RC22108 (W0107-008) July 3, 2001
Mathematics

IBM Research Report

The probabilistic relationship between the assignment and asymmetric traveling salesman problems

Alan Frieze
Department of Mathematical Sciences,
Carnegie Mellon University,
Pittsburgh PA 15213

Gregory B. Sorkin
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

The probabilistic relationship between the assignment and asymmetric traveling salesman problems

Alan Frieze* Gregory B. Sorkin†

25 June 2001

Abstract

We consider the gap between the cost of an optimal assignment in a complete bipartite graph with random edge weights, and the cost of an optimal traveling salesman tour in a complete directed graph with the same edge weights. Using an improved "patching" heuristic, we show that with high probability the gap is $O((\ln n)^2/n)$, and that its expectation is $\Omega(1/n)$. One of the underpinnings of this result is that the largest edge weight in an optimal assignment has expectation $\Theta(\ln n/n)$. A consequence of the small assignment-TSP gap is an $e^{\tilde{O}(\sqrt{n})}$-time algorithm which, with high probability, exactly solves a random asymmetric traveling salesman instance. In addition to the assignment-TSP gap, we also consider the expected gap between the optimal and second-best assignments; it is at least $\Omega(1/n^2)$ and at most $O(\ln n/n^2)$.

1 Introduction

The *Assignment Problem* (AP) is the problem of finding a minimum-weight perfect matching in an edge-weighted bipartite graph. An instance of the AP can be specified by an $n \times n$ matrix $C = (C(i,j))$; here $C(i,j)$ represents the weight (or "cost") of the edge between $i \in X$ and $j \in Y$, where X and Y are disjoint copies of $[n] = \{1,2,\ldots,n\}$ and X is the set of "left vertices" and Y is the set of "right vertices" in the complete bipartite graph $K_{X,Y}$. The AP can be stated in terms of the matrix C as follows. Find a permutation π of $[n] = \{1,2,\ldots,n\}$ that minimizes $\sum_{i=1}^n C(i,\pi(i))$. Let $\mathrm{AP}(C)$ be the optimal value of the instance of the AP specified by C.

The *Asymmetric Traveling-Salesman Problem* (ATSP) is the problem of finding a Hamiltonian circuit of minimum weight in an edge-weighted directed graph. An instance of the ATSP can be specified by an $n \times n$ matrix $C = (C(i,j))$ in which $C(i,j)$ denotes the weight of edge (i,j). The ATSP can be stated in terms of the matrix C as follows: find a *cyclic* permutation π of $[n]$ that minimizes $\sum_{i=1}^n C(i,\pi(i))$; here a cyclic permutation is one whose cycle structure consists of a single cycle. Let $\mathrm{ATSP}(C)$ be the optimal value of the instance of the ATSP specified by C.

*Research supported by NSF grant CCR-9818411, Department of Mathematical Sciences, Carnegie Mellon University, Pittsburgh PA15213, e-mail alan@random.math.cmu.edu

†Department of Mathematical Sciences, IBM T.J. Watson Research Center, Yorktown Heights NY 10598, e-mail sorkin@watson.ibm.com

It is evident from the parallelism between the above two definitions that $\mathrm{AP}(C) \leq \mathrm{ATSP}(C)$. The ATSP is NP-hard, whereas the AP is solvable in time $O(n^3)$. Several authors, e.g. Balas and Toth [5], have investigated whether the AP can be used effectively in a branch-and-bound method to solve the ATSP and have observed that the AP gives extremely good bounds on random instances.

Karp was able to explain this in an important paper [15]. He assumed that the entries of C were independent uniform [0,1] random variables, and proved the surprising result that

$$\mathbf{E}(\mathrm{ATSP}(C) - \mathrm{AP}(C)) = o(1). \tag{1}$$

Since **whp**[1] $\mathrm{AP}(C) > 1$ we see that this rigorously explains the quality of the assignment bound, a significant plus for probabilistic analysis. Karp proved (1) constructively, analysing an $O(n^3)$ *patching* heuristic that transformed an optimal Assignment Problem solution into a good TSP solution. Karp and Steele [16] simplified and sharpened this analysis, and Dyer and Frieze [8] improved the error bound in (1) to $O\left(\frac{(\ln n)^4}{n \ln\ln n}\right)$. Our first theorem sharpens this further.

Theorem 1 *Over random cost matrices C,*

$$\begin{aligned} \textbf{whp}, \qquad\qquad \mathrm{ATSP}(C) - \mathrm{AP}(C) &\leq c_1 \frac{(\ln n)^2}{n} \\ &\textit{and} \\ \mathbf{E}(\mathrm{ATSP}(C) - \mathrm{AP}(C)) &\geq \frac{c_0}{n}. \end{aligned}$$

In this paper, $c_0, c_1, \ldots$ are positive absolute constants whose precise values are not too important to us. The proofs of Theorems 1–4 constitute the body of the paper.

As in previous works, we will prove the upper bound in Theorem 1 by analysing an $O(n^3)$ heuristic which patches an optimal AP solution into a good ATSP solution. We note a related discretized result of Frieze, Karp and Reed [12], who consider the $C(i,j)$ to be random positive *integers* chosen from a range $[0, L = L(n)]$, and determine for what functions $L(n)$ one has ATSP = AP **whp**.

Karp and Steele showed that **whp** the greatest cost of an edge used in the optimal assignment was $O\left(\frac{(\ln n)^2}{n}\right)$; our next theorem improves upon this. Let $C_{\max} = C_{\max}(C)$ denote the maximum cost of an edge used in an optimal assignment.

Theorem 2 **Whp** *over random cost matrices C,*

$$(1 - o(1))\frac{\ln n}{n} \leq C_{\max} \leq c_2 \frac{\ln n}{n}.$$

It is perhaps of interest to estimate the expected difference Δ_1 between the cheapest and second-cheapest assignments. (Since 1 in n permutations is cyclic, it is plausible that the ATSP might typically be the n'th cheapest assignment, providing one reason that gaps between various cheap assignments are a natural object of study.)

[1] with high probability, i.e., with probability 1-o(1) as $n \to \infty$

Theorem 3 *Over random cost matrices C,*

$$\frac{1}{n^2}(1 - o(1)) \leq \mathbf{E}(\Delta_1) \leq c_3 \frac{\ln n}{n^2}.$$

The algorithm with the best known worst-case time for solving the ATSP exactly is the $O(n^2 2^n)$ dynamic programming algorithm of Held and Karp [13]. The next theorem describes a modest, probabilistic improvement.

Theorem 4 **Whp**, *a random instance of the* ATSP *can be solved exactly in time $e^{\tilde{O}(\sqrt{n})}$.*

Here $\tilde{O}$ is the standard notation for ignoring logarithmic factors.

2 Analysis of the Assignment Problem

In this section we will prove Theorem 2. The difficult part of the proof — showing that the longest edge in an optimal assignment has length $O(\ln n)$ — has its essence in Lemma 5 below.

Define the k-neighborhood of a vertex to be the k vertices nearest it, where distance is given by the matrix C; let the k-neighborhood of a set be the union of the k-neighborhoods of its vertices. In particular, for a complete bipartite graph $K_{X,Y}$ and any $S \subseteq X$, $T \subseteq Y$,

$$N_k(S) \doteq \{y \in Y : \exists s \in S \text{ s.t. } (s, y) \text{ is one of the } k \text{ shortest arcs out of } s\}, \tag{2}$$
$$N_k(T) \doteq \{x \in X : \exists t \in T \text{ s.t. } (x, t) \text{ is one of the } k \text{ shortest arcs into } t\}. \tag{3}$$

Given the complete bipartite graph $K_{X,Y}$, any permutation $\pi : X \to Y$ has an associated matching $M_\pi = \{(x, y) : x \in X,\ y \in Y,\ y = \pi(x)\}$. Given a cost matrix C and permutation π, define the digraph

$$\vec{D} = \vec{D}_{C,\pi} = (X \cup Y,\ \vec{E}) \tag{4}$$

consisting of *backwards* matching edges and forward "short" edges:

$$\vec{E} = \{(y, x) : y \in Y,\ x \in X,\ y = \pi(x)\} \cup \{(x, y) : x \in X,\ y \in N_{40}(x)\} \cup \{(x, y) : y \in Y,\ x \in N_{40}(y)\}. \tag{5}$$

