A CONDITIONAL HARDNESS OF THE W_∞-MATCHING PROBLEM

In this section, we prove a conditional hardness result for approximating the W_∞ distance.

Suppose we are given the bipartite graph $G = (V, E)$, where $V = V_1 \cup V_2$ with $|V_1|, |V_2| = n$, $V_1 \cap V_2 = \varnothing$, and $E \subseteq V_1 \times V_2$. We will reduce this problem of determining if G contains a perfect matching to computing an approximate W_∞-distance.

Construct the following metric ρ on V. For each distinct $v_1, v_2 \in V$, if $(v_1, v_2) \in E$ or $(v_2, v_1) \in E$ then define $\rho(v_1, v_2) = 1$. Otherwise, define $\rho(v_1, v_2) = 2$. For completeness, one can define $\rho(v, v) = 0$ for all $v \in V$. The finite metric space (V, ρ) can be constructed in $O(n^2)$ time, given G. It is easy to see that (V, ρ) is a metric space: (i) by definition, $\rho(v, v) = 0$ and $\rho(v, v') > 0$ for all $v' \neq v$; (ii) if $\rho(v_1, v_2) = 1$ for some $v_1 \neq v_2$, then either (v_1, v_2) or (v_2, v_1) is in E, in which case $\rho(v_2, v_1) = 1$; (iii) $\rho(v_1, v_2) \leq 2$ and $\rho(v_1, v_3) + \rho(v_3, v_2) \geq 1 + 1 = 2$ for all $v_1, v_2 \in V$ and $v_3 \neq v_1, v_2$, implying triangle inequality in combination with observation (i) to prove the degenerate case $v_3 = v_1$ or $v_3 = v_2$.

Additionally define the distributions $\mu_1 \colon V_1 \to [0, 1]$ and $\mu_2 \colon V_2 \to [0, 1]$ by $\mu_1(v_1) = \mu_2(v_2) = \frac{1}{n}$ for all $v_1 \in V_1$ and all $v_2 \in V_2$. Assume we are given an approximation algorithm $\mathcal{A}$ for the W_∞ distance, and let $\sigma_{\mathcal{A}}$ denote the transport plan from μ_1 to μ_2 computed by algorithm $\mathcal{A}$.

Since $\rho(v_1, v_2) \in \{1, 2\}$ for all $v_1, v_2 \in V_1 \times V_2$, it must also be the case that

$$w_\infty(\sigma) := \max_{x,y:\sigma(x,y)>0} \rho(x, y) \in \{1, 2\}$$

for any transport plan σ and therefore $W_\infty(\mu_1, \mu_2) \in \{1, 2\}$. We use this observation to prove the following crucial relationship between $W_\infty(\mu_1, \mu_2)$ and the maximum cardinality matching in G.

Lemma A.1. *A perfect matching exists in G if and only if* $W_\infty(\mu_1, \mu_2) = 1$.

Proof. If $W_\infty(\mu_1, \mu_2) = \min_\sigma w_p(\sigma) = 2$, then it is impossible to construct a transport plan σ where every edge has distance 1 and therefore at least one edge (u, v) where $\sigma(u, v) > 0$ satisfies $\rho(u, v) = 2$. By definition, the metric ρ is equal to 2 if and only if the pair is not an edge in G. Therefore, any matching must have size at most $n - 1$.

If $W_\infty(\mu_1, \mu_2) = 1$, then it is possible to construct a transport plan σ where every edge has a distance 1, and by standard network flow theory this transport plan σ is a convex combination of matchings in $(V_1 \cup V_2, V_1 \times V_2)$. By definition of ρ, we note $\rho(v_1, v_2) = 1$ if and only if $(v_1, v_2) \in E$. We conclude that any matching M where $\sigma(v_1, v_2) > 0$ for every $(v_1, v_2) \in M$ is also a perfect matching in G. □

Therefore, it suffices to determine if either $W_\infty(\mu_1, \mu_2) = 1$ or $W_\infty(\mu_1, \mu_2) = 2$, and extract any matching from the resulting transport plan in $O(n^2)$ time in the former case.

