Approximation Schemes for the Betweenness Problem in Tournaments and Related Ranking Problems (Revised Version)

Marek Karpinski[*] Warren Schudy[†]

Abstract

We settle the approximability of the *Minimum Betweenness* problem in tournaments by designing a *polynomial time approximation scheme (PTAS)*. No constant factor approximation was previously known. We also introduce a more general class of so-called *fragile* ranking problems and construct PTASs for them. The results depend on a new technique of dealing with fragile ranking constraints and could be of independent interest.

[*]Dept. of Computer Science, University of Bonn. Parts of this work done while visiting Microsoft Research. Email: marek@cs.uni-bonn.de

[†]IBM T. J. Watson Research Center. Parts of this work done while a student at Brown University and while visiting University of Bonn. Email: ws@cs.brown.edu

1 Introduction

We study the approximability of the Minimum Betweenness problem in tournaments (see [2]) that resisted so far efforts of designing polynomial time approximation algorithms with a constant approximation ratio. For the status of the general Betweenness problem, see e.g. [17, 9, 2, 8].

In this paper we design the first *polynomial time approximation scheme* (PTAS) for that problem, and generalize it to much more general class of ranking CSP problems, called here *fragile* problems. To our knowledge it is the first nontrivial approximation algorithm for the Betweenness problem in tournaments.

In the Betweenness problem we are given a ground set of *vertices* and a set of *betweenness constraints* involving 3 vertices and a *designated* vertex among them. The cost of a ranking of the elements is the number of betweenness constraints with the designated vertex not between the other two vertices. The goal is to find a ranking *minimizing* this cost. We refer to the Betweenness problem in tournaments, that is in instances with a constraint for every triple of vertices, as the BetweennessTour or *fully dense* Betweenness problem (see [2]). We consider also the k-ary extension k-FAST of the Feedback Arc Set in tournaments (FAST) problem (see [15, 1, 3]).

We extend the above problems by introducing a more general class of *fragile ranking* k-CSP problems inspired by the fragile (non-ranking) CSPs in [13]. A *constraint* S of a ranking k-CSP problem is called *fragile* if no two rankings of the vertices S that both satisfy the constraint differ by the position of a single vertex. A *ranking* k-CSP problem is called *fragile* if all its constraints are fragile.

We now formulate our main results.

Theorem 1. *There exists a PTAS for the* BetweennessTour *problem.*

The above answers an open problem of [2] on the approximation status of the Betweenness problem in tournaments.

We now formulate our first generalization.

Theorem 2. *There exist PTASs for all* fragile ranking *k-CSP problems in tournaments.*

Theorem 2 entails, among other things, existence of a PTAS for the k-ary extension of FAST. A PTAS for 2-FAST was given in [15].

Corollary. *There exists a PTAS for the* k-FAST *problem.*

We generalize BetweennessTour to arities $k \geq 4$ by specifying for each constraint S a pair of vertices in S that must be placed at the ends of the ranking induced by the vertices in S. Such constraints do not satisfy our definition of fragile, but do satisfy a weaker notion that we call *weak fragility.* The definition of weakly fragile is identical to the definition for fragile except that only four particular single vertex moves are considered, namely swapping the first two vertices, swapping the last two, and moving the first or last vertex to the other end. We now formulate our most general theorem.

Theorem 3. *There exist PTASs for all* weak-fragile ranking *k-CSP problems in tournaments.*

Corollary. *There exists a PTAS for the* k-BetweennessTour *problem.*

Additionally our algorithms are guaranteed not only to find a *low-cost* ranking but also a ranking that is *close to an optimal ranking* in the sense of Kendall-Tau distance. Karpinski and Schudy [14] recently utilized this extra feature to find an improved parameterized algorithm for BetweennessTour with runtime $2^{O(\sqrt{OPT/n})} + n^{O(1)}$.

Theorem 4. *The PTASs of Theorem 3 additionally return a set of $2^{\tilde{O}(1/\epsilon)}$ rankings, one of which is guaranteed to be both cheap (cost at most $(1+O(\epsilon))OPT$) and close to an optimal ranking (Kendall-Tau distance $O\left(\frac{poly(\frac{1}{\epsilon})OPT}{n^{k-2}}\right)$).*

All our PTASs are randomized but one can easily derandomize them by exhaustively considering every possible random choice.

Section 2 introduces notations and the problems we study. Section 3 introduces our algorithm and an intuitive sense of why it works. Section A analyzes the runtime. The remaining sections analyze the cost of the output of our algorithms.

2 Notation

First we state some core notation. Throughout this paper let V refer to the set of n objects (vertices) being ranked and $\epsilon > 0$ the desired approximation parameter. Our $O(\cdot)$ hides the arity k but not ϵ or n. Our $\tilde{O}(\cdot)$ additionally hides $(\log(1/\epsilon))^{O(1)}$. A *ranking* is a bijective mapping from a ground set $S \subseteq V$ to $\{1, 2, 3, \ldots, |S|\}$. An *ordering* is an injection from S into $\mathbb{R}$. Clearly every ranking is also an ordering. We use π and σ (plus superscripts) to denote rankings and orderings respectively. Let π^* denote an optimal ordering and OPT its cost. We let $\binom{n}{k}$ (for example) denote the standard binomial coefficient and $\binom{V}{k}$ denote the set of subsets of set V of size k.

For any ordering σ let $Ranking(\sigma)$ denote the ranking naturally associated with σ. To help prevent ties we relabel the vertices so that $V = \{1, 2, 3, \ldots, |V|\}$. We will often choose to place u in one of $O(1/\epsilon)$ positions $\mathcal{P}(u) = \{j\epsilon n + u/(n+1), 0 \le j \le 1/\epsilon\}$ (the $u/(n+1)$ term breaks ties). We say that an ordering is a *bucketed ordering* if $\sigma(u) \in \mathcal{P}(u)$ for all u. Let $Round(\pi)$ denote the bucketed ordering corresponding to π (rounding down), i.e. $Round(\pi)(u)$ equals $\pi(u)$ rounded down to the nearest multiple of ϵn, plus $u/(n+1)$.

Let $v \mapsto p$ denote the ordering over $\{v\}$ which maps vertex v to position $p \in \mathbb{R}$. For set Q of vertices and ordering σ with domain including Q let σ_Q denote the ordering over Q which maps $u \in Q$ to $\sigma(u)$, i.e. the restriction of σ to Q. For orderings σ^1 and σ^2 with disjoint domains let $\sigma^1 \mid \sigma^2$ denote the natural combined ordering over $Domain(\sigma^1) \cup Domain(\sigma^2)$. For example of our notations, $\sigma_Q \mid v \mapsto p$ denotes the ordering over $Q \cup \{v\}$ that maps v to p and $u \in Q$ to $\sigma(u)$.

A ranking k-CSP consists of a ground set V of *vertices*, an arity $k \ge 2$, and a *constraint system* c. Informally a constraint system c gives a 0/1 value (satisfied or not) for every ranking of every set $S \subseteq V$ of $|S| = k$ vertices. Formally a constraint system c is a function which maps rankings of vertices $S \subseteq V$ with $|S| = k$ to $\{0, 1\}$. For example if $k = 2$ and $V = \{u_1, u_2, u_3\}$ a constraint system c consists of the six values $c(u_1 \mapsto 1 \mid u_2 \mapsto 2)$, $c(u_1 \mapsto 2 \mid u_2 \mapsto 1)$, $c(u_1 \mapsto 1 \mid u_3 \mapsto 2)$, $c(u_1 \mapsto 2 \mid u_3 \mapsto 1)$, $c(u_2 \mapsto 1 \mid u_3 \mapsto 2)$, and $c(u_2 \mapsto 2 \mid u_3 \mapsto 1)$. A *weighted ranking CSP* has a *weighted constraint system* which maps rankings of vertices $S \subseteq V$, $|S| = k$ to non-negative reals $\mathbb{R}^+$. (To simplify terminology we present our results for unweighted CSPs only. We define weighted CSPs only because our algorithm uses one.) We refer to a set of vertices $S \subseteq V$, $|S| = k$ in the context of constraint system c as a *constraint*. We say constraint S is *satisfied* in ordering σ of S if $c(Ranking(\sigma)) = 0$. For brevity we henceforth abuse notation and omit the "*Ranking*" and write simply $c(\sigma)$. The objective of a ranking CSP is to find an ordering σ (w.l.o.g. a ranking) minimizing the number of unsatisfied constraints, which we denote by $C^c(\sigma) = \sum_{S \in \binom{Domain(\sigma)}{k}} c(\sigma_S)$.

