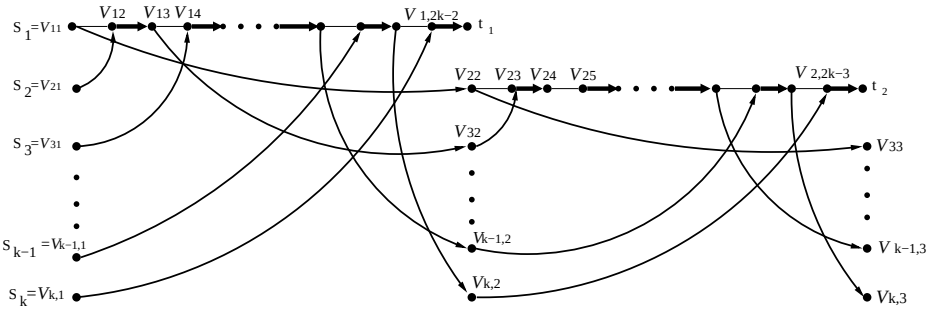

Fig. 1. The single-pair gadget (only the first two layers are shown). Here, direction edges are drawn as thick lines, cross edges as regular lines, and contradiction edges as thin lines.

We begin to analyze the single-pair gadget by highlighting a couple of structural properties that will be required to establish the uniqueness of shortest paths and the way in which they intersect. Observations 1 and 2 characterize the unique paths that connect vertices in one vertical column of the gadget (i.e, $v_{i,i}, \ldots, v_{k,i}$) to its successive column $(v_{i+1,i+1}, \ldots, v_{k,i+1})$. Somewhat informally, these observations will allow us to argue that for any s_i-t_i path, the sequence of column entry points $s_i = v_{i,1} \rightsquigarrow v_{i_2,2} \rightsquigarrow \cdots \rightsquigarrow v_{i_i,i}$ is non-decreasing in its vertical distance from s_i, that is, $i \leq i_2 \leq \cdots \leq i_i$.

Observation 1. For every $1 \leq i \leq k-1$ and $i < j_1 \leq j_2 \leq k$, there is only one path from $v_{j_1,i}$ to $v_{j_2,i+1}$. More specifically,

– If $j_1 = j_2$, this path takes the cross edge from $v_{j_1,i}$ to $v_{i,2j_1-i-1}$, then a single contradiction edge (in right-to-left direction), and finally the cross edge from $v_{i,2j_1-i-2}$ to $v_{j_1,i+1}$. Hence, the total weight of this path is 2.
– If $j_1 < j_2$, this path takes the cross edge from $v_{j_1,i}$ to $v_{i,2j_1-i-1}$, then travels in left-to-right direction in $\mathcal{V}_i$, alternating between direction and contradiction edges, and finally takes the cross edge from $v_{i,2j_2-i-2}$ to $v_{j_2,i+1}$. Hence, the total weight of this path is $2 + 2(j_2 - j_1)$.

Observation 2. For every $1 \leq i \leq k-1$ and $i < j_1 < j_2 \leq k$, there are no paths from $v_{j_2,i}$ to $v_{j_1,i+1}$.

With these observations in place, let us focus on one particular s_i-t_i path, p_i, which is schematically drawn in Figure 2 (for $i = 3$). This path repeatedly takes two cross edges and one contradiction edge $i-1$ times until it arrives to $v_{i,i}$, and then traverses $\mathcal{V}_i$ in left-to-right direction to reach $v_{i,2k-i} = t_i$. The next lemma shows that p_i must be shortest and unique.

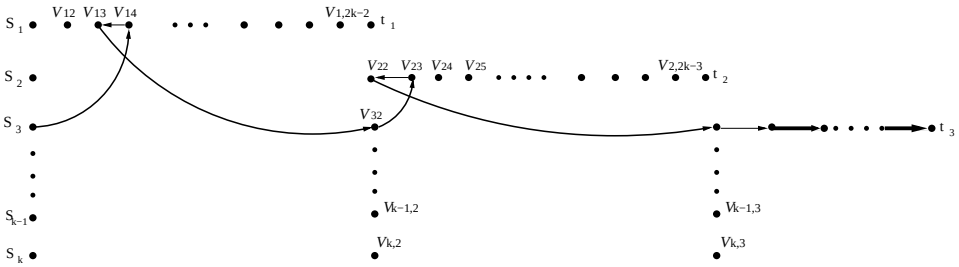

Fig. 2. The path p_3 connecting s_3 to t_3.