Lemma 5 **Whp** *over random cost matrices C, for every permutation π, the (unweighted) diameter of $\vec{D} = \vec{D}_{C,\pi}$ is at most $3 \log_4 n$.*

Proof. For $S \subseteq X$, $T \subseteq Y$, let

$$N_{\vec{D}}(S) = \{y \in Y : \exists s \in S \text{ such that } (s, y) \in \vec{E}\},$$
$$N_{\vec{D}}(T) = \{x \in X : \exists t \in T \text{ such that } (x, t) \in \vec{E}\}.$$

We first prove that **whp**, for all $S \subseteq X$ with $|S| \le \lceil n/5 \rceil$, $|N_{\vec{D}}(S)| \ge 4|S|$. (Note that only the cheap edges out of S, and not the matching edges into it, are involved here.)

$$\begin{aligned}
\mathbf{Pr}(\exists S:\ |S| \le \lceil n/5 \rceil\,,\ |N_{\vec{D}}(S)| < 4|S|) &\le \sum_{s=1}^{\lceil n/5 \rceil} \binom{n}{s}\binom{n}{4s}\left(\frac{\binom{4s}{40}}{\binom{n}{40}}\right)^s \\
&\le \sum_{s=1}^{\lceil n/5 \rceil} \left(\frac{ne}{s}\right)^s \left(\frac{ne}{4s}\right)^{4s} \left(\frac{4s}{n}\right)^{40s} \\
&= \sum_{s=1}^{\lceil n/5 \rceil} \left(\frac{e^5 4^{36} s^{35}}{n^{35}}\right)^s \\
&= o(1). \qquad (6)
\end{aligned}$$

Similarly, **whp**, for all $T \subseteq Y$ with $|T| \le \lceil n/5 \rceil$, $|N_{\vec{D}}(T)| \ge 4|T|$. (Again only the cheap edges, not the matching edges, are involved.)

In the remainder of this proof, assume that we are in the high-probability "good" case, in which all small sets S and T expand.

Now, choose an arbitrary $x \in X$, and define $S_0, S_1, S_2, \ldots$, by

$$S_0 = \{x\} \text{ and } S_i = \pi^{-1}(N_{\vec{D}}(S_{i-1})).$$

Since we are in the good case, $|S_i| \ge 4|S_{i-1}|$ provided $|S_{i-1}| \le n/5$, and so there exists a smallest index $i_S - 1 \le \log_4(n/5) \le \log_4 n - 1$ such that $|S_{i_S-1}| > n/5$. Arbitrarily discard vertices from S_{i_S-1} to create a smaller set S'_{i_S-1} with $|S'_{i_S-1}| = \lceil n/5 \rceil$, so that $S'_{i_S} = N_{\vec{D}}(S'_{i_S-1})$ has cardinality $|S'_{i_S}| \ge 4|S'_{i_S-1}| \ge 4n/5$.

Similarly, for an arbitrary $y \in Y$, define $T_0, T_1, \ldots$, by

$$T_0 = \{y\} \text{ and } T_i = \pi(N_{\vec{D}}(T_{i-1})).$$

Again, we will find an index $i_T \le \log_4 n$ whose modified set has cardinality $|T'_{i_T}| \ge 4n/5$.

With both $|S'_{i_S}|$ and $|T'_{i_T}|$ larger than $n/2$, there must be some $x' \in S'_{i_S}$ for which $y' = \pi(x') \in T'_{i_T}$. This establishes the existence of a walk and hence a path of length at most $2(i_S + i_T) \le 2\log_4 n$ from x to y in $\vec{D}$.

We have proved there is a short path from any $x \in X$ to any $y \in Y$. A short path from x to x' both in X can be formed by going from x to $y = \pi(x')$ and appending the backward edge to x'; a path from y to x' by starting with the backward edge from y to $x = \pi^{-1}(y)$ and then pursuing a path to x'; and a path from y to y' by taking a path from y to $x' = \pi^{-1}(y')$ and discarding its final backward edge. □

We will also need the following inequality, Lemma 4.2(b) of [11].

Lemma 6 *Suppose that $k_1 + k_2 + \cdots + k_M \le a \ln N$, and $Y_1, Y_2, \ldots, Y_M$ are independent random variables with Y_i distributed as the k_ith minimum of N independent uniform [0,1] random variables. If $\lambda > 1$ then*

$$\mathbf{Pr}\left(Y_1 + \cdots + Y_M \ge \frac{\lambda a \ln N}{N+1}\right) \le N^{a(1+\ln\lambda-\lambda)}.$$

Let the weight of a forward edge (x,y) be $C(x,y)$ and the weight of a backwards edge (y,x) be $-C(x,y)$.

Lemma 7 **Whp** *over random C, for all π, the weighted diameter of $\vec{D} = \vec{D}_{C,\pi}$ is $\leq c_2\frac{\ln n}{n}$.*

Proof. Let

$$Z_1 = \max\left\{\sum_{i=0}^{k} C(x_i,y_i) - \sum_{i=0}^{k-1} C(y_i,x_{i+1})\right\}, \tag{7}$$

where the maximum is over sequences $x_0, y_0, x_1, \ldots, x_k, y_k$ where (x_i, y_i) is one of the 40 shortest arcs leaving x_i for $i = 0, 1, \ldots, k \leq k_0 = \lceil 3\log_4 n \rceil$.

We estimate the probability that Z_1 is large. Indeed, for any $\zeta > 0$ we have

$$\mathbf{Pr}\left(Z_1 \geq \zeta\frac{\ln n}{n}\right) \leq \sum_{k=1}^{k_0} n^{2k+1}\frac{1}{(n-1)^k}\frac{1}{k!}\times$$
$$\int_{y=0}^{\infty}\left[\left(\frac{y\ln n}{n}\right)^k \sum_{\rho_1+\cdots+\rho_k\leq 40k} q(\rho_1,\ldots,\rho_k;\zeta+y)\right]dy$$

where

$$q(\rho_1,\ldots,\rho_k;\eta) = \mathbf{Pr}\left(X_1 + \cdots + X_k \geq \eta\frac{\ln n}{n}\right),$$

$X_1, \ldots, X_k$ are independent and X_j is distributed as the ρ_jth minimum of $n-1$ uniform [0,1] random variables.

Explanation: *We have $\leq n^{2k+1}$ choices for the sequence $x_0, y_0, x_1, \ldots, x_k, y_k$. The term $\frac{1}{k!}\left(\frac{y\ln n}{n}\right)^k$ bounds the probability that the sum of k independent uniforms, $C(y_0,x_1) + \cdots + C(y_{k-1},x_k)$, is at most $\frac{y\ln n}{n}$. We integrate over y. $\frac{1}{n-1}$ is the probability that (x_i,y_i) is the ρ_ith shortest edge leaving x_i, and these events are independent for $0 \leq i \leq k-1$. The final summation bounds the probability that the associated edge lengths sum to at least $\frac{(\zeta+y)\ln n}{n}$.*

It follows that if ζ is sufficiently large then, for all $y \geq 0$, $q(\rho_1,\ldots,\rho_k;\zeta+y) \leq n^{-(\zeta+y)/2}$ and

$$\begin{aligned}\mathbf{Pr}\left(Z_1 \geq \zeta\frac{\ln n}{n}\right) &\leq 2n^{1-\zeta/2}\sum_{k=1}^{k_0}\frac{(\ln n)^k}{k!}\binom{40k-1}{k-1}\int_{y=0}^{\infty} y^k n^{-y/2}dy \\ &\leq 2n^{1-\zeta/2}\sum_{k=1}^{k_0}\frac{(\ln n)^k}{k!}\left(\frac{40e}{\ln n}\right)^{k+1}\Gamma(k+1) \\ &\leq 2n^{1-\zeta/2}(40e)^{k_0+2} \\ &= o(1).\end{aligned}$$

Similarly, **whp** $Z_2 \leq \zeta\frac{\ln n}{n}$, where Z_2 is the maximum of the RHS of expression (7) over sequences where (x_i,y_i) is one of the 40 shortest arcs entering y_i.