We will frequently leave the CSP in question implicit in our notations, for exampling saying that a constraint S is *satisfied* without specifying the constraint system. In such cases the CSP should be clear from context. We use k, c and V to denote the arity, constraint system and ground set of the CSP that we are trying to optimize. We also use the shorthand $C(\sigma) = C^c(\sigma)$.

Definition 1. *A constraint S of constraint system c is* fragile *if no two orderings that satisfy it differ by the position of a single vertex. In other words constraint S is fragile if $c(\pi_S) + c(\pi'_S) \geq 1$ for all rankings π and π' over S that differ by a single vertex move, i.e. $\pi' = Ranking(v{\mapsto}p \,|\, \pi_{S\setminus\{v\}})$ for some $v \in S$ and half-integer $p \in \{1/2, 3/2, 5/2, \ldots, k + 1/2\}$.*

An alternate definition is that a satisfied fragile constraint becomes unsatisfied whenever a single vertex is moved, which is why it is called "fragile." Fragility is illustrated in Figure 1 (near Appendix A).

Definition 2. *A constraint S of constraint system c is* weakly fragile *if $c(\pi_S) + c(\pi'_S) \geq 1$ for all rankings π and π' that differ by a swap of the first two vertices, a swap of the last two, or a cyclic shift of a single vertex. In other words $\pi' = Ranking(v{\mapsto}p \,|\, \pi_{S\setminus\{v\}})$ for some $v \in S$ and $p \in \mathbb{R}$ with $(\pi(v), p) \in \{(1, 2 + \frac{1}{2}), (1, k + \frac{1}{2}), (k, k - \frac{3}{2}), (k, \frac{1}{2})\}$.*

Observe that weak fragility is equivalent to ordinary fragility for $k \leq 3$. Weak fragility is illustrated in Figure 2 (near Appendix A).

Our techniques handle ranking CSPs that are *fully dense* with weakly fragile constraints, i.e. every set S of k vertices corresponds to a weakly fragile constraint. Fully dense instances are also known as tournaments (by analogy with feedback arc set and tournament graphs).

Let $b^c(\sigma, v, p)$ denote the cost of the constraints in constraint system c involving vertex v in ordering $\sigma_{Domain(\sigma)\setminus\{v\}} \,|\, v{\mapsto}p$ formed by moving v to position p in ordering σ. Formally $b^c(\sigma, v, p) = \sum_{Q:\ldots} c(\sigma_Q \,|\, v{\mapsto}p)$, where the sum is over sets $Q \subseteq Domain(\sigma) \setminus \{v\}$ of size $k - 1$. Note that this definition is valid regardless of whether or not v is in $Domain(\sigma)$. The only requirement is that the range of σ excluding $\sigma(v)$ must not contain p. This ensures that the argument to $c(\cdot)$ is an ordering (injective). Analogously with the objective function C the superscript constraint system c in b^c defaults to the problem c that we are trying to solve when omitted.

We call a weighted ranking CSP instance with arity 2 a *feedback arc set (FAS) instance.* A FAS instance with vertex set V and constraint system w is equivalent to a weighted directed graph with arc weights $w_{uv} = w(u{\mapsto}2 \,|\, v{\mapsto}1)$ for $u, v \in V$. The objective function $C^w(\sigma)$ defined previously works out to finding an ordering of the vertices V minimizing the weight of the backwards arcs $C^w(\sigma) = \sum_{u,v:\sigma(u)>\sigma(v)} w_{uv}$. Similarly $b^w(\sigma, v, p) = \sum_{u \neq v} \begin{cases} w_{uv} & \text{if } \sigma(u) > p \\ w_{vu} & \text{if } \sigma(u) < p \end{cases}$. If a FAS instance with constraint system w satisfies $\alpha \leq w_{uv} + w_{vu} \leq \beta$ for all u, v and some $\alpha, \beta > 0$ we call it a (weighted) feedback arc set tournament (FAST) instance. We generalize to k-FAST as follows: a k-FAST constraint S is satisfied by one particular ranking of the vertices S and no others. Clearly k-FAST constraints are fragile.

We generalize BetweennessTour to $k \geq 4$ as follows. Each constraint S designates two vertices $\{u, v\}$, which must be the first and last positions, i.e. if π is the ranking of the vertices in S then $c(\pi) = \mathbb{1}(\{\pi(u), \pi(v)\} \neq \{1, k\})$. It is easy to see that BetweennessTour constraints are weakly fragile.

We use the following two results from the literature.

Theorem 5 ([15])**.** *Let w be a FAS instance satisfying $\alpha \leq w_{uv} + w_{vu} \leq \beta$ for $\alpha, \beta > 0$ and $\beta/\alpha = O(1)$. There is a PTAS for the problem of finding a ranking π minimizing $C^w(\pi)$ with runtime $n^{O(1)} 2^{\tilde{O}(1/\epsilon^6)}$.*

Theorem 6 (e.g. [6, 16])**.** *For any k-ary MIN-CSP and $\delta > 0$ there is an algorithm that produces a solution with cost at most δn^k more than optimal. Its runtime is $n^{O(1)} 2^{O(1/\delta^2)}$.*

Theorem 6 entails the following corollary.

Corollary 7. *For any $\delta > 0$ and constraint system c there is an algorithm* AddApprox *for the problem of finding a ranking π with $C(\pi) \leq C(\pi^*) + \delta n^k$, where π^* is an optimal ranking. Its runtime is $n^{O(1)} 2^{\tilde{O}(1/\delta^2)}$.*

3 Intuition and algorithm

We are in need for some new techniques different in nature from the techniques used for weighted FAST [15].

Our first idea is somehow analogous to the approximation of a differentiable function by a tangent line. Given a ranking π and any ranking CSP, the change in cost from switching to a similar ranking π' can be well approximated by the change in cost of a particular weighted feedback arc set problem (see proof of Lemma 23). Furthermore if the ranking CSP is fragile and fully dense the corresponding feedback arc set instance is a (weighted) tournament (Lemma 17). So *if* we somehow had access to a ranking similar to the optimum ranking π^* we could create this FAST instance and run the existing PTAS for weighted FAST [15] to get a good ranking.

We do not have access to π^* but we use techniques inspired by [13] to get close. We pick a random sample of vertices and guess their location in the optimal ranking to within (an additive) ϵn. We then create an ordering σ^1 greedily from the random sample. We show that this ordering is close to π^*, in that $|\pi^*(v) - \sigma^1(v)| = O(\epsilon n)$ for all but $O(\epsilon n)$ of the vertices (Lemma 12).

We then do a second greedy step (relative to σ^1), creating σ^2. We then identify a set U of *unambiguous* vertices (defined in Algorithm 1) for which we know $|\pi^*(v) - \sigma^2(v)| = O(\epsilon n)$ (Lemma 16). We temporarily set aside the $O(OPT/(\epsilon n^{k-1}))$ (Lemma 15) remaining vertices. These two greedy steps are similar in spirit to previous work on ordinary (non-ranking) everywhere-dense fragile CSPs [13] but substantially more involved.

We then use σ^2 to create a (weighted) FAST instance w that locally represents the CSP. It would not be so difficult to show that w is a close enough representation for an additive approximation, but we want a multiplicative $1 + \epsilon$ approximation. Showing this requires overcoming two obstacles that are our main technical contribution.

Firstly the error in σ^2 causes the weights of w to have significant error (Lemma 19) even in the extreme case of $OPT = 0$. At first glance even an exact solution to this FAST problem would seem insufficient, for how can solving a problem similar to the desired one lead to a precisely correct solution? Fortunately FAST is somewhat special. It is easy to see that a zero-cost instance of FAST cannot be modified to change its optimal ranking without modifying an arc weight by at least 1/2. We extend this idea to cases where OPT is small but non-zero (Lemma 23).