We briefly describe this (standard) procedure of how to construct a perfect matching M in $V_1 \times V_2$ using $\sigma_{\mathcal{A}}$ in $O(n^2)$ time such that for every $(v_1, v_2) \in M$, we have $\sigma_{\mathcal{A}}(v_1, v_2) > 0$. To do this, we iteratively choose an arbitrary edge $(u, v) \in V_1 \times V_2$ such that $\sigma(u, v) > 0$, add (u, v) to M, remove u from V_1 and v from V_2. Repeat until M is a perfect matching. Since σ is known to be a convex combination of perfect matchings, we conclude that the algorithm always constructs a perfect matching.

What follows is an observation that even when given relatively weak approximation algorithms for the ∞-Wasserstein distance, it is possible to distinguish between these two cases. We first prove this for the case that $\mathcal{A}$ is a relative approximation algorithm.

Lemma A.2. *Suppose for some* $\varepsilon > 0$ *there exists an algorithm running in* $O(n^2)$ *time which computes a* $(2 - \varepsilon)$*-approximate transport plan under the* W_∞ *metric between two discrete probability distributions with supports of size at most* n*. Then, for any bipartite graph G, one can compute a perfect matching in G or conclude that none exists in* $O(n^2)$ *time.*

Proof. We assume the existence of an algorithm $\mathcal{A}$ which, when given an input finite metric space (V, ρ) and distributions μ, ν, computes a transport plan σ such that $w_\infty(\sigma) \leq (2 - \varepsilon) \cdot W_\infty(\mu, \nu)$ in $O(n^2)$ time.

First suppose that $w_\infty(\sigma_{\mathcal{A}}) = 2$. By the approximation guarantee of the algorithm $\mathcal{A}$, we conclude that $W_\infty(\mu_1, \mu_2) \geq \frac{2}{2-\varepsilon} > 1$ and therefore $W_\infty(\mu_1, \mu_2) = 2$. Now suppose that $w_\infty(\sigma_{\mathcal{A}}) = 1$. Then we have immediately found a minimizing transport plan and conclude that $W_\infty(\mu_1, \mu_2) = 1$. We conclude that if $\mathcal{A}$ is a $(2-\varepsilon)$-approximation algorithm then necessarily $w_\infty(\sigma_{\mathcal{A}}) = W_\infty(\mu_1, \mu_2)$ for the constructed input metric space (V, ρ) and distributions μ_1, μ_2. The result follows after a simple application of Lemma A.1. $\square$

We emphasize that the metric space used to prove Lemma A.2 has spread $\Delta = 2$. Therefore, Lemma A.2 is useful even if the algorithm $\mathcal{A}$ has $O(n^2 f(\Delta))$ runtime for any function $f\colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ independent of n. Next, we prove the analogous claim for the case where $\mathcal{A}$ is an additive approximation algorithm.

Lemma A.3. *Suppose for some $\varepsilon > 0$ there exists an algorithm running in $O(n^2)$ time which computes a $(\frac{\Delta}{2} - \varepsilon)$-additive approximate transport plan under the W_∞ metric between two discrete probability distributions with supports of size at most n in a metric space with diameter Δ. Then, for any bipartite graph G, one can compute a perfect matching in G or conclude that none exists in $O(n^2)$ time.*

Proof. We assume the existence of an algorithm $\mathcal{A}$ which, when given an input finite metric space (V, ρ) and distributions μ, ν, computes a transport plan σ such that $w_\infty(\sigma) \leq W_\infty(\mu, \nu) + \left(\frac{\Delta}{2} - \varepsilon\right)$ in $O(n^2)$ time.