An alternating path P from $x \in X$ to $y \in Y$ defined in Lemma 5 can be decomposed into a path P_1 from x to $y' = \pi(x')$, the edge (y',x') and a path P_2 from x' to y. The cost of P is at most the sum of the costs of P_1, P_2 which is at most $Z_1 + Z_2 \leq 2\zeta\frac{\ln n}{n}$ **whp**.

We have proved there is a cheap path from any $x \in X$ to any $y \in Y$. Extending this to cheap paths between any two vertices is just as in the proof of Lemma 5. $\square$

We can now prove Theorem 2, repeated here for convenience.

Theorem 2 **Whp** *over random cost matrices* C,

$$(1-o(1))\frac{\ln n}{n} \leq C_{\max} \leq c_2 \frac{\ln n}{n}.$$

Proof. The lower bound follows easily from the fact that $\frac{\ln n}{n}$ is the threshold probability for a random bipartite graph to have a perfect matching, as shown by Erdős and Rényi [10].

For the upper bound, define $\vec{D} = \vec{D}_{C,\pi}$ as per (4) and (5). From the preceding lemma, we can assume the existence of a cheap alternating path from any x to $\pi(x)$,

$$x = x_0, y_0, x_1, y_1, \ldots, x_k, y_k = \pi(x), \qquad k \leq 2k_0 \tag{8}$$

consisting of cheap forward edges and backwards matching edges. Appending a final backwards edge $(\pi(x), x)$ creates an alternating cycle.

If any edge in the optimal matching has cost $C(x, \pi(x)) > \frac{c_2 \ln n}{n}$, then the canonical alternating cycle on x has reverse (matching) edge cost at least $\frac{c_2 \ln n}{n}$ yet **whp** has forward edge cost $Z_x \leq \frac{c_2 \ln n}{n}$. From the original matching, delete the alternating cycle's matching edges and replace them with its forward edges to produce a new matching of smaller cost — contradicting optimality. Thus **whp**, every edge in an optimal matching has cost $C(x, \pi(x)) \leq \frac{c_2 \ln n}{n}$. $\square$

3 Analysis of the Traveling Salesman Problem

Our goal in this section is to prove Theorem 1, recalled here for convenience.

Theorem 1 *Over random cost matrices* C,

$$\begin{array}{rrcl} \textbf{whp}, & \mathrm{ATSP}(C) - \mathrm{AP}(C) & \leq & c_1 \dfrac{(\ln n)^2}{n} \\ & & \textit{and} & \\ & \mathbf{E}(\mathrm{ATSP}(C) - \mathrm{AP}(C)) & \geq & \dfrac{c_0}{n}. \end{array}$$

We prove the Theorem's first assertion in sections through 3.3, and the second in section 3.4.

If $(i, \pi(i))$, $i \in X$, is a perfect matching of $K_{X,Y}$, then $(i, \pi(i))$ defines a *permutation digraph*, i.e., a set of vertex-disjoint directed cycles that cover all n vertices of the complete directed graph $\vec{K}_n$ associated with $K_{X,Y}$. The *size* $|\pi|$ of π is the number of cycles in the permutation.

Similarly a near-perfect matching gives rise to a near-permutation digraph (NPD), i.e., a digraph obtained from a permutation digraph by removing one edge. Thus an NPD Γ consists of any number of directed cycles and a single directed path PATH(Γ).

The edges (i, j) will be coloured: Red for $C(i,j) \in [0, c_2 \frac{\ln n}{n}]$; Blue for $C(i,j) \in (c_2 \frac{\ln n}{n}, 2c_2 \frac{\ln n}{n}]$; Green for $C(i,j) \in (2c_2 \frac{\ln n}{n}, 3c_2 \frac{\ln n}{n}]$; and Black otherwise.

We will use a *three phase* method as outlined below:

Phase 1. Solve the assignment problem to obtain an optimal assignment π and perfect matching M_π in $K_{X,Y}$; **whp**, only Red edges are used.

Phase 2. **Whp**, at cost $O(\frac{(\ln n)^2}{n})$ we increase the minimum cycle length in the permutation digraph to at least $n_0 = \lceil \frac{n \ln \ln n}{\ln n} \rceil$. We use Red and Blue edges.

Phase 3. **Whp**, at cost $O(\frac{(\ln n)^2}{n})$ we convert the **Phase 2** permutation digraph to a tour. We use Green edges.

3.1 Phase 1

That only Red edges are used in an optimal assignment is immediate from Theorem 2. Furthermore, given the optimal assignment and conditional on it only using Red edges, the edges which are not Red can be thought of as having *independent* lengths, uniform in $[c_2 \frac{\ln n}{n}, 1]$.

Also, **whp**, the optimal assignment π's associated permutation digraph Π_1 is of size $|\Pi_1| \leq 2 \ln n$. This holds because π is a random permutation; we will elaborate on this in Phase 2.

3.2 Phase 2

In this phase, to increase the minimum cycle length in the PD, we will deal with each small cycle in turn. Let us describe the essence of how one small cycle of a PD is repaired, setting aside the combinatorial and probabilistic issues. One edge (a, b) of the cycle is chosen. From vertex a, an alternating path is grown, alternating forward non-PD edges (starting with an edge out of a) with PD edges traversed backwards. From b a similar path is grown, alternating non-PD edges traversed backwards (starting with an edge into b) with PD edges traversed forwards. The a-path, followed by the edge joining its terminal to that of the b-path, followed by the reversed b-path, followed by the edge (b, a), defines an alternating cycle. The "sum" of this cycle and the original PD is a new PD. If the two paths, and the edge bridging their endpoints, are cheap, the new PD is not much more expensive than the old one. How does the new structure compare with the old one?

Consider the sum of the original PD and the path on a, as the path grows. When the path enters a vertex on a PD cycle and exits from the vertex's predecessor, the sum (an NPD) includes a directed path starting at a and going the long way around through the cycle. When the next cycle is struck, it is added to this string. If a cycle is hit a second time ("the string crosses itself"), the loop formed splits off as a cycle, and the path continues on. Similarly from b. As long as no cycles split off are small, and either a or b hits at least one large cycle, the new cycle containing a and b, and any other new cycles formed, will be large. We will try to arrange for this to be the case, otherwise declaring the attempt a failure.

If we fail for a cycle, then the entire algorithm fails. If we succeed, we proceed to the next small cycle, until all small cycles are repaired.

Of course the "new" PD of one case becomes the "original" PD of the next one, and the most difficult part of the analysis will be to avoid conditioning that might be introduced by this evolving cycle structure. (We will rely on the fact that a PD is induced by a bipartite matching *when the two sets of vertices are put into correspondence by a labelling*, and until that labelling is established, the PD and the matching are in a sense independent.)

The first detail is the construction of the cheap alternating paths out of vertices a and b. Paths alternating with respect to a PD as described above are — equivalently — alternating with respect to the corresponding bipartite matching. We begin by finding a cheap "alternating tree" (actually a directed acyclic graph, or DAG, but no matter), rooted at a, containing many cheap alternating paths. After doing the same for b, we (hopefully) find some cheap edge between an a-leaf and a b-leaf, and we use the paths selected by these leaves.

To define the trees, recall the definitions (2) and (3) of $N_k(S)$ and $N_k(T)$. For the remainder of this section let K be a suitably large constant. Let $E_K = \{(x,y) : \; y \in N_K(x) \text{ or } x \in N_K(y)\}$.