The second obstacle is that the incorrect weights in FAST instance w may increase the optimum cost of w far above OPT, leaving the PTAS for FAST free to return a poor ranking. To remedy this we create a new FAST instance $\bar{w}$ by canceling weight on opposing arcs, i.e. reducing w_{uv} and w_{vu} by the same amount. The resulting simplified instance $\bar{w}$ clearly has the same optimum ranking as w but a smaller optimum value. The PTAS for FAST requires that the ratio of the maximum and the minimum of $w_{uv} + w_{vu}$ must be bounded above by a constant so we limit the amount of cancellation to ensure this (Lemma 17). It turns out that this cancellation trick is sufficient to ensure that the PTAS for FAST does not introduce too much error (Lemma 20).

Finally we greedily insert the relatively few ambiguous vertices into the ranking output by the PTAS for FAST [15] (Appendix C).

For any ordering σ with domain U we will shortly define a weighted feedback arc set instance w^σ

which approximates the overall problem c in the neighborhood of ordering σ. In particular changes in the objective $C = C^c$ are approximately equal to changes in C^{w^σ}. Before giving the definition of w^σ we describe how it was chosen. For simplicity of illustration let us suppose that $|V| = k$ and hence we have only one constraint; the general case will follow by making w^σ_{uv} a sum over contributions by the various constraints $S \supseteq \{u, v\}$. We are looking for good approximations for the costs of nearby ordering, so let us consider the nearest possible ordering: let σ' be idential to σ except that two adjacent vertices, call them u and v, are swapped: $\sigma(u) < \sigma(v)$ and $\sigma'(u) > \sigma'(v)$. Clearly $C^{w^\sigma}(\sigma') - C^{w^\sigma}(\sigma) = w^\sigma_{uv} - w^\sigma_{vu}$. It is therefore natural to set $w^\sigma_{vu} = c(\sigma)$ and $w^\sigma_{uv} = c(\sigma')$, hence $C(\sigma') - C(\sigma) = C^{w^\sigma}(\sigma) - C^{w^\sigma}(\sigma)$ as desired.

So what about w^σ_{uv} for u and v that are not adjacent in σ? It turns out that we can pick practically anything for the other w^σ_{uv} as long as we keep $C^{w^\sigma}(\sigma)$ small and $w^\sigma_{uv} + w^\sigma_{vu}$ relatively uniform. We extend the above definition to non-adjacent u, v with $\sigma(u) < \sigma(v)$ as follows: set $w^\sigma_{vu} = c(\sigma)$ and $w^\sigma_{uv} = c(\sigma')$, where σ' is identical to σ except that v is placed immediately before u. (Another natural option would be to set $w_{vu} = 0$ and $w_{uv} = 1$ for non-adjacent u, v with $\sigma(u) < \sigma(v)$.)

With this motivation in hand we now give the formal definition of w^σ. For any ordering σ with domain U let w^σ_{uv} equal the number of the constraints $\{u, v\} \subseteq S \subseteq U$ with $c(\sigma') = 1$ where (1) $\sigma' = (\sigma_{S\setminus\{v\}} \,|\, v \mapsto p)$, (2) $p = \sigma(u) - \delta$ if $\sigma(v) > \sigma(u)$ and $p = \sigma(v)$ otherwise, and (3) $\delta > 0$ is sufficiently small to put p adjacent to $\sigma(u)$. In other words if v is after u in σ it is placed immediately before u in σ'. Observe that $0 \le w^\sigma_{uv} \le \binom{|U|-2}{k-2}$.

The following Lemma follows easily from the definitions.

Lemma 8. *For any ordering σ we have (1) $C^{w^\sigma}(\sigma) = \binom{k}{2} C(\sigma)$ and (2) $b^{w^\sigma}(\sigma, v, \sigma(v)) = (k-1) \cdot b(\sigma, v, \sigma(v))$ for all v.*

Proof. Observe that all w^σ_{uv} that contribute to $C^{w^\sigma}(\sigma)$ or $b^{w^\sigma}(\sigma, v, \sigma(v))$ satisfy $\sigma(u) > \sigma(v)$ and hence the σ' in the definition of w^σ_{uv} is equal to σ. It follows that each w^σ_{uv} that contributes to $C^{w^\sigma}(\sigma)$ or $b^{w^\sigma}(\sigma, v, \sigma(v))$ is equal to the number of constraints containing u and v that are unsatisfied in σ. The $\binom{k}{2}$ and $k-1$ factors appear because each constraint S contributes to w^σ_{uv} for a variety of $u, v \in S$. □

The weighted feedback arc set instance w^σ is insufficient for our purposes because its objective value can be large even when the optimal cost of c is small. To remedy this we cancel the weight on opposing arcs (within limits), yielding another weighted feedback arc set instance $\bar{w}^\sigma$. In particular for any ordering σ we define $\bar{w}^\sigma_{uv} = w^\sigma_{uv} - \min(\frac{1}{10 \cdot 3^{k-1}} \binom{|U|-2}{k-2}, w^\sigma_{uv}, w^\sigma_{vu})$, where U is the domain of σ. Observe that $C^{w^\sigma}(\pi') - C^{\bar{w}^\sigma}(\pi')$ is a non-negative constant independent of ranking π'. Therefore the feedback arc set problems induced by w^σ and $\bar{w}^\sigma$ have the same optimal rankings but an approximation factor of $(1+\epsilon)$ is a stronger guarantee for $\bar{w}^\sigma$ than for w^σ.

For any orderings σ and σ' with domain U, we say that $\{u, v\} \subseteq U$ is a *σ/σ'-inversion* if $\sigma(u) - \sigma(v)$ and $\sigma'(u) - \sigma'(v)$ have different signs. Let $d(\sigma, \sigma')$ denote the number of σ/σ'-inversions (a.k.a. Kendall Tau distance). We say that v does a *left to right (σ, p, σ', p')-crossing* if $\sigma(v) < p$ and $\sigma'(v) > p'$. We say that v does a *right to left $(\sigma, p/\sigma', p')$-crossing* if $\sigma(v) > p$ and $\sigma'(v) < p'$. We say that v does a *(σ, p, σ', p')-crossing* if v does a crossing of either sort. We say that *u σ/σ'-crosses* $p \in \mathbb{R}$ if it does a (σ, p, σ', p)-crossing.

With these notations in hand we present our Algorithm 1 for approximating a weak fragile rank k-CSP. The non-deterministic "guess (by exhaustive sampling)" on line 2 of our algorithm should be implemented in the traditional manner: place the remainder of the algorithm in a loop over possible orderings of the sample, with the overall return value equal to the best of the π^4 rankings

Algorithm 1 A $(1+O(\epsilon))$-approximation for weak fragile rank k-CSPs in tournaments.

Input: Vertex set V, $|V| = n$, arity k, system c of fully dense arity k constraints, and approximation parameter $\epsilon > 0$.

1: Run AddApprox($\epsilon^5 n^k$) and return the result if its cost is at least $\epsilon^4 n^k$

2: Pick sets $T_1, \dots, T_t$ uniformly at random with replacement from $\binom{V}{k-1}$, where $t = \frac{14\ln(40/\epsilon)}{\binom{k}{2}\epsilon}$. Guess (by exhaustion) bucketed ordering σ^0, which is the restriction of $Round(\pi^*)$ to the sampled vertices $\bigcup_i T_i$, where π^* is an optimal ranking.

3: Compute bucketed ordering σ^1 greedily with respect to the random samples and σ^0, i.e.: $\sigma^1(u) = \text{argmin}_{p\in\mathcal{P}(u)} \hat{b}(u,p)$ where $\hat{b}(u,p) = \frac{\binom{n}{k-1}}{t} \sum_{i:u\notin T_i} c({\sigma^0}_{T_i} \mid v{\mapsto}p)$.

4: For each vertex v: If $b(\sigma^1, v, p) \le 13k^4 3^{k-1}\epsilon\binom{n-1}{k-1}$ for some $p \in \mathcal{P}(v)$ then call v *unambiguous* and set $\sigma^2(v)$ to the corresponding p (pick any if multiple p satisfy). Let U denote the set of unambiguous vertices, which is the domain of bucketed ordering σ^2.

5: Compute feedback arc set instance $\bar{w}^{\sigma^2}$ over unambiguous vertices U (see text). Solve it using the FAST PTAS [15]. Do single vertex moves until local optimality (with respect to the FAST objective function), yielding ranking π^3 of U.