Lemma 1. *For every $1 \leq i \leq k$, the path p_i is the unique shortest s_i-t_i path.*

Proof. By definition of p_i, this path traverses $2(i-1)$ cross edges and $i-1$ contradiction edges prior to arriving at $v_{i,i}$. Then it traverses $k-i$ additional pairs of direction and cross edges before reaching t_i. Therefore, the total weight of p_i is exactly $2(i-1) + 2(k-i) = 2k-2$.

Now consider some other s_i-t_i path, $p \neq p_i$, and let $v_{j,i}$ be the entry point of p into the ith column (whose vertices are $v_{i,i}, \ldots, v_{k,i}$). Suppose $j = i$ and consider all the entry points of p into columns $2, \ldots, i-1$. By Observation 2 all these points must be at layer i and, hence, p identifies with p_i, contradicting our initial assumption. Thus, we may assume that $j > i$. By Observations 1 and 2, it follows that p traverses $2(i-1)$ cross edges and $j-i$ direction edges prior to arriving at $v_{j,i}$. The combined weight of those edges is $2(i-1)+2(j-i) = 2j-2$. From $v_{j,i}$, the path p must traverse the cross edge to $v_{i,2j-i-1}$ and then $k-j+1$ additional direction edges before reaching t_i. Consequently, the total weight of p is $(2j-2)+1+2(k-j+1) = 2k+1$, which is strictly greater than the weight of p_i, a contradiction. □

We conclude that for every pair $(s_i, t_i) \in P$ there exists an orientation satisfying this pair, in which all contradiction edges along p_i are oriented from s_i to t_i. It remains to show that any orientation satisfies at most one pair. Suppose to the contrary that there exists an orientation $\vec{G}$ that satisfies both (s_{i_1}, t_{i_1}) and (s_{i_2}, t_{i_2}), for some $i_1 < i_2$, meaning in particular that both p_{i_1} and p_{i_2} must agree with $\vec{G}$. However, these paths intersect in exactly one contradiction edge, $(v_{i_1,2i_2-i_1-2}, v_{i_1,2i_2-i_1-1})$, where in p_{i_1} it is orientated from left to right, while in p_{i_2} its direction is from right to left, a contradiction.

2.2 Reduction from Independent Set

We are now ready to make use of the single-pair gadget in order to prove the hardness of approximating MSPO. To simplify the presentation, we first establish this result for the more general setting in which the underlying graph is mixed (i.e., contains both directed and undirected edges) and weighted, similar to the construction described in Section 2.1.

Theorem 3. *For any fixed $\epsilon > 0$, it is NP-hard to approximate MSPO to within factors $\Omega(1/k^{1-\epsilon})$ and $\Omega(1/m^{1/2-\epsilon})$ of optimum in mixed weighted graphs.*

Proof. The basis for our reduction is the Independent Set problem, which is known to be hard to approximate to within a factor of $\Omega(1/n^{1-\epsilon})$ on an n-vertex graph for any fixed $\epsilon > 0$ [9]. Given an Independent Set instance $G = (V, E)$, we begin by constructing a single-pair gadget for $k = |V|$. In this construction, every layer $\mathcal{V}_i$ represents a vertex $v_i \in V$. Next, for every pair of vertices v_i and v_j such that $(v_i, v_j) \notin E$, we replace the cross edges $(v_{j,i}, v_{i,2j-i-1})$ and $(v_{i,2j-i-2}, v_{j,i+1})$ by a single directed edge $(v_{j,i}, v_{j,i+1})$ of weight 2. This modification is illustrated in Figure 3.