Lemma 8 *For any fixed K,* **whp** *over random matrices C, every set of $s \leq s_0 = \frac{\ln n}{2 \ln \ln n}$ vertices, spans at most s edges from E_K.*

Proof. Since K is large, we know that **whp** every edge in E_K has length at most $2K\frac{\ln n}{n}$. So the probability there exists a small set S containing $|S|+1$ edges is at most

$$o(1) + \sum_{s=1}^{s_0} \binom{n}{s} \binom{s(s-1)}{s+1} \left(2K\frac{\ln n}{n}\right)^{s+1} \leq o(1) + \sum_{s=1}^{s_0} \frac{s}{n}(2e^2 K \ln n)^{s+1} = o(1).$$

□

Lemma 9 **Whp** *over random matrices C, for all $S \subseteq X$, $T \subseteq Y$, with $|S|, |T| \leq n^{3/4}$,*

$$|N_K(S)| \geq (K-2)|S| \text{ and } N_K(T)| \geq (K-2)|T|. \tag{9}$$

Proof. Just as in deriving (6),

$$\begin{aligned} &\mathbf{Pr}(\exists S \text{ or } T : \neg(9)) \\ &\leq 2 \sum_{s=1}^{n^{3/4}} \binom{n}{s} \binom{n}{(K-2)s} \left(\frac{\binom{(K-2)s}{K}}{\binom{n}{K}} \right)^s \\ &= o(1). \end{aligned}$$

□

As before, we use this expansion to create many short alternating paths. Let a bijection (matching) ρ_i between X and Y be given, and let one matching edge (a_i, b_i) be specified. Define branching factors

$$r_1 = \lceil K \ln n \rceil \qquad \text{and} \qquad r_t = K$$

for a first generation $t=1$ and for all subsequent generations $t \geq 2$ respectively. For each i we construct a pair of "trees" (actually DAGs), S_i rooted at a_i and T_i at b_i, which we will use to modify bijection $\rho = \rho_i$. Their depth-t nodes consist of the sets $S_i^{(t)}$ and $T_i^{(t)}$ respectively. The depth-0 node sets are the singletons

$$S_i^{(0)} = \{a_i\} \text{ and } T_i^{(0)} = \{b_i\}.$$

Define

$$s_0 = \frac{\ln n}{12 \ln\ln n},$$

and for $1 \le t \le s_0$ let

$$S_i^{(t)} = \rho^{-1}(N_{r_t}(S_i^{(t-1)})) \text{ and } T_i^{(t)} = \rho(N_{r_t}(T_i^{(t-1)})).$$

For $t > s_0$ let

$$S_i^{(t)} = \rho^{-1}(N_{r_t}(S_i^{(t-1)})) \setminus (\bigcup_{i'=1}^{i-1} \bigcup_{u=1}^{\ln\ln n} S_{i'}^{(u)} \cup \bigcup_{i'=1}^{i-1} \bigcup_{u=1}^{\ln\ln n} \rho^{-1}(T_{i'}^{(u)}))$$

$$T_i^{(t)} = \rho(N_{r_t}(T_i^{(t-1)})) \setminus (\bigcup_{i'=1}^{i-1} \bigcup_{u=1}^{\ln\ln n} T_{i'}^{(u)} \cup \bigcup_{i'=1}^{i-1} \bigcup_{u=1}^{\ln\ln n} \rho(S_{i'}^{(u)})).$$

It is immediate that $|S_i^{(1)}| = |T_i^{(1)}| = r_1$. For $t \ge 2$ and (as will always be the case) $i < 4\ln n$, it follows from Lemmas 8 and 9 that **whp** $|S_i^{(t)}| \ge (K-3)|S_i^{(t-1)}|$ and $|T_i^{(t)}| \ge (K-3)|T_i^{(t-1)}|$ as long as both $S_i^{(t-1)}$ and $T_i^{(t-1)}$ are of size at most $n^{3/4}$. Lemma 8 means that for all $i' < i$,

$$\left| \bigcup_{t=1}^{s_0} S_i^{(t)} \cap \bigcup_{t=1}^{s_0} S_{i'}^{(t)} \right| \le 2. \tag{10}$$

(Otherwise, if the repeated points are a_1, a_2, a_3, then the paths between i, i' and a_1, a_2, a_3 form a bicyclic graph with at most $6s_0$ vertices, contradicting the lemma.)) Combining this with Lemma 9 means that for $t \le s_0$, $|S_i^{(t)}| \ge (K-3)|S_i^{(t-1)}|$. For generations $t > s_0$, for each i' the sets subtracted out are of size $O(K^{\ln\ln n})$, and so as long as $i < 4\ln n$, in all, the sets subtracted out are of size $O(K^{\ln\ln n}\ln n)$, much smaller than the size $\Omega((K-3)^{s_0})$ to which the set $S_i^{(t)}$ has by then grown. By throwing away random vertices if necessary, we can assume that $|S_i^{(t)}| = (K-3)|S_i^{(t-1)}|$ and $|T_i^{(t)}| = (K-3)|T_i^{(t-1)}|$. Thus if

$$\tau = \lceil 1 + \log_{K-3}(n^{3/4}/\lceil K\ln n\rceil)\rceil,$$

then **whp**

$$\forall i: \; n^{3/4} \le |S_i^{(\tau)}| = |T_i^{(\tau)}| \le Kn^{3/4}. \tag{11}$$

Each $x \in S_i^{(t)}$ defines a *walk* from a_i to x, of length $2t$, which is alternating w.r.t. the matching M_ρ; prune it to define a *path* $P[i,x]$. Similarly, each $y \in T_i^{(t)}$ defines a path $Q[i,y]$ from y to b_i, of length at most $2t$, which is alternating w.r.t. M_ρ.

We say that a cycle C of Π_1 is *small* if $|C| < n_0$; recall that we defined

$$n_0 = \left\lceil \frac{n\ln\ln n}{\ln n} \right\rceil.$$

Detailed analyses of random permutations have been undertaken by Arratia, Barbour, and Tavaré [3], in which the joint distribution of counts k_i of cycles of length i is approximated by independent Poissons $Z_i \sim \text{Pois}(1/i)$, and by Arratia and Tavaré [4], which provides a tighter bound on the distance between the true distribution and the Poisson approximation. From these (or more elementary analyses) we observe first that the expected number of vertices on small cycles is $n_0 - 1$ and so with probability $1 - O(n_0/n)$

$$\text{There are less than } 2n_0 \text{ vertices on small cycles.} \tag{12}$$

(The distance between the true distribution and independent Poisson estimate dominates the bound; the probability the Poissons exceed their expectation of n_0 by a factor of 2 is much smaller.) Assume from now on that π satisfies (12).

Let the small cycles of Π_1 be $C_1, C_2, \ldots$. At the start of Phase 2, from each small cycle C we choose an edge (a, b) of C. Let the chosen edges be $(a_i, b_i), i = 1, 2, \ldots, \lambda$. We now describe how we try to remove a C_i without creating any new small cycles. (See Figure 3.2.)

Suppose we have removed $C_1, C_2, \ldots, C_{i-1}$ and the original permutation π has become $\rho = \rho_i$. Assume that we have not already serendipitously removed C_i as well. Let (a_i, b_i) be the chosen edge of C_i.

Each alternating path $P[i, x]$ starts with a "forward" edge which is one of the $K \ln n$ shortest edges leaving a_i (the first branching factor was $r_1 = K \ln n$), has up to $\tau - 1$ other forward edges each of which is one of the K shortest edges leaving a vertex,[2] and has another up to τ "backward", matching edges (edges in M_ρ); a symmetric condition holds for Q_i.

It follows from the proof of Lemma 7 that **whp** each of these paths is such that the total length of its forward edges minus the total length of its backward edges is bounded by $c_4 \frac{\ln n}{n}$.

We now see that if we find $\xi_i \in S_i^{(\tau)}$ and $\eta_i \in T_i^{(\tau)}$ such that (ξ_i, η_i) is Red or Blue (recall the definition from the start of Section 3) then it — together with the edge (a_i, b_i) and the paths $P[i, \xi_i]$ and $Q[i, \eta_i]$ — defines an alternating cycle whose action on the current perfect matching increases the matching's cost by at most $(2c_4 + 2c_2)\frac{\ln n}{n}$. We now show that we can **whp** find at least one such alternating cycle whose action *does not create any new small cycles.* Furthermore, if such a path contains an edge of $C_{i'}$, $i' > i$, then this alternating cycle will also remove the small cycle $C_{i'}$.