6: Create ordering σ^4 over V defined by $\sigma^4(u) = \begin{cases} \pi^3(u) & \text{if } u \in U \\ \text{argmin}_{p=v/(n+1)+j, 0\le j\le n}\, b(\pi^3, u, p) & \text{otherwise} \end{cases}$. In other words insert each vertex $v \in V \setminus U$ into $\pi^3(v)$ greedily.

7: Return $\pi^4 = Ranking(\sigma^4)$.

found. Our algorithm can be derandomized by choosing $T_1, \dots, T_t$ non-deterministically rather than randomly; see the runtime analysis in Appendix A for details.

If $OPT \ge \epsilon^4 n^k$ then the first line of the algorithm is sufficient for a PTAS so for the remainder of the analysis we assume that $OPT \le \epsilon^4 n^k$. For most of the analysis we actually need something weaker, namely that OPT is at most some sufficiently small constant times $\epsilon^2 n^k$. We only need the full $OPT \le \epsilon^4 n^k$ in one place in Appendix C.

4 Analysis of σ^1

Let $\sigma^\square = Round(\pi^*)$. Call vertex v *costly* if $b(\sigma^\square, v, \sigma^\square(v)) \ge 2\binom{k}{2}\epsilon\binom{n-1}{k-1}$ and *non-costly* otherwise.

Lemma 9. *The number of costly vertices is at most* $\frac{k \cdot OPT}{\epsilon\binom{k}{2}\binom{n-1}{k-1}}$.

Lemma 9 is proven in Appendix B. The outline of the proof is $kC(\pi^*) = \sum_v b(\pi^*, v, \pi^*(v)) \approx \sum_v b(\sigma^\square, v, \sigma^\square(v)) \ge (\text{number costly})2\binom{k}{2}\epsilon\binom{n-1}{k-1}$.

Lemma 10. *Let σ be an ordering of V, $|V| = n$, $v \in V$ be a vertex and $p, p' \in \mathbb{R}$. Let B be the set of vertices (excluding v) between p and p' in σ. Then* $b(\sigma, v, p) + b(\sigma, v, p') \ge \frac{|B|}{(n-1)3^{k-1}}\binom{n-1}{k-1}$.

Proof. By definition

$$b(\sigma, v, p) + b(\sigma, v, p') = \sum_{Q:\cdots} \left[c(\sigma_Q \mid v{\mapsto}p) + c(\sigma_Q \mid v{\mapsto}p')\right] \qquad (1)$$

where the sum is over sets $Q \subseteq U \setminus \{v\}$ of $k-1$ vertices.

We consider the illustrative special case of betweenness tournament (or more generally fragile problems with arity $k = 3$) here and defer the general case to Appendix B. Betweenness constraints have a special property: the quantity in brackets in (1) is at least 1 for every Q that has at least one vertex between p and p' in π. There are at least $|B|(n-2)/2$ such sets, which can easily be lower-bounded by the desired $\frac{|B|}{(n-1)3^{3-1}}\binom{n-1}{3-1}$. $\square$

For vertex v we say that a position $p \in \mathcal{P}(v)$ is *v-out of place* if there are at least $6\binom{k}{2}3^{k-1}\epsilon n$ vertices between p and $\sigma^{\square}(v)$ in $\sigma^{\square}$. We say vertex v is *out of place* if $\sigma^1(v)$ is v-out of place.

Lemma 11. *The number of non-costly out of place vertices is at most $\epsilon n/2$ with probability at least 9/10.*

The proof is in Appendix B. It uses Lemma 10 and the definitions of out-of-place and costly to show that $b(\sigma^{\square}, v, \sigma^{\square}(v))$ is much smaller than $b(\sigma^{\square}, v, p)$ for any v-out of place p, and then Chernoff and union bounds to show that $\hat{b}(v,p)$ is sufficiently concentrated about its mean $b(\sigma^{\square}, v, p)$ so that the minimum $\hat{b}(v,p)$ must occur for a p that is not v-out of place.

Lemma 12. *With probability at least 9/10 the following are simultaneously true:*

1. *The number of out of place vertices is at most ϵn.*
2. *The number of vertices v with $|\sigma^1(v) - \sigma^{\square}(v)| > 3k^2 3^{k-1}\epsilon n$ is at most ϵn*
3. *$d(\sigma^1, \sigma^{\square}) \leq 6k^2 3^{k-1}\epsilon n^2$*

Proof. By Lemma 9 and the fact $OPT \leq \epsilon^4 n^k$ we have at most $\frac{k \cdot OPT}{\binom{k}{2}\epsilon\binom{n-1}{k-1}} \leq \epsilon n/2$ costly vertices for n sufficiently large. Therefore Lemma 11 implies the first part of the Lemma. We finish the proof by showing that whenever the first part holds the second and third parts hold as well.

Observe that there are exactly ϵn vertices in $\sigma^{\square}$ between any two consecutive positions in $\mathcal{P}(v)$. It follows that any vertex with $|\sigma^1(v) - \sigma^{\square}(v)| > 3k^2 3^{k-1}\epsilon n \geq (6\binom{k}{2}3^{k-1} + 1)\epsilon n$ must necessarily be v-out of place, completing the proof of the second part of the Lemma.

For the final part observe that if u and v are a $\sigma^1/\sigma^{\square}$-inversion and not among the ϵn out of place vertices then, by definition of out-of-place, there can be at most $2 \cdot 6\binom{k}{2}3^{k-1}\epsilon n$ vertices between $\sigma^{\square}(v)$ and $\sigma^{\square}(u)$ in $\sigma^{\square}$. For each u there are therefore only $24\binom{k}{2}3^{k-1}\epsilon n$ possibilities for v. Therefore $d(\sigma^1, \sigma^{\square}) \leq \epsilon n^2 + 24\binom{k}{2}3^{k-1}\epsilon n \cdot n/2 \leq 6\epsilon k^2 3^{k-1} n^2$. $\square$

The remainder of our analysis assumes that the event of Lemma 12 holds without stating so explicitly.

5 Analysis of σ^2

The following key Lemma shows the sensitivity of $b(\sigma, v, p)$ to its first and third arguments. It is proven in Appendix B.

Lemma 13. *For any constraint system c with arity $k \geq 2$, orderings σ and σ' over vertex set $T \subseteq V$, vertex $v \in V$ and $p, p' \in \mathbb{R}$ we have*

1. $$|b^c(\sigma, v, p) - b^c(\sigma', v, p')| \leq \binom{n-2}{k-2}(\textit{number of crossings}) + \binom{n-3}{k-3} d(\sigma, \sigma')$$
2. $$|b^c(\sigma, v, p) - b^c(\sigma', v, p')| \leq \binom{n-2}{k-2}\left(|\textit{net flow}| + k\sqrt{d(\sigma, \sigma')}\right)$$

where $\binom{n-3}{k-3} = 0$ if $k = 2$, (net flow) is $|\{v \in T : \sigma'(v) > p'\}| - |\{v \in T : \sigma(v) > p\}|$, and (number of crossings) is the number of $v \in T$ that do a (σ, p, σ', p')-crossing.

Observe that the quantity *net flow* in Lemma 13 is zero whenever $p = p'$ and σ and σ' are both *rankings*. Therefore we have the following useful corollary.

Corollary 14. *Let π and π' be rankings over vertex set U and w a FAST instance over U. Then $|b^w(\pi, v, p) - b^w(\pi', v, p)| \le 2(\max_{r,s} w_{rs})\sqrt{d(\pi, \pi')}$ for all v and $p \in \mathbb{R} \setminus \mathbb{Z}$.*

We let U denote the set of unambiguous vertices as defined in Algorithm 1.

Lemma 15. *We have $|V \setminus U| \le \frac{k \cdot OPT}{\epsilon \binom{k}{2}\binom{n-1}{k-1}} = O(\frac{n}{\epsilon} \cdot \frac{OPT}{n^k})$.*

Proof. Observe that the number of vertices that $\sigma^\square/\sigma^1$-cross a particular p is at most $2 \cdot 6k^2 3^{k-1}\epsilon n$ by Lemma 12 (first part). Therefore we apply Lemmas 12 and 13, yielding

$$|b(\sigma^\square, v, p) - b(\sigma^1, v, p)| \le \binom{n-2}{k-2} 12k^2 3^{k-1}\epsilon n + \binom{n-3}{k-3} 6k^2 3^{k-1}\epsilon n^2 \le 12\epsilon k^4 3^{k-1}\binom{n-1}{k-1} \quad (2)$$

for all v and p.