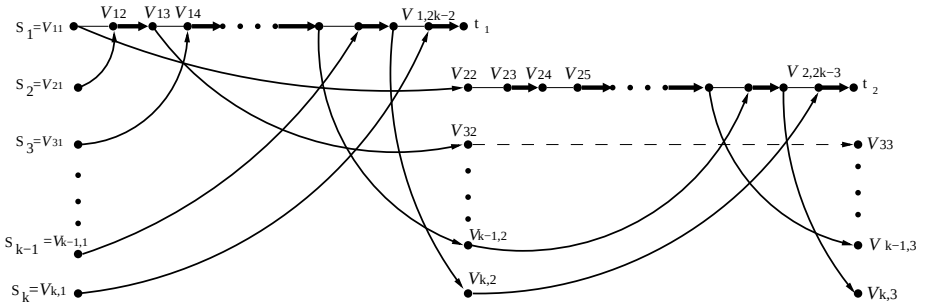

Fig. 3. An example modification for v_2 and v_3, where their newly added edge is drawn as a dashed line.

Now, for an original vertex v_i, let us focus once again on one particular s_i-t_i path, $\tilde{p}_i$. This path is created from the unique shortest path p_i in the original single-pair gadget by replacing every ⟨cross, contradiction, cross⟩ sequence of edges along p_i with its corresponding newly-added edge, whenever this modification has been made. By adapting the analysis given in Section 2.1, it is easy to verify that $\tilde{p}_i$ becomes the unique shortest s_i-t_i path. We proceed by observing that for every pair of original vertices v_i and v_j, $i < j$, the unique shortest paths $\tilde{p}_i$ and $\tilde{p}_j$, respectively connecting s_i to t_i and s_j to t_j, are edge-disjoint if and only if $(v_i, v_j) \notin E$. This follows from the way in which $\tilde{p}_i$ and $\tilde{p}_j$ were derived from p_i and p_j, along with our previous observation that p_i and p_j intersect in exactly one contradiction edge. This edge, $(v_{i_1,2i_2-i_1-2}, v_{i_1,2i_2-i_1-1})$, will be skipped in the modified instance by $\tilde{p}_j$ if and only if $(v_i, v_j) \notin E$.

It follows that there is a one-to-one correspondence between solutions $\{v_i : i \in I\}$ to the Independent Set instance and sets of pairs $\{(s_i, t_i) : i \in I\}$ that can be satisfied by some orientation. As the resulting MSPO instance consists of n pairs and $O(n^2)$ edges, the hardness of approximation for Independent Set implies bounds of $\Omega(1/k^{1-\epsilon})$ and $\Omega(1/m^{1/2-\epsilon})$ on the approximability of MSPO. □

It remains to show that the above reduction can be extended to the setting of undirected and unweighted graphs. For the former, we will show that when every directed edge is replaced in the single-pair gadget by an undirected edge, shortest paths remain unchanged. The following lemmas establish the correctness of this alteration. Due to space limitations and the rather involved nature of the corresponding proofs, these are deferred to the full version of our paper.

Lemma 2. *For every $1 \leq i \leq k$, a shortest s_i-t_i path in the undirected single-pair gadget cannot traverse cross edges in a direction different than the one defined in the mixed gadget.*

Lemma 3. *For every $1 \leq i \leq k$, a shortest s_i-t_i path in the undirected single-pair gadget cannot traverse direction edges from right to left.*

It remains to show how to remove edge weights from our construction. To this end, we first transform the original weights in the single-pair gadget so that these become positive integers. While cross and direction edges are associated with weights 1 and 2, respectively, contradiction edges are associated with zero weights. Our objective is to "scale" these values without changing the shortest path structure on the one hand, and while avoiding the use of large values on the other hand so as not to affect the inapproximability bound by much.

We begin by setting the weight of contradiction edges to $1/k$. This implies that for every $1 \leq i \leq k$, the total weight of the unique shortest s_i-t_i path p_i (see Section 2.1), which has been preserved during the reduction from mixed to undirected graphs, is at most $2k - 2 + (k-1)/k$. This is lighter than any other s_i-t_i path, which has weight at least $2k + 1$ according to the proof of Lemma 1. We proceed by scaling all edge weights by a factor of k to make them integral. Last, we replace each edge e of weight $w(e)$ by a path consisting of $w(e)$ unit-weight edges. As a result, the number of vertices and edges blows up to $O(k^3)$ instead of $O(k^2)$ as in the original gadget. Combined with our reduction from the Independent Set problem, the next inapproximability result follows.