Let ϕ be a random permutation of $[n]$ associating the vertices of X to those of Y, and let matrix $\hat{C}$ be defined by $\hat{C}(i, j) = C(i, \phi(j))$. If ψ is the (w.p. 1, unique) minimum solution to the assignment problem with matrix $\hat{C}$ then $\pi = \phi\psi$ is the minimum solution to the original problem. We exploit the randomness of ϕ, which produces a random permutation π from ψ. Instead of taking π as given, we assume that ψ is given and π is to be obtained through a random permutation ϕ. We condition on the cycle structure of π. Defining k_i as the number of cycles of length i in π, we assume that (i) $\sum_{i=1}^{n_0} ik_i \leq 2n_0$ and that (ii) $\sigma = \sum_{i=1}^{n} k_i \leq 2 \ln n$; these conditions hold **whp**.

How do we sample a random permutation conditioned upon having a cycle structure dictated by $k_1, k_2, \ldots$, i.e., dictated by the multiset $\{k_i \times i : \; i \in [n]\}$ in which cycle length i appears k_i times? Let Π denote the set of permutations of X with the given cycle structure. Let γ be any fixed permutation with the given cycle structure. (For example, if $t_1 = 0$,

[2] fewer than $\tau - 1$ if the path $P[i, x]$ resulted from nontrivially pruning a $(\tau - 1)$-long walk

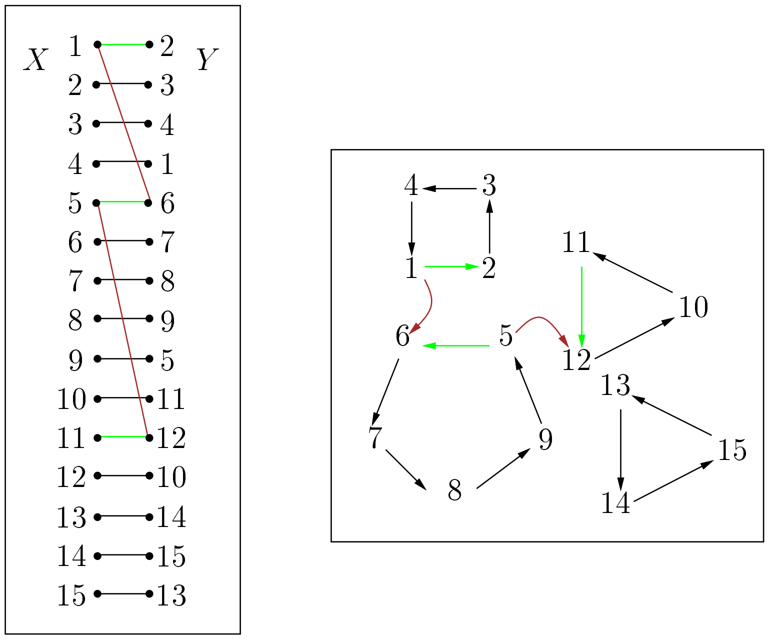

Figure 1: In the left box is a bipartite graph with matching edges shown as horizontals (black or grey). The right box shows the corresponding oriented cycle cover indicated by straight arrows (black or grey), for example the arrow $1 \longrightarrow 2$ indicating that X vertex 1 is matched to Y vertex 2. We imagine that only the pentagon is a "long" cycle, and all the others are short cycles needing repair.

Suppose that, to repair the cycle $1, 2, 3, 4$, we had selected edge $(1, 2)$. In the bipartite graph we find a path, rooted at 1, of cheap (Red) forward edges (shown as slanted grey lines in the left box) alternating with matching edges (horizontal solid lines), in this case the path $x_1, y_6, x_5, y_{12}, x_{11}$. The right box shows the NPD obtained from this alternating path, the light grey edges being removed from the cycle cover and the bent edges added to it. If in symmetry to the alternating path $1 \longrightarrow 11$ we found a cheap alternating path $y \longrightarrow 2$, and if the edge $(11, y)$ happened to be cheap (Red or Blue), we would repair the cycle $1, 2, 3, 4$. In this case we would also serendipitously repair the cycle $10, 11, 12$. The cycle $13, 14, 15$ (like most cycles, most of the time) is uninvolved.

$t_{\sigma+1} = n$, and the multi-sets $\{t_{j+1} - t_j : j \in [\sigma]\}$ and $\{k_i \times i : i \in [n]\}$ coincide, then we may define γ by: if $x, y \in C_j$ and $y = x + 1 \mod t_{j+1} - t_j$ then $\gamma(x) = y$.) Then given a bijection $f : X \to X$ we define a permutation π_f on X by $\pi_f = f^{-1}\gamma f$. Each permutation $\pi \in \Pi$ appears precisely $\prod_{i=1}^n k_i! i^{k_i}$ times as π_f. Thus choosing a random mapping f, chooses a random π_f from Π. (This is equivalent to randomly choosing $\phi = f^{-1}\gamma f \psi^{-1}$.)

The most natural way to look at this is to think of having oriented cycles on the plane whose vertices are at points $P_1, P_2, \ldots, P_n$ and then randomly labelling these points with X. Then if P' follows P on one of the cycles and P, P' are labelled x, x' by f then $\pi_f(x) = x'$.

To give a concrete example, Figure 3.2 included a "canonical" digraph 4-cycle labelled $1, 2, 3, 4$, arising from a corresponding canonically labelled structure in the bipartite graph, the matching edges $(x_1, y_2), (x_2, y_3), (x_3, y_4), (y_4, x_1)$. In a random labelling dictated by a random permutation f, these matching edges would be labelled $(x_{f(1)}, y_{f(2)})$, $(x_{f(2)}, y_{f(3)})$, $(x_{f(3)}, y_{f(4)})$, $(x_{f(4)}, y_{f(1)})$, and the digraph's 4-cycle would be labelled $f(1)$, $f(2)$, $f(3)$, $f(4)$.

As we proceed through Phase 2 we have to expose parts of f (equivalently ϕ). x is *clean* if $f(x)$ is unexposed (the label x has not yet been used) and *dirty* otherwise. Thus imagine that we have cycles, mostly unlabelled, but with a few vertices labelled. Let us use $\tilde{}$ to denote a partially labelled graph.

We can now describe how to eliminate the small cycles. We proceed in order through the selected edges $i \in [\lambda]$. At stage i we should have eliminated $C_1, C_2, \ldots, C_{j-1}$ for some j, and have a current perfect matching M_i, defining ρ_i. (Consider M_i to be fully revealed, but the labels on its vertices not revealed except for the selected edges in short cycles; thus all that is revealed of ρ_i is its cycle structure and labels on these few edges.)

We construct the trees S_i and T_i and then seek Red or Blue edges between the leaves $S_i^{(\tau)}$ of S_i and $T_i^{(\tau)}$ of T_i. We will consider only vertices of $S_i^{(t)}$ and $T_i^{(t)}$ that are clean and whose paths to their respective roots also contain only clean vertices; call these vertices *squeaky clean*. We take the squeaky clean vertices $v \in S_i^{(\tau)}$ in some fixed order. For each such vertex v we look, again in some fixed order, at each squeaky clean vertex $w \in T_i^{(\tau)}$. Each such edge (v, w) is either Red or has length uniform in $[c_2 \frac{\ln n}{n}, 1]$. Thus the probability that it is Red or Blue is at least $c_2 \frac{\ln n}{n}$. This lower bound holds conditionally on the current history — see comment at the beginning of Section 3.1. We assert that

$$\textbf{whp} \text{ there are } \geq n^{3/5} \text{ squeaky clean vertices in each of } S_i^{(\tau)} \text{ and } T_i^{(\tau)}; \qquad (13)$$

this will be shown in the last paragraph in this subsection. Thus if we run through $n^{2/5}$ squeaky clean vertices $v \in S_i^{(\tau)}$, the expected number of Red or Blue edges to squeaky clean vertices $w \in T_i^{(\tau)}$ is $\geq c_2 \frac{\ln n}{n} \cdot n^{2/5} \cdot n^{3/5}$, and with probability $\geq 1 - e^{-c_2 \ln n} = 1 - n^{-c_2}$, we find at least one Red or Blue edge. For a given v, if we find no Red or Blue edge to any w, we move on to the next v. If we find a Red or Blue edge (v, w), we test if for *acceptability* as described in the next paragraphs; if the edge is acceptable, the cycle can be repaired and we move on to the next cycle i. If the edge is not acceptable, v and w have been dirtied in the course of the testing, and we move on to the next v. Because we find a Red or Blue edge after exploring about $n^{2/5}$ vertices from $S_i^{(\tau)}$, and $|S_i^{(\tau)} \geq n^{3/5}$, we can find at least $n^{1/5}$ Red or Blue edges to test; we will soon see that the failure probability is $\ll n^{-1/5}$, so eventual success is assured **whp**.