Fix a non-costly v. By definition of costly $b(\sigma^\square, v, \sigma^\square(v)) \le 2\binom{k}{2}\epsilon\binom{n-1}{k-1} \le k^4 3^{k-1}\epsilon\binom{n-1}{k-1}$, hence $b(\sigma^1, v, \sigma^\square(v)) \le 13k^4 3^{k-1}\epsilon\binom{n-1}{k-1}$, so $v \in U$.

Finally recall Lemma 9. □

We define $\pi^\circledast$ to be the ranking induced by the restriction of π^* to U, i.e. $\pi^\circledast = \mathit{Ranking}({\pi^*}_U)$.

Lemma 16. *All vertices in the unambiguous set U satisfy $|\sigma^2(v) - \pi^\circledast(v)| = O(\epsilon n)$.*

Proof. The triangle inequality $|\sigma^2(v) - \pi^\circledast| \le |\sigma^2(v) - \pi^*(v)| + |\pi^*(v) - \pi^\circledast|$ allows us to instead bound the two terms $|\sigma^2(v) - \pi^*(v)|$ and $|\pi^*(v) - \pi^\circledast|$ separately by $O(\epsilon n)$. We bound $|\sigma^2(v) - \pi^*(v)|$ first.

Since π^* is a ranking the number of vertices $|B|$ between $\pi^*(v)$ and $\sigma^2(v)$ in π^* is at least $|\pi^*(v) - \sigma^2(v)| - 1$. Therefore we have

$$\begin{aligned} \frac{|\pi^*(v) - \sigma^2(v)| - 1}{(n-1)3^{k-1}}\binom{n-1}{k-1} &\le b(\pi^*, v, \sigma^2(v)) + b(\pi^*, v, \pi^*(v)) && \text{(Lemma 10)} \\ &\le 2b(\pi^*, v, \sigma^2(v)) && \text{(Optimality of } \pi^*\text{).} \quad (3) \end{aligned}$$

We next apply the first part of Lemma 13 to π^* and $\sigma^\square$, bounding the number of crossings and $d(\pi^*, \sigma^\square)$ using the definition $\sigma^\square = \mathit{Round}(\pi^*)$, yielding

$$b(\pi^*, v, \sigma^2(v)) \le b(\sigma^\square, v, \sigma^2(v)) + O(\epsilon n^{k-1}). \quad (4)$$

Next recalling (2) from the proof of Lemma 15 we have

$$b(\sigma^\square, v, \sigma^2(v)) \le b(\sigma^1, v, \sigma^2(v)) + O(\epsilon n^{k-1}). \quad (5)$$

Combining (3), (4) and (5) we conclude that $|\pi^*(v) - \sigma^2(v)| = O(\epsilon n)$.

Now we prove $|\pi^*(v) - \pi^\circledast| = O(\epsilon n)$. Lemma 15, the definition of $\pi^\circledast$, and the assumption that $OPT \le \epsilon^4 n^k$ imply that $|\pi^\circledast(v) - \pi^*(v)| \le \frac{k \cdot OPT}{\epsilon\binom{k}{2}\binom{n-1}{k-1}} = O(\epsilon n)$. □

6 Analysis of π^3

Note that all orderings and costs in this section are over the set of unambiguous vertices U as defined in Algorithm 1, not V. We note that Lemma 15 and the assumption that $OPT \le \epsilon^4 n^k$ is small imply that $|U| = n - O(\epsilon^3 n)$.

Lemma 17. $\frac{1}{3^{k-1}}(1-2/10)\binom{|U|-2}{k-2} \le \bar{w}^{\sigma^2}_{uv} + \bar{w}^{\sigma^2}_{vu} \le 2\binom{|U|-2}{k-2}$*, i.e.* $\bar{w}^{\sigma^2}$ *is a weighted FAST instance.*

The proof, which uses weak fragility, is in Appendix B.

Lemma 18. *Assume ranking* π *and ordering* σ *satisfy* $|\pi(u) - \sigma(u)| = O(\epsilon n)$ *for all* u*. For any* u, v*, let* N_{uv} *denote the number of* $S \supset \{u, v\}$ *such that not all pairs* $\{s, t\} \neq \{u, v\}$ *are in the same order in* σ *and* π*. We have* $N_{uv} = O(\epsilon n^{k-2})$*.*

Proof. Such a pair $\{s, t\}$ must satisfy $|\pi(s) - \pi(t)| = 2 \cdot O(\epsilon n)$, but few constraints contain such a pair. □

Lemma 19. *The following inequalities hold:*

1. $w^{\sigma^2}_{uv} \le w^{\pi^\circledast}_{uv} + O(\epsilon n^{k-2})$
2. $\bar{w}^{\sigma^2}_{uv} \le (1 + O(\epsilon)) w^{\pi^\circledast}_{uv}$

Proof. The only constraints $S \supset \{u, v\}$ that contribute differently to the left- and right-hand sides of the first part are those containing a $\{s, t\} \neq \{u, v\}$ that are a $\sigma^2/\pi^\circledast$-inversion. By Lemmas 16 and 18 we can bound the number of such constraints by $O(\epsilon n^k)$, completing the proof of the first part.

If $w^{\pi^\circledast}_{uv} \ge \frac{1}{2 \cdot 3^{k-1}}\binom{|U|-2}{k-2}$ the second part follows from the first part and the trivial fact $\bar{w} \le w$. Otherwise by the first part we have $w^{\sigma^2}_{uv} < 0.6 \frac{1}{3^{k-1}}\binom{|U|-2}{k-2}$. Therefore by Lemma 17 $w^{\sigma^2}_{vu} > 0.2 \frac{1}{3^{k-1}}\binom{|U|-2}{k-2}$ hence $\bar{w}^{\sigma^2}_{uv} = w^{\sigma^2}_{uv} - \min(0.1 \frac{1}{3^{k-1}}\binom{|U|-2}{k-2}, w^{\sigma^2}_{uv}) = \min(w^{\sigma^2}_{uv} - 0.1\frac{1}{3^{k-1}}\binom{|U|-2}{k-2}, 0) \le \min(w^{\pi^\circledast}_{uv}, 0) \le w^{\pi^\circledast}_{uv}$ using the first part of the Lemma in the penultimate inequality. □

Lemma 20.

1. $C^{\bar{w}^{\sigma^2}}(\pi^\circledast) \le (1 + O(\epsilon))\binom{k}{2} C(\pi^\circledast)$
2. $C^{\bar{w}^{\sigma^2}}(\pi^3) \le (1 + O(\epsilon))\binom{k}{2} C(\pi^\circledast)$
3. $C^{\bar{w}^{\sigma^2}}(\pi^3) - C^{\bar{w}^{\sigma^2}}(\pi^\circledast) = O(\epsilon C(\pi^\circledast))$

Proof. From the second part of Lemma 19 and Lemma 8 we conclude that

$$C^{\bar{w}^{\sigma^2}}(\pi^\circledast) \le (1 + O(\epsilon)) C^{w^{\pi^\circledast}}(\pi^\circledast) = (1 + O(\epsilon))\binom{k}{2} C(\pi^\circledast).$$

proving the first part of this Lemma.

The PTAS for FAST (Theorem 5) guarantees

$$C^{\bar{w}^{\sigma^2}}(\pi^3) \le (1 + O(\epsilon)) C^{\bar{w}^{\sigma^2}}(\pi^\circledast), \tag{6}$$

which combined with the first part of this Lemma yields the second part.

Finally the first part of Lemma 19 followed by the first part of this Lemma imply

$$C^{\bar{w}^{\sigma^2}}(\pi^3) - C^{\bar{w}^{\sigma^2}}(\pi^\circledast) \le O(\epsilon) C^{w^{\sigma^2}}(\pi^\circledast) \le O(\epsilon C(\pi^\circledast)),$$

completing the proof of the third part of this Lemma. □

Lemma 21. $d(\pi^3, \pi^{\circledast}) = O(C(\pi^{\circledast})/n^{k-2})$

Proof. By Lemma 20 we have $C^{\bar{w}^{\sigma^2}}(\pi^{\circledast}) + C^{\bar{w}^{\sigma^2}}(\pi^3) = O(C(\pi^{\circledast}))$. For any $u, v \in U$ that are a $\pi^{\circledast}/\pi^3$ inversion u, v contribute $\bar{w}^{\sigma^2}_{uv} + \bar{w}^{\sigma^2}_{vu}$ to $C^{\bar{w}^{\sigma^2}}(\pi^{\circledast}) + C^{\bar{w}^{\sigma^2}}(\pi^3)$. By Lemma 17 this is $\Omega(n^{k-2})$, proving that the number of $\pi^{\circledast}/\pi^3$ inversions $d(\pi^3, \pi^{\circledast})$ is $O(C(\pi^{\circledast})/n^{k-2})$ as desired. □

Lemma 22. *We have* $|\pi^3(v) - \pi^{\circledast}(v)| = O(\epsilon n)$ *for all* $v \in U$.