Theorem 4. *For any fixed $\epsilon > 0$, it is NP-hard to approximate MSPO to within factors of $\Omega(1/k^{1-\epsilon})$ and $\Omega(1/m^{1/3-\epsilon})$ of optimum.*

Interestingly, we can use our construction to provide similar hardness of approximation results for the problem variant studied by Gitter at al. [8], for which non-trivial bounds were not known before. Further details will be provided in the full version of this paper.

3 Approximation Algorithms

In this section we provide an approximation algorithm for MSPO whose performance guarantee is sub-linear in either the number of vertices of the underlying graph or in the number of input pairs. In light of the hardness results established in Section 2, we cannot expect to come significantly closer to the optimal number of satisfied pairs, and the only possible avenue for improvement is decreasing the exponent we attain. However, a detailed inspection of Theorem 4 and its proof reveals that these do not exclude the possibility of obtaining better performance guarantees when one is willing to relax the strict requirement of satisfying pairs only via shortest paths and, instead, make use of approximately shortest paths. We explore this option as well, and show how to improve our previously-mentioned algorithm by utilizing such paths.

3.1 Exact shortest paths

To tackle MSPO, we adapt the approximation algorithm of Elberfeld et al. [3], which was initially suggested for MGO in mixed graphs. In that setting, pairs could be satisfied via any connecting path, regardless of its length, whereas in the current setting, connecting paths are required to be shortest.

Let (G, P) be an MSPO instance. For every $(s_i, t_i) \in P$, choose arbitrarily a shortest path p_i between them. Let $\mathcal{P} = \{p_i : (s_i, t_i) \in P\}$. The algorithm is iterative. At any point in time, we will be holding a partial orientation G_ℓ of G and a subset $\mathcal{P}_\ell \subseteq \mathcal{P}$ of shortest paths, where these sets are indexed according to the step number that has just been completed. Initially $G_0 = G$ and $\mathcal{P}_0 = \mathcal{P}$. Now, as long as none of the termination conditions described below is met, we proceed as follows:

1. Let $\hat{p} = (s, \ldots, t)$ be a minimum-length path in $\mathcal{P}_\ell$.
2. Orient $\hat{p}$ in the direction from s to t to obtain $G_{\ell+1}$.
3. To prevent the edges in $\hat{p}$ from being re-oriented in subsequent iterations, discard from $\mathcal{P}_\ell$ the path $\hat{p}$ as well as any path that overlaps (in edges) with it, obtaining $\mathcal{P}_{\ell+1}$.

There are two conditions that will cause the greedy iterations to terminate. For now, we state both conditions in terms of two parameters, $\alpha \geq 0$ and $\beta \geq 0$, whose values will be optimized later on.

1. $|\mathcal{P}_\ell| \leq n^\alpha$. In this case, we orient an arbitrary path from $\mathcal{P}_\ell$.
2. There exists a vertex v such that at least $|\mathcal{P}_\ell|^\beta$ paths in $\mathcal{P}_\ell$ go through v. Let $\mathcal{P}'_\ell$ be this sub-collection of paths and let P' be the collection of corresponding pairs. We show in the full version of this paper that one can satisfy at least $1/4$ of these pairs.

Under both termination conditions, we complete the orientation by directing the remaining edges in an arbitrary manner. With some modifications through their analysis, the arguments of Elberfeld et al. [3] essentially give rise to the next claim.

Lemma 4. *When the algorithm terminates due to condition 1, the number of satisfied pairs is* $\Omega(k/n^{\max\{1-\alpha(1-2\beta),\alpha\}})$. *Termination due to condition 2 leads to* $\Omega(k/\max\{n^{1-\alpha(1-2\beta)}, k^{1-\beta}\})$ *satisfied pairs.*

To obtain the best-possible performance guarantee, we pick values for α and β so as to minimize the maximum of all exponents mentioned above. To this end, the optimal values are $\alpha^* = \sqrt{1/2}$ and $\beta^* = 1 - \sqrt{1/2}$, in which case the maximal exponent becomes $\sqrt{1/2} \approx 0.707$.