First suppose that $w_\infty(\sigma_{\mathcal{A}}) = 2$. By the approximation guarantee of the algorithm $\mathcal{A}$, we conclude that $W_\infty(\mu_1, \mu_2) \geq w_\infty(\sigma_{\mathcal{A}}) - \left(\frac{\Delta}{2} - \varepsilon\right) = 2 - (1-\varepsilon) > 1$ and therefore $W_\infty(\mu_1, \mu_2) = 2$. Now suppose that $w_\infty(\sigma_{\mathcal{A}}) = 1$. Then we have immediately found a minimizing transport plan and conclude that $W_\infty(\mu_1, \mu_2) = 1$. We conclude that if $\mathcal{A}$ is a $\left(\frac{\Delta}{2} - \varepsilon\right)$-approximation algorithm then necessarily $w_\infty(\sigma_{\mathcal{A}}) = W_\infty(\mu_1, \mu_2)$ for the constructed input metric space (V, ρ) and distributions μ_1, μ_2. The result follows after a simple application of Lemma A.1. $\square$

Then a simple combination of Lemmas A.2 and A.3 implies Theorem 1.3.

B k-Level Clustering

In Section 2, we used a two-layered clustering to approximate d up to multiplicative factor 4 using $O(n^{3/2})$ space. This approach can be generalized to a k-level clustering. Extending to k-level clustering has the benefit of a reduced number of clusters in which any point in P_0 is expected to participate, at the expense of an increase in the stretch factor of the data structure.

In this section, we describe a clustering based distance function that can be constructed in $O(n^2)$ time, approximates d up to a factor of $(4+\varepsilon)(k-1)$ and that can be represented using $O(kn^{1+1/k}\varepsilon^{-1}\log\Delta)$ space. Similar to Section 2.1, we can use this clustering to construct a spanner and maintain (weighted) bichromatic closest pairs. Then once we have those data structures, the algorithms in Section 3 work in the same manner. Applying the combinatorial algorithm in Section 3.2 to the k-level clustering for $k = 3$ gives the $(8+\varepsilon)$-approximation result in Theorem 1.2.

k-layered clustering. We now construct a k-level clustering of points of P. Let $P_0 = P$. For $i = 1, \ldots, k-1$, we next choose a subset $P_i \subseteq P_{i-1}$ by sampling each point in P_{i-1} independently with probability $\theta = n^{-1/k}$. The expected size of P_i is $\mathbb{E}[|P_i|] = n\theta^i = n^{1-i/k}$ for each $i \leq k-1$.

Set $t = \lceil \log_{(1+\frac{\varepsilon}{4})} \Delta \rceil$. Let $r_0 = 0$ and $r_i = (1+\frac{\varepsilon}{4})^i$ for $1 \leq i \leq t$. For each $0 \leq l < k-1$, $q \in P_l \setminus P_{l+1}$ and $1 \leq i \leq t$, define the cluster $C_q[i]$ as

$$C_q[i] = \{x \in V(q, P_{l+1}) \mid \mathsf{d}(x, q) \leq r_i\}.$$

Finally for each $q \in P_{k-1}$ and $1 \leq i \leq t$, define the cluster $C_q[i]$ as

$$C_q[i] = \{x \in P_0 \mid \mathsf{d}(x, q) \leq r_i\}.$$

Let $\mathcal{C} = \{C_q[i] \mid q \in P, i \leq t\}$ be the collection of all clusters. Again define the *degree* of a point $p \in P$ as $\deg_{\mathcal{C}}(p) := |\{C \in \mathcal{C} \mid p \in C\}|$. In an analogous manner to Lemma 2.1, we prove that the expected degree of each point in P is small.

Lemma B.1. $\mathbb{E}[\deg_{\mathcal{C}}(p)] = O(kn^{1/k}\varepsilon^{-1}\log\Delta)$ *for all* $p \in P$.