Proof. Fix $v \in U$. In this proof we write w (resp. $\bar{w}$) as a short-hand for w^{σ^2} (resp. $\bar{w}^{\sigma^2}$). Observe that there are at least $(|\pi^3(v) - \pi^{\circledast}(v)| - 1)$ vertices between $\pi^3(v)$ and $\pi^{\circledast}(v) + 1/2$ in π^3. Any such vertex u must contribute w_{uv} to one of $b^{\bar{w}}(\pi^3, v, \pi^{\circledast}(v) + 1/2)$ and $b^{\bar{w}}(\pi^3, v, \pi^3(v))$ and contribute w_{vu} to the other. By Lemma 17 and local optimality of π^3 we have

$$\begin{aligned}(|\pi^3(v) - \pi^{\circledast}(v)| - 1)\frac{(1 - 2/10)}{3^{k-1}}\binom{|U| - 2}{k - 2} &\leq b^{\bar{w}}(\pi^3, v, \pi^{\circledast}(v) + 1/2) + b^{\bar{w}}(\pi^3, v, \pi^3(v)) \\ &\leq 2b^{\bar{w}}(\pi^3, v, \pi^{\circledast}(v) + 1/2).\end{aligned}$$

Now apply Corollary 14

$$b^{\bar{w}}(\pi^3, v, \pi^{\circledast}(v) + 1/2) \leq b^{\bar{w}}(\pi^{\circledast}, v, \pi^{\circledast}(v)) + 2\sqrt{d(\pi^{\circledast}, \pi^3)}2\binom{|U| - 2}{k - 2}$$

and then recall $\sqrt{d(\pi^{\circledast}, \pi^3)} = O(\epsilon n)$ by Lemma 21 and the assumption that $OPT = O(\epsilon^2 n^k)$.

Next

$$\begin{aligned}b^{\bar{w}}(\pi^{\circledast}, v, \pi^{\circledast}(v)) &\leq (1 + O(\epsilon))b^{w^{\pi^{\circledast}}}(\pi^{\circledast}, v, \pi^{\circledast}(v)) && \text{(Second part of Lemma 19)} \\ &= (1 + O(\epsilon))b(\pi^{\circledast}, v, \pi^{\circledast}(v)) && \text{(Lemma 8)}\end{aligned} \tag{7}$$

Finally

$$\begin{aligned}b(\pi^{\circledast}, v, \pi^{\circledast}(v)) &\leq b(\sigma^1, v, \sigma^2(v)) + O(n^{k-2}(\epsilon n + \sqrt{\epsilon^2 n^2})) && \text{(Lemmas 13, 12 and 16)} \\ &= O(\epsilon n^{k-1}) && (v \in U).\end{aligned}$$

which completes the proof of the Lemma. □

Lemma 23. $C(\pi^3) \leq (1 + O(\epsilon))C(\pi^{\circledast})$.

Proof. First we claim that

$$|(C(\pi^3) - C(\pi^{\circledast})) - (C^{w^{\sigma^2}}(\pi^3) - C^{w^{\sigma^2}}(\pi^{\circledast}))| \leq E_1, \tag{8}$$

where E_1 is the number of constraints that contain one pair of vertices u, v in different order in π^3 and $\pi^{\circledast}$ and another pair $\{s, t\} \neq \{u, v\}$ with relative order in π^3, $\pi^{\circledast}$ and σ^2 not all equal. Indeed constraints ordered identically in π^3 and $\pi^{\circledast}$ contribute zero to both sides of (8), regardless of σ^2. Consider some constraint S containing a $\pi^3/\pi^{\circledast}$-inversion $\{u, v\} \subset S$. If the restrictions of the three orderings to S are identical except possibly for swapping u, v then S contributes equally to both sides of (8), proving the claim.

To bound E_1 observe that the number of inversions u, v is $d(\pi^3, \pi^{\circledast}) \equiv D$. For any u, v Lemmas 22, 16 and 18 allow us to show at most $O(\epsilon n^{k-2})$ constraints containing $\{u, v\}$ contribute to E_1, so $E_1 = O(D\epsilon n^{k-2}) = O(\epsilon C(\pi^{\circledast}))$ (Lemma 21).

Finally bound $C^{w^{\sigma^2}}(\pi^3) - C^{w^{\sigma^2}}(\pi^{\circledast}) = C^{\bar{w}^{\sigma^2}}(\pi^3) - C^{\bar{w}^{\sigma^2}}(\pi^{\circledast}) \leq O(\epsilon C(\pi^{\circledast}))$, where the equality follows from the definition of w and the inequality is the third part of Lemma 20. □

Extending Lemma 23 to a bound on the cost of π^4 is relatively straightforward. We do so and prove our theorems in Appendix C.

Acknowledgements

We would like to thank Venkat Guruswami, Claire Mathieu, Prasad Raghavendra and Alex Samorodnitsky for interesting remarks and discussions.

References

[1] N. Ailon. Aggregation of Partial Rankings, p-ratings and top-m Lists. In *18th SODA*, pages 415–424, 2007.

[2] N. Ailon and N. Alon. Hardness of Fully Dense Problems. *Inf. Comput.*, 205:1117–1129, 2007.

[3] N. Ailon, M. Charikar, and A. Newman. Aggregating Inconsistent Information: Ranking and Clustering. *J. ACM*, 55, 2008.

[4] N. Alon, W. Fernandez de la Vega, R. Kannan, and M. Karpinski. Random Sampling and Approximation of MAX-CSP Problems. In *34th ACM STOC*, pages 232–239, 2002. Journal version in J. Comput. System Sciences 67 (2003), pp. 212-243.

[5] N. Alon, D. Lokshtanov, and S. Saurabh. Fast FAST. In *36th ICALP*, pages 49–58, 2009.

[6] S. Arora, D. Karger, and M. Karpinski. Polynomial Time Approximation Schemes for Dense Instances of NP-Hard Problems. In *27th ACM STOC*, pages 284–293, 1995. Journal version in J. Comput. System Sciences 58 (1999), pp. 193-210.

[7] C. Bazgan, W. Fernandez de la Vega, and M. Karpinski. Polynomial Time Approximation Schemes for Dense Instances of the Minimum Constraint Satisfaction Problem. *Random Structures and Algorithms*, 23:73–91, 2003.

[8] M. Charikar, V. Guruswami, and R. Manokaran. Every Permutation CSP of Arity 3 is Approximation Restitant. In *24th IEEE CCC*, 2009.

[9] B. Chor and M. Sudan. A Geometric Approach to Betweenness. *SIAM J. Discrete Math.*, 11:511–523, 1998.

[10] W. Fernandez de la Vega, R. Kannan, and M. Karpinski. Approximation of Global MAX–CSP Problems. Technical Report TR06-124, Electronic Colloquim on Computation Complexity, 2006.

[11] A. Frieze and R. Kannan. Quick Approximation to Matrices and Applications. *Combinatorica*, 19:175–220, 1999.

[12] G. Gutin, E. J. Kim, M. Mnich, and A. Yeo. Ordinal embedding relaxations parameterized above tight lower bound. *CoRR/arXiv*, abs/0907.5427, 2009.

[13] M. Karpinski and W. Schudy. Linear Time Approximation Schemes for the Gale-Berlekamp Game and Related Minimization Problems. In *41st ACM STOC*, pages 313–322, 2009.

[14] M. Karpinski and W. Schudy. Faster Algorithms for Feedback Arc Set Tournament, Kemeny Rank Aggregation and Betweenness Tournament. In *Proc. 21^{st} ISAAC*, pages 3–14, 2010.