Theorem 5. *MSPO can be approximated to within factor* $\Omega(1/\max\{n,k\}^{1/\sqrt{2}})$.

3.2 Approximate shortest paths

In order to improve on the performance guarantee attained in Theorem 5, we proceed by providing bi-criteria approximation algorithms for MSPO. Here, we relax the strict requirement of satisfying pairs only via shortest paths and, instead, allow approximately-shortest paths.

The precise setting we consider is as follows: For $\sigma \geq 1$, we say that a given orientation $\vec{G}$ σ-satisfies the pair (s_i, t_i) when it contains a directed s_i-t_i path of length at most σ times that of a shortest path, i.e., $\delta_{\vec{G}}(s_i, t_i) \leq \sigma \cdot \delta_G(s_i, t_i)$. For $\alpha \leq 1$ and $\sigma \geq 1$, we say that a given algorithm guarantees an (α, σ)-approximation when, for any instance of the problem, it computes an orientation that σ-satisfies at least $\alpha \cdot \mathrm{OPT}$ pairs. Here, OPT stands for the maximal number of pairs that can be 1-satisfied by any orientation.

An $(\Omega(1/\log n), \tilde{O}(\log n))$*-approximation via embedding.* With a slight adaptation of the metric embeddings terminology to our particular setting, the basic idea in this approach is to compute a random spanning tree $T \subseteq G$, sampled from a distribution $\mathcal{T}$ over a set of spanning trees in a way that pairwise distances do not get "stretched" by much in expectation. This line of work [2, 4] has evolved into a near-optimal bound due to Abraham, Bartal, and Neiman [1], who showed how to sample a random spanning tree such that the expected stretch is $\tilde{O}(\log n)$ uniformly over all vertex pairs, that is,

$$\max_{(u,v)\in V\times V} \mathrm{E}_{T\sim\mathcal{T}}\left[\frac{\delta_T(u,v)}{\delta_G(u,v)}\right] \leq \psi(n) = O(\log n \log\log n (\log\log\log n)^3) \ .$$

Here, $\mathrm{E}_{T\sim\mathcal{T}}[\cdot]$ denotes expectation with respect to the random choice of T, and $\psi(n)$ is our notation for the precise upper bound on the maximal expected stretch. In what follows, we argue that this result can be exploited to obtain logarithmic error bounds in both the number of satisfied pairs and in the extent to which distances are stretched.

Theorem 6. *There is a randomized algorithm that* $\tilde{O}(\log n)$*-satisfies* $\Omega(k/\log n)$ *pairs, with constant probability.*

Proof. We begin by computing a random spanning tree T using the embedding method of Abraham et al. [1]. With respect to this tree, let $P_{\text{small}} \subseteq P$ be the collection of pairs whose shortest path distances have not been significantly stretched beyond a factor of $\psi(n)$, which will be formally defined as $P_{\text{small}} = \{(s_i, t_i) \in P : \delta_T(s_i, t_i) \leq 2\psi(n) \cdot \delta_G(s_i, t_i)\}$. Since $\mathrm{E}_{T \sim \mathcal{T}}[\delta_T(s_i, t_i)] \leq \psi(n) \cdot \delta_G(s_i, t_i)$ for every pair $(s_i, t_i) \in P$, by Markov's inequality, each of these pairs is indeed a member of P_{small} with probability at least $1/2$. For this reason, $\mathrm{E}[|P_{\text{small}}|] \geq k/2$, which implies that $|P_{\text{small}}| \geq k/4$ with probability at least $1/3$, since

$$\begin{aligned}
\frac{k}{2} &\leq \mathrm{E}\left[|P_{\text{small}}|\right] \\
&= \Pr\left[|P_{\text{small}}| \geq \frac{k}{4}\right] \cdot \mathrm{E}\left[|P_{\text{small}}| \,\middle|\, |P_{\text{small}}| \geq \frac{k}{4}\right] \\
&\quad + \Pr\left[|P_{\text{small}}| < \frac{k}{4}\right] \cdot \mathrm{E}\left[|P_{\text{small}}| \middle| |P_{\text{small}}| < \frac{k}{4}\right] \\
&\leq \Pr\left[|P_{\text{small}}| \geq \frac{k}{4}\right] \cdot k + \left(1 - \Pr\left[|P_{\text{small}}| \geq \frac{k}{4}\right]\right) \cdot \frac{k}{4} \ .
\end{aligned}$$