Now consider squeaky clean $\xi_i \in S_i^{(\tau)}, \eta_i \in T_i^{(\tau)}$ such that (ξ_i, η_i) is Red or Blue. In the bipartite graph, there is a squeaky clean alternating cycle $C = P[i, \xi_i]$, (ξ_i, η_i), $Q[i, \eta_i]$, (b_i, a_i). (C may start life as a circuit rather than a cycle, in which case we prune it down to a cycle containing (b_i, a_i).) We will define what it means for a cycle to be "acceptable", and show that C is acceptable with probability at least $(\ln n)^{-\alpha}$, where

$$\alpha = 7(\ln K)^{-1} < 1/2.$$

For any $x \in S_i^{(t)}$, consider $P[i, x] = (x_0 = a_i, y_1, x_1, y_2, \ldots, y_t, x_t = x)$ where $y_j = \rho_i(x_j)$ for $j \geq 1$. $P[i, x]$ defines a sequence $M^{(0)}, M^{(1)}, \ldots, M^{(t)}$ of near-perfect matchings (see Figure 3.2) $M^{(s)} = (M^{(s-1)} \cup \{(x_{s-1}, y_s)\}) \setminus \{(x_s, y_s)\}$. Let $\Gamma^{(0)}, \Gamma^{(1)}, \ldots, \Gamma^{(t)}$ be the associated NPD's. We say that $\Gamma^{(s)}$ is *acceptable* if (i) $|\,\mathrm{PATH}(\Gamma^{(s)})| \geq n_0$ and (ii) the small cycles of $\Gamma^{(s)}$ are a subset of $\{C_{i+1}, \ldots, C_\lambda\}$. We say that x is acceptable if $\Gamma^{(0)}, \Gamma^{(1)}, \ldots, \Gamma^{(t)}$ are all acceptable.

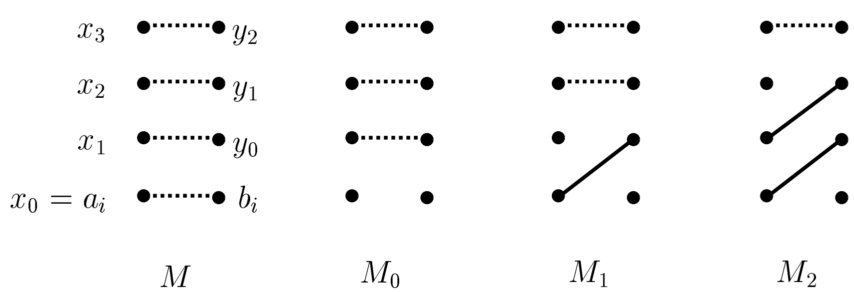

Figure 2: For $t = 3$, a perfect matching (dashed edges) and the sequence of near-perfect matchings (dashed and solid edges) defined by an alternating path $(x_0 = a_i, y_1, x_1, y_2, \ldots, x_t = x)$ (the union of all edges shown).

Going back to $P[i, x = x_t]$ let us estimate the probability that x_t is acceptable, given that it is clean and x_{t-1} is acceptable. Assume that we have revealed $f(x_{t-1})$ and that we have a partially labelled NPD $\tilde{\Gamma}_i^{(t-1)}$. We randomly choose $f(x_t)$ from the unlabelled points and label it with x_t. We then replace the arc $(f(x_t), \cdot)$ of $\tilde{\Gamma}_i^{(t-1)}$ by $(f(x_{t-1}), \cdot)$.[3] When $t = 1$, x_1 is acceptable unless $f(x_1)$ lies in a small cycle; it follows from (12) that given the previous exposures, this has probability at most $p_1 = 3\frac{\ln \ln n}{\ln n}$. For $t > 1$, x_t will be acceptable if $f(x_t)$ is not within n_0 of an endpoint of $\mathrm{PATH}(\tilde{\Gamma}_i^{(t-1)})$. We will see that

whp only $O((\ln n)^{2\alpha})$ squeaky clean cycles need to be checked

before an acceptable one is found.

Since each path has $O(\ln n)$ points, and we repeat for $O(\ln n)$ cycles, in all, at most $O((\ln n)^{2+2\alpha})$ values of f are exposed. So if x_t is clean, it will be unacceptable with

[3] When dealing with the path from b_i to η_i we randomly choose $f(x_t)$ and then replace the arc $(f(x_{t-1}), \cdot)$ by $(f(x_t), \cdot)$.

probability at most $p_2 = \frac{2n_0}{n-O((\ln n)^{2+2\alpha})} \leq \frac{3\ln\ln n}{\ln n}$ conditional on previous exposures. A similar analysis holds for the paths $Q[i,y]$.

If all vertices on C are clean then the probability that C is not acceptable is at most $p_1 + 1 - (1-p_2)^{2\tau} \leq \frac{3\ln\ln n}{\ln n} + 1 - \left(1 - \frac{3\ln\ln n}{\ln n}\right)^{2\tau} \leq 1 - (\ln n)^{-\alpha}$. Thus if we can find $(\ln n)^{2\alpha}$ clean cycles then one of them will succeed, with probability at least $1-(1-(\ln n)^{-\alpha})^{(\ln n)^{2\alpha}} \geq 1 - \exp\{-(\ln n)^{\alpha}\}$. As remarked earlier, we can in fact find far more clean cycles than this — in fact around $n^{1/5}$ of them — as long as there are around $n^{3/5}$ squeaky clean vertices in each of $S_i^{(\tau)}$ and $T_i^{(\tau)}$; this is all that remains to be shown.

Let $A_i^{(t)}$ denote the squeaky clean vertices of $S_i^{(t)}$, $t = 1, 2, \ldots, \tau$. It follows from Lemmas 8 and 9 that $|A_i^{(1)}| \geq K\ln n - 4\lambda - (\ln n)^{2\alpha}$ and that $|A_i^{(t)}| \geq (K-3)|A_i^{(t-1)}| - 4\lambda - (\ln n)^{2\alpha}$ for $2 \leq t \leq \ln\ln n$. Here we use (10) to argue that for $i' < i$, the first $\ln\ln n$ levels of each $S_{i'}, T_{i'}$ dirty at most 2 vertices of the first $\ln\ln n$ levels of S_i, giving the 4λ term. The $(\ln n)^{2\alpha}$ term comes from vertices dirtied during failed acceptability tests for the current cycle i, one dirtied vertex per level t per failed test. The higher levels of $S_{i'}, T_{i'}$ do not dirty any of the lower levels of S_i, by construction. In general, for $t > \ln\ln n$, Lemma 9 implies that $|A_i^{(t+1)}| \geq (K-3)|A_i^{(t)}| - 4\lambda\tau(\ln n)^{2\alpha}$. Thus, $|A_i^{(\tau)}| \geq n^{3/5}$. A similar argument holds for squeaky clean vertices of $T_i^{(\tau)}$, verifying the assertion (13).

3.3 Phase 3

For Phase 3 we use the Green edges. We can assert that **whp** at the end of Phase 2, all cycles are of length at least n_0 and so there are $o(\ln n)$ cycles. Given two cycles C_1, C_2 each of length at least n_0 then the probability that we cannot patch them together (delete edges (a_i, b_i) from $C_i, i = 1, 2$ and replace them by Red or Blue or Green edges $(a_1, b_2), (a_2, b_1)$) is at most $(1 - \frac{c_2^2(\ln n)^2}{n^2})^{n_0^2} \leq e^{-c_2^2(\ln\ln n)^2}$. Doing this $o(\ln n)$ times increases the cost by at most $o\left(\frac{(\ln n)^2}{n}\right)$ and so Phase 3 succeeds **whp**.

This completes the proof of the high-probability upper bound on ATSP − AP. We now consider the lower bound.