Proof. Let $x \in P_0$ and fix $i < k-1$. Let $w_1, \ldots, w_l$ be the elements of P_i in order of non-decreasing distances to x. If $x \in C_{w_j}$ then $\mathsf{d}(x, w_j) < \mathsf{d}(x, P_{i+1})$. So it must be the case that $w_1, \ldots, w_{j-1} \notin P_{i+1}$. Note that $\theta = \Pr[x \in P_{i+1} \mid x \in P_i]$. Then,

$$\Pr\left[x \in C_{w_j}\right] \leq \prod_{t<j} \Pr\left[w_t \notin P_{i+1} \mid w_t \in P_i\right] = (1-\theta)^{(j-1)}.$$

The expected number of clusters that contain x at level $i < k-1$ is

$$\sum_{t \leq l} \Pr\left[x \in C_{w_t}\right] \leq \sum_{t \leq l} (1-\theta)^t \leq \theta^{-1} = n^{1/k}.$$

For level $i = k-1$, the point x is contained in every cluster C_w for $w \in P_{k-1}$. The expected number of clusters at level $i = k-1$ is

$$E[|P_{k-1}|] = n\Pr\left[x \in P_{k-1}\right] = n\theta^{k-1} = n^{1/k}$$

Therefore the total expected number of clusters containing x is $kn^{1/k}$. □

Cluster-induced distance approximation. As in Section 2, for any cluster $C = C_q[i] \in \mathcal{C}$ and any pair of points $x, y \in C_q[i]$, define the *cluster-induced distance* between x and y as

$$\mathsf{d}_C(x, y) = 2r_i.$$

Then define $\mathcal{C}(x, y) = \{C \in \mathcal{C} \mid x, y \in C\}$ and $\mathsf{d}_{\mathcal{C}}(x, y) = \min_{C \in \mathcal{C}(x,y)} \mathsf{d}_C(x, y)$. We prove that this minimum cluster-induced distance approximates $\mathsf{d}(x, y)$ within a factor or $(4+\varepsilon)(k-1)$.

Lemma B.2. $\mathsf{d}(x, y) \leq \mathsf{d}_{\mathcal{C}}(x, y) \leq (4+\varepsilon)(k-1) \cdot \mathsf{d}(x, y)$

Proof. First we show that if x and y are separated by P_l, i.e. if there exist $\alpha, \beta \in P_l$ such that $\mathsf{d}(x, \alpha) < \mathsf{d}(x, y)$ and $\mathsf{d}(y, \beta) < \mathsf{d}(x, y)$, then $\mathsf{d}(y, P_l) \geq \mathsf{d}(y, P_{l+1}) - 2\mathsf{d}(x, y)$. Let α and β be the closest points to x and y in P_{l+1}, respectively. Let a and b be the closest points to x and y in P_l respectively. Then

$$\begin{aligned}
\mathsf{d}(x, a) &\geq \mathsf{d}(y, a) - \mathsf{d}(x, y) && \text{[triangle inequality]} \\
&\geq \mathsf{d}(y, \beta) - \mathsf{d}(x, y) && \text{[definition of } \beta\text{]} \\
&\geq \mathsf{d}(x, \beta) - 2\mathsf{d}(x, y) && \text{[triangle inequality]} \\
&\geq \mathsf{d}(x, \alpha) - 2\mathsf{d}(x, y) && \text{[definition of } \alpha\text{]}.
\end{aligned}$$

Let x be inserted in round l. Then $y \notin C_x$ if and only if $\mathsf{d}(y, P_{l+1}) \leq \mathsf{d}(x, y)$. Let C_z be the cluster containing both x and y that minimizes $\mathsf{d}(y, z)$. Assume that C_z was inserted in round j. Note that $j < l$ and $l - j \leq k-2$. Then by the claim above,

$$\begin{aligned}
\mathsf{d}(y, z) &\leq \mathsf{d}(y, P_{l+1}) + 2(l-j)\mathsf{d}(x, y) \\
&\leq (2(k-2)+1)\mathsf{d}(x, y).
\end{aligned}$$

Then by the triangle inequality it follows that $\mathsf{d}(x, z) \leq 2(k-1)\mathsf{d}(x, y)$. Note that $\mathsf{d}(x, z) \leq r_i < (1+\frac{\varepsilon}{4})\mathsf{d}(x, z)$, where C_z has minimum index i. Therefore $\mathsf{d}_{\mathcal{C}}(x, y) = 2r_i \leq 2(1+\frac{\varepsilon}{4})\mathsf{d}(x, z) \leq 4(k-1)(1+\frac{\varepsilon}{4})\mathsf{d}(x, y)$. □

C Missing Details from the Main Text

We omitted many of the proofs of technical lemmas from the main text for the sake of space. In this section, we conclude with any missing details.