Thus, with constant probably we obtain a spanning tree for which $|P_{\text{small}}|$, i.e., the number of pairs in P with stretch smaller than $2\psi(n) = \tilde{O}(\log n)$, contains a constant fraction of the pairs in P. Since we formed a tree instance, the maximum tree orientation algorithm of Medvedovsky et al. [10] can be used to compute an orientation that satisfies $\Omega(1/\log n) \cdot |P_{\text{small}}| = \Omega(k/\log n)$ pairs. □

An $(\tilde{\Omega}(1/\sqrt{k}), 1+\epsilon)$-approximation. Even though our embedding-based algorithm improves on the one described in Section 3.1 by orders of magnitude, at least as far as the number of satisfied pairs is concerned, it uses paths that may be $\tilde{\Omega}(\log n)$-fold longer than needed. In the remainder of this section, we propose another direction for improvement, in which pairs are guaranteed to be $(1+\epsilon)$-satisfied, for any required degree of accuracy $\epsilon > 0$. As it turns out, by resorting to ϵ-approximate paths, it is possible to satisfy $\tilde{\Omega}(1/k^{1/2})$ pairs, rather than $\Omega(1/\max\{n,k\}^{1/\sqrt{2}})$ as in the exact case.

Prior to formally describing our algorithm, it is worth pointing out that when a constant fraction of the pairs $(s_i, t_i) \in P$ are connected via very short paths, or more precisely, when $\delta_G(s_i, t_i) \leq 1/\epsilon$, the setting in question becomes very simple. In this case, a random orientation where the direction of each edge is picked at random, with equal probabilities for both options (independently of other edges), 1-satisfies each pair with probability at least $2^{-1/\epsilon}$. Therefore, the expected fraction of pairs that are satisfied is $\Omega(2^{-1/\epsilon})$. For this reason, we focus attention only on pairs for which $\delta_G(s_i, t_i) > 1/\epsilon$, and assume from this point on that all other pairs have already been discarded from P.

Let $\beta = \beta(n, k, \epsilon)$ be a parameter whose value will be optimized later on. As in the greedy algorithm, we use p_i to denote some shortest s_i-t_i path, arbitrarily picked in advance, and define $\mathcal{P} = \{p_i : (s_i, t_i) \in P\}$. Moreover, for a path $p \in \mathcal{P}$,

let $I_p(\mathcal{P})$ be the set of paths in $\mathcal{P}$ that intersect p, i.e, share at least one common edge. With these definitions in place, our algorithm works in two phases:

1. As long as there exists a path $p \in \mathcal{P}$, say from s to t, such that $|I_p(\mathcal{P})| < \beta$:
   (a) Orient p in the direction from s to t.
   (b) Discard from $\mathcal{P}$ the path p as well as all paths in $I_p(\mathcal{P})$.
2. Once the condition in phase 1 is no longer satisfied, let p be the shortest among all paths in $\mathcal{P}$, connecting s to t.
   (a) Partition the path p into at most $1/\epsilon$ edge-disjoint subpaths, each of length at most $\lceil \epsilon \cdot \delta_G(s,t) \rceil \leq 2\epsilon \cdot \delta_G(s,t)$, where this inequality holds since $\delta_G(s,t) \geq 1/\epsilon$.
   (b) Identify a subpath $\tilde{p}$ for which $|I_{\tilde{p}}(\mathcal{P})| \geq (\epsilon/2) \cdot |I_p(\mathcal{P})| \geq \epsilon\beta/2$, and let r be some arbitrary vertex in $\tilde{p}$.
   (c) Construct an r-rooted shortest-path tree T in the subgraph that results from unifying $\tilde{p}$ and all paths in $I_{\tilde{p}}(\mathcal{P})$. At this point in time, we have just created an instance of the maximum tree orientation problem, where the underlying tree is T and the collection of pairs are those corresponding to the paths in $I_{\tilde{p}}(\mathcal{P})$. Hence, we can use the algorithm of [10] to compute an orientation that satisfies $\Omega(1/\log n) \cdot |I_{\tilde{p}}(\mathcal{P})| = \Omega(\epsilon\beta/\log n)$ pairs.