3.4 Proof of the lower bound

The Assignment Problem can be expressed as a *linear program*:

$$\text{Minimise} \sum_{i,j} C(i,j) z_{i,j} \text{ subject to } \sum_i z_{i,k} = \sum_j z_{k,j} = 1, \forall k, 0 \leq z_{i,j} \leq 1, \forall i,j. \qquad \text{(LP)}$$

This has the dual linear program:

$$\text{Maximise} \sum_i u_i + \sum_j v_j \text{ subject to } u_i + v_j \leq C(i,j), \forall i,j. \qquad \text{(DLP)}$$

Remark 10 Condition on an optimal basis for (LP). We may w.l.o.g. take $u_1 = 0$ in (DLP), whereupon w.p. 1 the other dual variables are uniquely determined. Furthermore, the reduced costs of the *non-basic* variables $\bar{C}(i,j) = C(i,j) - u_i - v_j$ are independently and uniformly distributed, with $\bar{C}(i,j) \in_{\text{unif}} [\max\{0, -u_i - v_j\}, 1 - u_i - v_j]$.[4]

[4]Do not be misled by the notation: $-u_i - v_j$ can be (and often is) positive.

Proof. The $2n-1$ dual variables are unique w.p. 1 because they satisfy $2n-1$ linear equations. The only conditions on the non-basic edge costs are that $C(i,j) \in [0,1]$ (equivalently $\bar{C}(i,j) \in [-u_i - v_j, 1 - u_i - v_j]$) and $\bar{C}(i,j) \geq 0$; intersecting these intervals yields the last claim. $\square$

Lemma 11 Whp

$$\max_{i,j} \{|u_i|, |v_j|\} \leq c_5 \frac{\ln n}{n}. \tag{14}$$

Proof. Optimal dual values u_i, v_j can be characterised as shortest distances, as follows [1]. Consider a directed bipartite digraph Γ on $X \cup Y$ with "forward" edges (x_i, y_j), $i, j \in [n]$, $j \neq \pi(i)$, of length $C(i,j)$; and "backward" edges (y_j, x_i), $i, j \in [n]$, $j = \pi(i)$, of length $-C(i, \pi(i))$. If $u_1 = 0$, then $-u_i$ is the shortest distance $d(x_1, x_i)$ from x_1 to x_i in Γ, and v_j is the shortest distance from x_1 to y_j.[5]

Lemma 7 implies that $-u_i, v_i \leq c_6 \frac{\ln n}{n}$ for $i \in [n]$. Furthermore, using the fact that a cheapest path is also a cheapest walk (derived from the optimal assignment, Γ has no negative-cost cycles), $-u_j = d(x_1, x_j) \leq d(x_1, x_i) + d(x_i, x_j) \leq -u_i + c_6 \frac{\ln n}{n}$ implies $u_i - u_j \leq c_6 \frac{\ln n}{n}$. Immediately, $|u_i| \leq c_6 \frac{\ln n}{n}$ and also, with $\bar{u} = \sum u_i / n$, $|\bar{u}| \leq c_6 \frac{\ln n}{n}$. Likewise, $v_i - v_j \leq c_6 \frac{\ln n}{n}$, from which $|v_i - \bar{v}| \leq c_6 \frac{\ln n}{n}$. But we know that **whp** the optimal assignment cost satisfies $1.51 < \sum_i u_i + \sum_j v_j < 1.94$ [14, 9, 17, 7], so $\bar{v} \in (1.51/n - \bar{u}, 1.94/n - \bar{u})$, giving $|\bar{v}| \leq c_6 \frac{\ln n}{n} + O(1/n)$ and finally $|v_i| \leq c_7 \frac{\ln n}{n}$. $\square$

Having solved LP we will have n basic variables $z_{i,j}$, $(i,j) \in I_1$, with value 1 and $n-1$ basic variables $z_{i,j}$, $(i,j) \in I_2$, with value 0. The edges (x_i, y_j), $(i,j) \in I = I_1 \cup I_2$ form a tree T^* in $K_{X,Y}$. We show that with probability at least $c_9 > 0$ there exists $(i,i) \in I_1$ (a loop) such that (x_i, y_i) is a pendant edge in T^*; w.l.o.g. suppose x_i is its leaf. In this case the optimal TSP tour, viewed as a bipartite matching, cannot use the edge (x_i, y_i) (a loop), and must use some other edge $(x_i, y_{i'})$; since x_i is a leaf in T^*, $z_{i,i'}$ is not a basic LP variable. The expected value of the reduced cost of $z_{i,i'}$ is at least $\frac{c_{10}}{n}$ and so $\mathbf{E}(\mathrm{ATSP} - \mathrm{AP}) \geq \frac{c_9 c_{10}}{n}$ and the lower bound follows.

To prove the existence of (i,i) we show that **whp** the optimal assignment ψ for $\hat{C}$ of Section 3 has at least $c_{11} n$ leaves L. After applying the random permutation ϕ, the number of leaves giving rise to loops is, at least, a random variable whose distribution is asymptotically Poisson with density c_{11}; thus

$$\mathbf{Pr}(\exists \text{ at least one leaf-loop}) \geq (1 - o(1))(1 - e^{-c_{11}}).$$

By taking a spanning tree T of $K_{X,Y}$ which contains a perfect matching M and shrinking the edges of M we obtain a tree isomorphic to a spanning tree T' of K_n. Each T arises from exactly 2^{n-1} T's because we have two choices as to how to configure each non-M edge. (An (i,j) edge in T' can in T be expanded to (x_i, y_j) or to (x_j, y_i).) Let $b(T) = b(T')$ denote the number of branching nodes (degree ≥ 3) of T and T'. A tree T' is ϵ-bushy if $b(T') \leq \epsilon n$.

[5] It is easy to see this from the graph with edge costs $\bar{C}(i,j) = C(i,j) - u_i - v_j \geq 0$. This graph includes a spanning tree of 0-cost edges, so all distances are 0. The C-cost of any path is almost the same as its $\bar{C}$-cost: of the two directed edges leading into and out of any intermediate node, one has a u_i (or v_j) added, and the other has the same quantity subtracted. The cancellation fails only at the path's source (but we defined $u_1 = 0$) and at its terminal, resulting in C-distance $-u_i$ or $+v_j$ as claimed.

Bohman and Frieze used this concept in [6] and showed that the number of ϵ-bushy trees is at most $n!e^{\theta(\epsilon)n}$ where $\theta(\epsilon) \to 0$ as $\epsilon \to 0$. It follows that the number of ϵ-bushy trees of $K_{X,Y}$ which have a perfect matching is at most $e^{\theta(\epsilon)n}2^{n-1}n!$. Observe that the number of leaves in T is at least $b(T)$. We complete the proof by showing that, for a sufficiently small constant ϵ,

$$\mathbf{Pr}(T^* \text{ is } \epsilon\text{-bushy}) = o(1). \tag{15}$$

For any tree T with a perfect matching, we can put $u_1 = 0$ and then solve the equations $u_i + v_j = C(i,j)$ for $(x_i, y_j) \in T$ to obtain the associated dual variables. T is optimal if $\bar{C}(i,j) = C(i,j) - u_i - v_j \geq 0$ for all $(x_i, y_j) \notin T$. Let $Z_T = \sum_i u_i + \sum_j v_j$. Let Now **whp** the optimal tree T^* satisfies $Z_{T^*} \in [1.51, 1.94]$, because Z_{T^*} is the optimal assignment cost, and it is known both that expectation is in the stated range [17, 7] and that the actual value is concentrated about the expectation [2]. Then if $\mathcal{E}$ denotes the event $\{(14) \text{ and } Z_T \in [1.51, 1.94]\}$, for any tree T, over random matrices $C(i,j)$,

$$\begin{aligned}
&\mathbf{Pr}(Z_T \in [1.51, 1.94] \text{ and } (14) \text{ and } \bar{C}(i,j) \geq 0\ \forall (i,j) \notin I) \\
&\quad \leq \mathbf{Pr}(\bar{C}(i,j) \geq 0\ \forall (i,j) \notin T \mid \mathcal{E}) \times \mathbf{Pr}(Z_T \in [1.51, 1.94]) \\
&\quad \leq \frac{1.94^n}{n!} \mathbf{E}\Big(\prod_{(x_i,y_j) \notin T} (1 - (u_i + v_j)^+) \mid \mathcal{E}\Big) \\
&\quad \leq \mathbf{E}(\exp\{- \sum_{(x_i,y_j) \notin T} (u_i + v_j)\} \mid \mathcal{E}) \frac{1.94^n}{n!} \\
&\quad \leq \mathbf{E}(e^{-nZ_T} \exp\{ \sum_{(x_i,y_j) \in T} (u_i + v_j)\} \mid \mathcal{E})) \frac{1.94^n}{n!} \\
&\quad \leq e^{-1.51n} n^{2c_5} \frac{1.94^n}{n!}.
\end{aligned}$$