[15] C. Mathieu and W. Schudy. How to Rank with Few Errors. In *39th ACM STOC*, pages 95–103, 2007. In Submission `http://www.cs.brown.edu/~ws/papers/fast_journal.pdf`, 2009.

[16] C. Mathieu and W. Schudy. Yet Another Algorithm for Dense Max Cut: Go Greedy. In *Proc. 19th ACM-SIAM SODA*, pages 176–182, 2008.

[17] J. Opatrny. Total Ordering Problem. *SIAM J. Comput.*, 8:111–114, 1979.

[18] M. Rudelson and R. Vershynin. Sampling from Large Matrices: An Approach through Geometric Functional Analysis. *J. ACM*, 54:21, 2007.

Appendix

A Runtime analysis

By Theorem 7 the additive approximation step takes time $n^{O(1)}2^{\tilde{O}(1/\epsilon^{10})}$. There are at most $(1/\epsilon)^{t\cdot(k-1)} = 2^{\tilde{O}(1/\epsilon)}$ bucketed orderings σ^0 to try. The PTAS for FAST takes time $n^{O(1)}2^{\tilde{O}(1/\epsilon^6)}$ by Theorem 5. The overall runtime is

$$n^{O(1)}2^{\tilde{O}(1/\epsilon^{10})} + 2^{\tilde{O}(1/\epsilon)} \cdot \left(n^{O(1)} + n^{O(1)}2^{\tilde{O}(1/\epsilon^6)}\right) = n^{O(1)}2^{\tilde{O}(1/\epsilon^{10})}.$$

Derandomization increases the runtime of the two algorithms that we use as subroutines to $n^{poly(1/\epsilon)}$. There are at most $n^{t\cdot(k-1)} = n^{\tilde{O}(1/\epsilon)}$ possible sets $T_1, \ldots T_t$ that the derandomized algorithm must consider. Therefore the overall runtime is

$$(n^{poly(1/\epsilon)} + n^{poly(1/\epsilon)} \cdot 2^{\tilde{O}(1/\epsilon)} \cdot n^{poly(1/\epsilon)} = n^{poly(1/\epsilon)}.$$

B Proofs

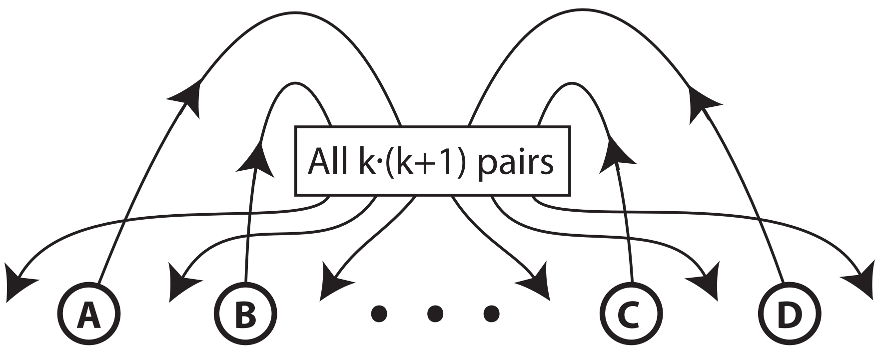

Figure 1: An illustration of fragility. For a constraint to be fragile all the illustrated single vertex moves must make any satisfied constraint unsatisfied.

Proof of Lemma 9. Fix costly vertex v. Consider picking a constraint containing v but no $\pi^*/\sigma^{\square}$-inversions one vertex at a time (starting with v). Each vertex does a $\pi^*/\sigma^{\square}$-inversion with at most $\epsilon n - 1$ other vertices, so there are at least $n - i\epsilon n$ possible choices for the vertex chosen after $1 \leq i \leq k-1$ vertices are already chosen. The total number of such constraints is therefore $(n-\epsilon n)(n-2\epsilon n)\cdots(n-(k-1)\epsilon n)/(k-1)! \geq \binom{n-1}{k-1}(1-\epsilon\binom{k}{2})$. It follows that the total number of constraints containing v and at least one $\pi^*/\sigma^{\square}$-inversion is at most $\epsilon\binom{k}{2}\binom{n-1}{k-1}$.

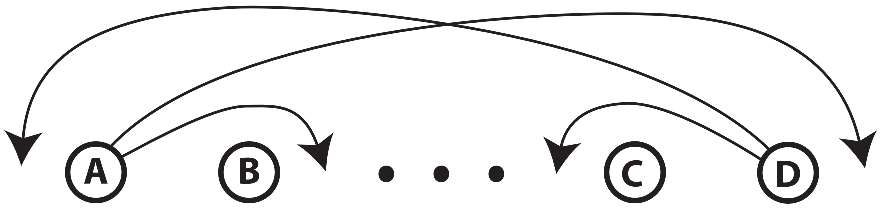

Figure 2: An illustration of weak fragility. For a constraint to be weak fragile all the illustrated single vertex moves must make any satisfied constraint unsatisfied.

Therefore for any costly v we have

$$2\binom{k}{2}\epsilon\binom{n-1}{k-1} \le b(\sigma^{\Box}, v, \sigma^{\Box}(v)) \le b(\pi^*, v, \pi^*(v)) + \epsilon\binom{k}{2}\cdot\binom{n-1}{k-1}.$$

Rearranging we get

$$b(\pi^*, v, \pi^*(v)) \ge 2\binom{k}{2}\epsilon\binom{n-1}{k-1} - \epsilon\binom{k}{2}\cdot\binom{n-1}{k-1} = \epsilon\binom{k}{2}\cdot\binom{n-1}{k-1}.$$

Finally observe that $kC(\pi^*) = \sum_v b(\pi^*, v, \pi^*(v)) \ge (\text{number costly})\epsilon\binom{k}{2}\binom{n-1}{k-1}$, completing the proof. $\Box$

Proof of Lemma 10 in the general weak fragile case. Observe that the quantity in brackets in (1) is at least 1 for every Q that either has all $k-1$ vertices between p and p' in σ^2 or has one vertex between them and the remaining $k-2$ either all before or all after. To lower-bound the number of such Q we consider two cases.

If $|B| \ge |V|/3$ then the number of such Q is at least $\binom{|B|}{k-1} = \frac{|B|}{k-1}\binom{|B|-1}{k-2} \ge \frac{|B|}{2\cdot(k-1)3^{k-2}}\binom{n-2}{k-2}$ for sufficiently large n.

If $|B| < |V|/3$ then either at least $|V|/3$ vertices are before or at least $|V|/3$ vertices are after hence the number of such Q is at least $|B|\binom{|V|/3}{k-2} \ge \frac{|B|}{2\cdot 3^{k-2}}\binom{n-2}{k-2} \ge \frac{|B|}{(k-1)\cdot 3^{k-1}}\binom{n-2}{k-2}$ for sufficiently large n. $\Box$

Proof of Lemma 11. Focus on some $v \in V$ and $p \in \mathcal{P}(v)$. From the definition of out-of-place and Lemma 10 we have

$$b(\sigma^{\Box}, v, \sigma^{\Box}) + b(\sigma^{\Box}, v, p) \ge \frac{6\binom{k}{2}3^{k-1}\epsilon n}{(n-1)3^{k-1}}\binom{n-1}{k-1} \ge 6\epsilon\binom{k}{2}\binom{n-1}{k-1}$$

for any v-out of place p. Next recall that for non-costly v we have

$$b(\sigma^{\Box}, v, \sigma^{\Box}(v)) < 2\binom{k}{2}\epsilon\binom{n-1}{k-1} \tag{9}$$

hence

$$b(\sigma^{\Box}, v, p) > 4\binom{k}{2}\epsilon\binom{n-1}{k-1} \tag{10}$$

for any v-out of place p.

Recall that

$$\hat{b}(v,p) = \frac{\binom{n}{k-1}}{t} \sum_{i:v\notin T_i} c(\sigma^0{}_{T_i} \mid v \mapsto p)$$

for any p. Each term of the sum is a 0/1 random variable with mean $\mu(p) = \frac{1}{\binom{n}{k-1}} \sum_{Q\in\binom{V}{k-1}:v\notin Q} c(\sigma^{\Box}{}_Q \mid v\mapsto p) = \frac{1}{\binom{n}{k-1}} b(\sigma^{\Box}, v, p)$. Therefore $\mathbf{E}\left[\hat{b}(v,p)\right] = b(\sigma^{\Box}, v, p)$. We can bound $\mu(\sigma^{\Box}(v)) \leq 2\binom{k}{2}\epsilon\binom{n-1}{k-1}/\binom{n}{k-1} \equiv M$ using (9). For any v-out of place p we can bound $\mu(p) \geq 2M$ by (10).