Obviously, all pairs that were connected in phase 1 are 1-satisfied, since these connections are due to exact shortest paths. For this reason, it remains to show that every connection in phase 2 uses a $(1+\epsilon)$-approximate shortest path. This follows from the next claim, where we derive an upper bound on the factor by which pairwise distances can grow in T (for the relevant subset of pairs).

Lemma 5. *For every path $p_i \in I_{\tilde{p}}(\mathcal{P})$ connecting s_i to t_i,*

$$\delta_T(s_i, t_i) \leq (1 + 4\epsilon) \cdot \delta_G(s_i, t_i) \ .$$

Proof. Consider some path $p_i \in I_{\tilde{p}}(\mathcal{P})$, and let y_{s_i} be its first vertex (in the direction from s_i to t_i) that also belongs to the subpath $\tilde{p}$. Similarly, let y_{t_i} be the last vertex in p_i that still resides in $\tilde{p}$. Since T is an r-rooted shortest path tree in the union of $\tilde{p}$ and all paths in $I_{\tilde{p}}(\mathcal{P})$, and since the entire length of $\tilde{p}$ is at most $2\epsilon \cdot \delta_G(s,t)$ and $\delta_G(s,t) \leq \delta_G(s_i,t_i)$, we must have

$$\begin{cases} \delta_T(r, s_i) \leq \delta_G(r, y_{s_i}) + \delta_G(y_{s_i}, s_i) \leq 2\epsilon \cdot \delta_G(s_i, t_i) + \delta_G(y_{s_i}, s_i) \\ \delta_T(r, t_i) \leq \delta_G(r, y_{t_i}) + \delta_G(y_{t_i}, t_i) \leq 2\epsilon \cdot \delta_G(s_i, t_i) + \delta_G(y_{t_i}, t_i) \end{cases}$$

These inequalities can now be used to prove the desired claim, since:

$$\begin{aligned} \delta_T(s_i, t_i) &\leq \delta_T(s_i, r) + \delta_T(r, t_i) \\ &\leq (2\epsilon \cdot \delta_G(s_i, t_i) + \delta_G(y_{s_i}, s_i)) + (2\epsilon \cdot \delta_G(s_i, t_i) + \delta_G(y_{t_i}, t_i)) \\ &\leq (\delta_G(s_i, y_{s_i}) + \delta_G(y_{s_i}, y_{t_i}) + \delta_G(y_{t_i}, t_i)) + 4\epsilon \cdot \delta_G(s_i, t_i) \\ &= \delta_G(s_i, t_i) + 4\epsilon \cdot \delta_G(s_i, t_i) \\ &\leq (1 + 4\epsilon) \cdot \delta_G(s_i, t_i) \ . \end{aligned}$$

□

We conclude the description of the algorithm by showing how to optimize the value of $\beta = \beta(n, k, \epsilon)$ such that it balances between the worst-case performances of phases 1 and 2.

Theorem 7. *For any fixed $\epsilon > 0$, there is a deterministic algorithm that $(1+\epsilon)$-satisfies a fraction of $\Omega(1/\sqrt{(k \log n)/\epsilon})$ of the pairs.*

Proof. Let D be the number of paths that were eliminated from $\mathcal{P}$ in phase 1. By the condition to terminate this phase, at least D/β of these paths must have been oriented so that the corresponding pairs are satisfied. In addition, as shown above, the number of $(1+\epsilon)$-satisfied pairs in phase 2 is $\Omega(\epsilon\beta/\log n)$. Therefore, the overall number of $(1+\epsilon)$-satisfied pairs is at least

$$\begin{aligned}
\frac{D}{\beta} + \Omega\left(\frac{\epsilon\beta}{\log n}\right) &= \frac{1}{\beta} \cdot D + \Omega\left(\frac{\epsilon\beta}{(|P| - D)\log n}\right) \cdot (|P| - D) \\
&= \Omega\left(\min\left\{\frac{1}{\beta}, \frac{\epsilon\beta}{(|P| - D)\log n}\right\}\right) \cdot |P| \\
&= \Omega\left(\min\left\{\frac{1}{\beta}, \frac{\epsilon\beta}{k \log n}\right\}\right) \cdot k \ .
\end{aligned}$$