C.1 Details from Section 2.1

We first prove that the shortest path distance $\mathsf{d}_{G,p}$ in the directed graph G with weights w_p approximates $\mathsf{d}(\cdot,\cdot)^p$.

Proof of Lemma 2.3. Let $p \in [1,\infty)$ be an arbitrarily chosen value, and let $a \in A, b \in B$ be arbitrary points. We note by construction of G, for any path π from a to b in the graph G, there exists a unique edge $e = (a_C \to b_C) \in E$ for some cluster $C \in \mathcal{C}$ such that $w(e) > 0$.

To prove the upper bound, we show that there exists a path where the edge $e = (a_C \to b_C)$ satisfies $w_p(e) \le (4+\varepsilon)^p \cdot \mathsf{d}^p(a,b)$. By Lemma 2.2, observe that there exists a cluster $C \in \mathcal{C}$ with index i where $a,b \in C$ and $2r_i \le (4+\varepsilon)\cdot \mathsf{d}(a,b)$. Since $a \in C$, we observe that the edge $a \to a_C$ exists in G. Similarly, since $b \in C$, the edge $b_C \to b$ exists in G. Therefore, the path $\pi = a \to a_C \to b_C \to b$ is a path from a to b in G. The weight of the edge $a_C \to b_C$ is $w_p(a_C \to b_C) = (2r_i)^p \le ((4+\varepsilon)\cdot \mathsf{d}(a,b))^p = (4+\varepsilon)^p \cdot d^p(a,b)$.

To prove the lower bound, we show that for any such path π from a to b in G, $w_p(e) \ge d^p(a,b)$. Suppose $\pi = a \to a_C \to b_C \to b$ is a path from a to b in G. By construction, observe that if the path $\pi = a \to a_C \to b_C \to b$ is a path in G for some cluster $C \in \mathcal{C}$ with index i, then both a and b are contained in C. Since a and b are contained in C, there exists a point $q \in C$ (in particular, choose the center of the cluster $C_q[i]$) such that $\mathsf{d}(a,q) \le r_i$ and $\mathsf{d}(b,q) \le r_i$. Conclude that $\mathsf{d}(a,b) \le \mathsf{d}(a,q) + \mathsf{d}(q,b) \le 2r_i$ by triangle inequality and therefore $\mathsf{d}^p(a,b) \le (2r_i)^p = w_p(a_C \to b_C)$. ☐

We next prove that the global heap H constructed for the BCP data structure will contain the weighted bichromatic closest pair at its root.

Proof of Lemma 2.4. Let (a^*, b^*) be the pair stored at the root of H, with key $\phi_{C^*} = \mathsf{d}_w(a^*, b^*)$. Then $\phi_{C^*} = \min_{C\in\mathcal{C}} \phi_C$. Then we have the following,

$$\begin{aligned}\min_{C\in\mathcal{C}} \phi_C &= \min_{C\in\mathcal{C}}\{2r_i - \max_{(a,b)\in C}\{w(a) - w(b)\}\}\\ &= \min_{(a,b)\in A\times B}\{\mathsf{d}_C(a,b) - w(a) + w(b)\}\\ &= \min_{(a,b)\in A\times B} \mathsf{d}_w(a,b).\end{aligned}$$

This completes the proof. ☐

Finally, we prove that updates to the dynamic BCP data structure can be done in $O(\sqrt{n}\varepsilon^{-1}\log n \log\Delta)$ time.

Proof of Lemma 2.5. Without loss of generality assume $p \in A$. There are two steps to insert (or delete) a point p into (from) the BCP$_w$ data structure. For each cluster $C \in \mathcal{C}$ that contains p,

1. First, we add (delete) p to (from) the max-heap of C, and update the edge (a_C, b_C) for the cluster.
2. Next, for each updated cluster, update the key $\phi_C = 2r_i - w(a_C) + w(b_C)$ in the min-heap H.