We can bound the probability that sum in $\hat{b}(v, \sigma^{\Box}(v))$ is at least $(1+1/3)Mt$ using a Chernoff bound as

$$\exp(-(1/3)^2 Mt/3) \leq \exp\left(-\frac{1}{9}\cdot\frac{1}{\binom{n}{k-1}}\cdot 2\binom{k}{2}\epsilon\binom{n-1}{k-1}\cdot\frac{14\ln(40/\epsilon)}{\binom{k}{2}\epsilon}\cdot\frac{1}{3}\right) \leq \epsilon/40$$

for sufficiently large n. Similarly for any v-out of place p we can bound the probability that $\hat{b}(v,p)$ is at most $(1-1/3)Mt$ by $\exp(-(1/3)^2Mt/2) \leq (\epsilon/40)^3$. Therefore by union bound the probability of some v-out of place p having $\hat{b}(v,p)$ too small is at most $\epsilon^2/40^3 \leq \epsilon/40$. Clearly $4(1-1/3) \geq 2(1+1/3)$ so each vertex v is out of place with probability at least $\epsilon/20$. A Markov bound completes the proof. □

Proof of Lemma 13. Fix σ, σ', T, v, p and p'. Let L (resp. R) denote the vertices in T that do left to right (resp. right to left) (σ, p, σ', p')-crossings. It is easy to see that a constraint $\{v\} \cup Q$, $Q \in \binom{T\setminus\{v\}}{k-1}$ contributes identically to $b(\sigma, v, p)$ and $b(\sigma', v, p')$ unless it is one of the following two types:

1. Q and $(L \cup R)$ have non-empty intersection (or)
2. Q contains a σ/σ'-inversion $\{s,t\}$.

The first part of the Lemma follows easily.

We prove the second part as a consequence of the first part. Observe that $|L| = |R| + (\mathit{net\ flow})$. Assume w.l.o.g. that $(\mathit{net\ flow}) \geq 0$. Observe that every pair $v \in L$ and $w \in R$ are a σ/σ'-inversion, hence $d(\sigma,\sigma') \geq |L|\cdot|R| = (|R| + (\mathit{net\ flow}))|R| \geq |R|^2$. We conclude that

$$(\mathit{number\ of\ crossings}) = |L| + |R| = 2|R| + (\mathit{net\ flow}) \leq 2\sqrt{d(\sigma,\sigma')} + (\mathit{net\ flow}). \tag{11}$$

We now bound the second term of the first part of the Lemma:

$$\begin{aligned}\binom{n-3}{k-3} d(\sigma,\sigma') &= \binom{n-2}{k-2}\sqrt{d(\sigma,\sigma')}\cdot\frac{k-2}{n-2}\cdot\sqrt{d(\sigma,\sigma')} \\ &\leq \binom{n-2}{k-2}\sqrt{d(\sigma,\sigma')}\cdot(k-2)\frac{\sqrt{n(n-1)/2}}{n-2} \leq (k-2)\binom{n-2}{k-2}\sqrt{d(\sigma,\sigma')} \end{aligned}\tag{12}$$

for sufficiently large n.

The second part of the Lemma follows from substituting (11) and (12) into the first part of the Lemma. □

Proof of Lemma 17. We prove the more interesting lower-bound and leave the straightforward proof of the upper bound to the reader. Fix $u, v \in U$. We consider two cases.

If there are at least $|U|/3$ vertices between u and v in σ^2 then we note that by weak fragility every constraint $S \supseteq \{u, v\}$ with all vertices in S between u and v in σ^2 contributes at least 1 to $w_{uv} + w_{vu}$. Therefore $w_{uv} + w_{vu} \geq \binom{|U|/3}{k-2} \geq \frac{1}{2\cdot 3^{k-2}}\binom{n-2}{k-2}$ for sufficiently large n and small ϵ.

If there are at most $|U|/3$ vertices between u and v in σ^2 then consider constraints with all their vertices either all before or all after u and v. We note that by weak fragility each such constraint $S \supseteq \{u, v\}$ contributes at least 1 to $w_{uv} + w_{vu}$. There are clearly either at least $|U|/3$ vertices before u and v or at least $|U|/3$ vertices after, hence at least $\binom{|U|/3}{k-2} \geq \frac{1}{2\cdot 3^{k-2}}\binom{n-2}{k-2}$ constraints for sufficiently large n and small ϵ.

We conclude that $w_{uv} + w_{vu} \geq \frac{1}{2\cdot 3^{k-2}}\binom{n-2}{k-2} \geq \frac{1}{3^{k-1}}\binom{n-2}{k-2}$. The Lemma follows from the definition of $\bar{w}$. $\square$

C Analysis of π^4

In this section we prove Theorem 3 and 4:

$$C(\pi^4) \leq (1 + O(\epsilon))OPT \tag{13}$$

and

$$d(\pi^4, \pi^*) = O\left(\frac{OPT}{poly(\epsilon)n^{k-2}}\right). \tag{14}$$

If $OPT > \epsilon^4 n^k$ then, as discussed in Section 3, Equation (13) follows from the algorithm and the additive error guarantee. Equation (14) is vacuous in this case. It remains to show (13) and (14) in the case that that Sections 4-6 dealt with: $OPT \leq \epsilon^4 n^k$.

First we prove (13). We consider three contributions to these costs separately: constraints with 0, 1, or 2+ vertices in $V \setminus U$.

The contribution of constraints with 0 vertices in $V \setminus U$ to the left- and right-hand sides of (13) are clearly $C(\pi^3)$ and $C(\pi^{\circledast})$ respectively. We showed $C(\pi^3) \leq C(\pi^{\circledast}) + O(\epsilon)C(\pi^{\circledast})$ in Lemma 23.

Second we consider the contribution of constraints with exactly 1 vertex in $V \setminus U$. Consider some $v \in V \setminus U$. We want to compare $b(\pi^3, v, \sigma^4(v))$ and $b((\pi^*{}_U), v, \pi^*(v))$. Let p be the half-integer so that $Ranking(v \mapsto p \mid \pi^{\circledast}{}_U) = Ranking(v \mapsto \pi^*(v) \mid \pi^*{}_U)$. The algorithm's greedy choice minimizes $b(\pi^3, v, \sigma^4(v))$ so $b(\pi^3, v, \sigma^4(v)) \leq b(\pi^3, v, p)$. Now using Lemmas 13 and 21 we have $b(\pi^3, v, p) \leq b(\pi^{\circledast}, v, p) + O(\sqrt{d(\pi^3, \pi^{\circledast})}n^{k-2}) = b(\pi^{\circledast}, v, p) + O(\sqrt{OPT/n^k}n^{k-1})$. Note $b(\pi^{\circledast}, v, p) = b((\pi^*{}_U), v, \pi^*(v))$. Let $\gamma = OPT/n^k$. We conclude by Lemma 15 that the contribution of constraints with exactly 1 vertex in $V \setminus U$ is $O(|V \setminus U|\sqrt{OPT/n^k}n^{k-1}) = O(\frac{\gamma^{3/2}n^k}{\epsilon}) = O(\epsilon OPT)$.

Finally by Lemma 15 there are at most $|V \setminus U|^2 n^{k-2} = O((\frac{\gamma}{\epsilon})^2 n^2 n^{k-2}) = O(\epsilon^2 OPT)$ constraints containing two or more vertices from $V \setminus U$.

This ends the proof of (13).

Finally we prove (14). By Lemma 21 we have

$$d(\pi^3, \pi^{\circledast}) = O(C(\pi^{\circledast})/n^{k-2}).$$

Finally a pair of vertices can only be counted in $d(\pi^4, \pi^*)$ but not $d(\pi^3, \pi^{\circledast})$ if at least one of the vertices is in the ambiguous set $V \setminus U$. By Lemma 15 $|V \setminus U| = O(\frac{n}{\epsilon} \cdot \frac{OPT}{n^k})$ so there at most $O(n \cdot \frac{OPT}{\epsilon n^{k-1}}) = O(\frac{OPT}{\epsilon n^{k-2}})$ such pairs.