Explanation $\frac{1.94^n}{n!}$ *bounds the probability that the sum of the lengths of the edges in the perfect matching of T is at most 1.94. The product term is the probability that each non-basic reduced cost is non-negative.*

Thus

$$\begin{aligned}
&\mathbf{Pr}(\exists \text{ an } \epsilon\text{-bushy tree } T : Z_T \in [1.51, 1.94] \text{ and } (14) \text{ and } \bar{C}(i,j) \geq 0\ \forall (i,j) \notin I) \\
&\quad \leq n!2^n e^{\theta(\epsilon)n} \times e^{-1.51n} n^{2c_5} \frac{1.94^n}{n!} \\
&\quad = o(1)
\end{aligned}$$

for ϵ sufficiently small. This implies (15).

4 An enumerative algorithm

We can now prove Theorem 4, restated here for convenience.

Theorem 4 **Whp**, *a random instance of the* ATSP *can be solved exactly in time* $e^{\tilde{O}(\sqrt{n})}$.

Proof. Let I_k denote the interval $[2^{-k}c_1\frac{(\ln n)^2}{n}, 2^{-(k-1)}c_1\frac{(\ln n)^2}{n}]$ for $k \geq 1$. It follows from Lemmas 11 and 10 that **whp** (i) there are $\leq c_1 2^{-(k-1)} n \ln n$ non-basic variables $z_{i,j}$ whose reduced cost is in $I_k, 1 \leq k \leq k_0 = \frac{1}{2}\log_2 n$ and (ii) there are $\leq 2c_1\sqrt{n}\ln n$ non-basic variables $z_{i,j}$ whose reduced cost is $\leq c_1\frac{(\ln n)^2}{n^{3/2}}$.

We can search for an optimal solution to ATSP by choosing a set of non-basic variables, setting them to 1 and then re-solving the assignment problem. If we try all sets and choose the best tour we find, then we will clearly solve the problem exactly. However, it follows from Theorem 1 that we need only consider sets which contain $\leq 2^k$ variables with reduced costs in I_k and none with reduced cost $\geq c_1\frac{(\ln n)^2}{n}$. Thus **whp** we need only check at most

$$2^{2c_1\sqrt{n}\ln n}\prod_{k=1}^{k_0}\sum_{t=1}^{2^k}\binom{c_1 2^{-(k-1)}n\ln n}{t} = e^{\tilde{O}(\sqrt{n})}$$

sets. □

5 Second best assignment

We recall and prove Theorem 3, on the gap Δ_1 between the costs of the cheapest and second-cheapest assignments.

Theorem 3 *Over random cost matrices C,*

$$\frac{1}{n^2}(1-o(1)) \leq \mathbf{E}(\Delta_1) \leq c_3\frac{\ln n}{n^2}.$$

Proof. Δ_1 is equal to the minimum non-basic reduced cost.[6] From Lemma 11 and $\sum_i u_i + \sum_j v_j > 1.51$ **whp**, it follows that **whp** there are at least $n_1 = c_7\frac{n^2}{\ln n}$ pairs i,j such that $u_i + v_j > 0$. Each such pair corresponds to a non-basic variable $C(i,j)$, and it follows from Remark 10 that the minimum reduced cost among this set is at most $\frac{1}{n_1+1}$ in expectation, proving the upper bound.

For the lower bound, again from Remark 10, the $n^2 - 2n + 1$ non-basic reduced costs $\bar{C}(i,j)$ are independent, with $\bar{C}(i,j) \in_{\text{unif}} [a_{i,j}, b_{i,j}]$ where each $a_{i,j} \geq 0$ and (from Lemma 11) each $b_{i,j} \geq 1 - 2c_5\ln n/n$. The minimum of this collection satisfies $\mathbf{E}(\min\{\bar{C}(i,j)\}) \geq \frac{1}{n^2-2n}(1 - 2c_5\ln n/n) = \frac{1}{n^2}(1-o(1))$. □

References

[1] R.K. Ahuja, T.L. Magnanti and J.B. Orlin, *Network Flows Theory, Algorithms and Applications*, Prentice-Hall, 1993.

[6] From linear programming, Δ_1 is at least the minimum non-basic reduced cost. Also, Δ_1 is no more than this: For the assignment problem, edges corresponding to basic variables form a tree. Adding any non-basic edge creates an alternating cycle, whose symmetric difference with the optimal matching gives a second matching. The cost increase is the cost of the cycle, which is the sum of its (signed) edge costs. The sum of the signed costs of the edges around a cycle is equal to the same sum of the reduced costs, because each u_i, for example, is added twice, with opposite signs, to the two edges incident on x_i. The cycle in question contains only a single non-basic edge, so the sum of its reduced edge costs is just the cost of this edge.

[2] D. Aldous, *Asymptotics in the random assignment problem*, Probability Theory and Related Fields 93 (1992) 507–534.

[3] R. Arratia, A.D. Barbour and S. Tavaré, *Poisson process approximations for the Ewens sampling formula*, Ann. Appl. Probab. **2** (1992), no. 3, 519–535.

[4] R. Arratia and S. Tavaré, *The cycle structure of random permutations*, Ann. Probab. **20** (1992), no. 3, 1567–1591.

[5] E. Balas and P. Toth, *Branch and Bound Methods*, in The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization (E.L. Lawler, J.K. Lenstra, A.H. Rinnooy Kan and D.B. Shmoys (editors)), Wiley, 1986, 361–400.

[6] T. Bohman and A.M. Frieze, *Avoiding a giant component*, to appear.

[7] D. Coppersmith and G.B. Sorkin, *Constructive bounds and exact expectations for the random assignment problem*, Random Structures and Algorithms 15 (1999), 133–144.

[8] M.E. Dyer and A.M. Frieze, *On patching algorithms for random asymmetric traveling salesman problems*, Mathematical Programming 46 (1990) 361–378.

[9] M.E. Dyer, A.M. Frieze and C. McDiarmid, *Linear programs with random costs*, Mathematical Programming 35 (1986) 3–16.

[10] P. Erdős and A. Rényi, *On random matrices*, Publ. Math. Inst. Hungar. Acad. Sci. 8 (1964) 455–461.

[11] A.M. Frieze and G.R. Grimmett, *The shortest path problem for graphs with random arc-lengths*, Discrete Applied Mathematics 10 (1985) 57–77.

[12] A.M. Frieze, R.M. Karp and B. Reed, *When is the assignment bound asymptotically tight for the asymmetric traveling-salesman problem?*, SIAM Journal on Computing 24 (1995) 484–493.

[13] M. Held and R.M. Karp, *A dynamic programming approach to sequencing problems*, SIAM Journal of Applied Mathematics 10 (1962) 196–210.

[14] R.M. Karp, *An upper bound on the expected cost of an optimal assignment,* Discrete Algorithms and Complexity: Proceedings of the Japan-US Joint Seminar (D. Johnson et al., eds.) 1–4, Academic Press, New York, 1987.

[15] R.M. Karp, *A patching algorithm for the non-symmetric traveling salesman problem*, SIAM Journal on Computing 8 (1979) 561–573.

[16] R.M. Karp and J.M. Steele, *Probabilistic analysis of heuristics*, in The traveling salesman problem: a guided tour of combinatorial optimization, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys Eds. (1985) 181–206.

[17] B. Olin, *Asymptotic properties of random assignment problems*, Ph.D. thesis, Kungl Tekniska Högskolan, Stockholm, Sweden, 1992.