To obtain the best-possible performance guarantee, we pick a value for β so as to maximize $\min\{\frac{1}{\beta}, \frac{\epsilon\beta}{k \log n}\}$. The latter term attains its maximal value at $\beta^* = \sqrt{(k \log n)/\epsilon}$. □

Acknowledgments. We would like to thank Ofer Neiman for valuable discussions and pointers regarding metric embeddings. RS was supported by a research grant from the Israel Science Foundation (grant no. 241/11).

References

[1] I. Abraham, Y. Bartal, and O. Neiman. Nearly tight low stretch spanning trees. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 781–790, 2008.

[2] N. Alon, R. M. Karp, D. Peleg, and D. B. West. A graph-theoretic game and its application to the k-server problem. *SIAM Journal on Computing*, 24(1):78–100, 1995.

[3] M. Elberfeld, D. Segev, C. R. Davidson, D. Silverbush, and R. Sharan. Approximation algorithms for orienting mixed graphs. In *Proceedings of the 22nd Annual Symposium on Combinatorial Pattern Matching*, pages 416–428, 2011.

[4] M. Elkin, Y. Emek, D. A. Spielman, and S.-H. Teng. Lower-stretch spanning trees. *SIAM Journal on Computing*, 38(2):608–628, 2008.

[5] S. Fields. High-throughput two-hybrid analysis. The promise and the peril. *The FEBS Journal*, 272(21):5391–5399, 2005.

[6] I. Gamzu, D. Segev, and R. Sharan. Improved orientations of physical networks. In *10th International Workshop on Algorithms in Bioinformatics*, pages 215–225, 2010.

[7] A. Gavin, M. Bosche, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J. M. Rick, A. Michon, C. Cruciat, M. Remor, C. Hofert, M. Schelder, M. Brajenovic, H. Ruffner, A. Merino, K. Klein, M. Hudak, D. Dickson, T. Rudi, V. Gnau, A. Bauch, S. Bastuck, B. Huhse, C. Leutwein, M. Heurtier, R. R. Copley, A. Edelmann, E. Querfurth, V. Rybin, G. Drewes, M. Raida, T. Bouwmeester, P. Bork, B. Seraphin, B. Kuster, G. Neubauer, and G. Superti-Furga. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147, Jan. 2002.

[8] A. Gitter, J. Klein-Seetharaman, A. Gupta, and Z. Bar-Joseph. Discovering pathways by orienting edges in protein interaction networks. *Nucleic Acids Research*, 39(4):e22, 2011.

[9] J. Håstad. Clique is hard to approximate within $n^{1-\text{epsilon}}$. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, pages 627–636, 1996.

[10] A. Medvedovsky, V. Bafna, U. Zwick, and R. Sharan. An algorithm for orienting graphs based on cause-effect pairs and its applications to orienting protein networks. In *Proceedings of the 8th International Workshop on Algorithms in Bioinformatics*, pages 222–232, 2008.

[11] D. Silverbush, M. Elberfeld, and R. Sharan. Optimally orienting physical networks. In *Proceedings of the 15th Annual International Conference on Research in Computational Molecular Biology*, pages 424–436, 2011.

[12] A. Vinayagam, U. Stelzl, R. Foulle, S. Plassmann, M. Zenkner, J. Timm, H. E. Assmus, M. A. Andrade-Navarro, and E. E. Wanker. A directed protein interaction network for investigating intracellular signal transduction. *Science Signaling*, 4(189):rs8, 2011.

[13] C.-H. Yeang, T. Ideker, and T. Jaakkola. Physical network models. *Journal of Computational Biology*, 11(2/3):243–262, 2004.