By Lemma 2.1, the expected degree of p is $O(n^{1/2}\varepsilon^{-1}\log\Delta)$. Both steps take $O(\log n)$ time. So each update takes an expected $O(n^{1/2}\varepsilon^{-1}\log\Delta\log n)$ time. ☐

We briefly note that the details of correctness and time complexity for the weighted nearest neighbor data structure in Section 2.1 were also omitted. The analysis of the described data structure is nearly identical to that of the dynamic BCP data structure: nearest neighbors are stored at the root of the heap, where the weight is most extreme, and insertion or deletion will modify every cluster containing the point.

C.2 Details from Section 3.1

We omitted a formal cost analysis for the transport plan computed by the minimum-cost flow based algorithm from the main text since intuitively the fact that shortest paths in G approximate d^p should imply a minimum cost flow will also approximate W_p. In this section, we give a formal cost analysis of the algorithm in Section 3.1 for completeness. We first prove the case when p is finite.

Proof of Theorem 1.1 for $p < \infty$. By standard analysis of network flow algorithms, since the capacity of each edge $s \to x$ is $\mu(x)$ and the capacity of each edge $y \to t$ is $\nu(y)$, we conclude that the resulting σ is a transport plan. Moreover, given f^*, the transport plan σ can be constructed in $O(|E|)$ time since every $s - t$ path in G has constant length and at every iteration the flow along at least one edge is decremented to zero. We conclude with a cost analysis of σ.

First, we prove the lower bound. By construction of the graph G, we note that any path from s to t must be of the form $s \to a \rightsquigarrow b \to t$ where $a \rightsquigarrow b$ is a path in G from $a \in A$ to $b \in B$. By Lemma 2.3, conclude that $\sum_{e \in a \rightsquigarrow b} w_p(e) \geq \mathsf{d}_{G,p}(a,b) \geq \mathsf{d}^p(a,b)$ for any $a \in A, b \in B$ and any path $a \rightsquigarrow b$ in G. Moreover, every edge of the form $s \to a$ and $b \to t$ has cost zero. This implies

$$W_p(\mu,\nu) \leq w_p(\sigma) = \left(\sum_{a,b \in A \times B} \sigma(a,b) \cdot \mathsf{d}^p(a,b)\right)^{1/p} \leq \left(\sum_{e \in E} f^*(e) \cdot w_p(e)\right)^{1/p}.$$

We now prove the upper bound. Suppose we are given the optimal transport plan σ^*. We construct a flow f with cost at most $(4+\epsilon) \cdot w_p(\sigma^*)$. For each $x, y \in A \times B$, by Lemma 2.3 there exists a path $x \rightsquigarrow y$ in G such that $\sum_{e \in x \rightsquigarrow y} w_p(e) \leq (4+\epsilon) \cdot d(x,y)$. Let $\pi(x,y)$ denote this specific path for each pair x, y.

Initially, set f to be zero everywhere. Then for each pair x, y where $\sigma^*(x,y) > 0$, we increment flow f on every edge along $s \to x$, $y \to t$ and every edge of $\pi(x,y)$ by $\sigma^*(x,y)$. The resulting flow is a max flow since the capacity on every edge $s \to x$ leaving s is $\mu(x)$ and the capacity on every edge $y \to t$ entering t is $\nu(y)$. Moreover by construction of f and the fact that every edge $s \to x$ and $y \to t$ has cost zero, the cost of the flow f is

$$\begin{aligned}
\left(\sum_{e \in E} f(e) \cdot w_p(e)\right)^{1/p} &= \left(\sum_{x,y : \sigma^*(x,y) > 0} \sigma^*(x,y) \cdot \mathsf{d}_{G,p}(x,y)\right)^{1/p} \\
&\leq \left(\sum_{x,y : \sigma^*(x,y) > 0} \sigma^*(x,y) \cdot ((4+\epsilon)^p \cdot \mathsf{d}^p(x,y))\right)^{1/p} \\
&\leq (4+\epsilon) \cdot w_p(\sigma^*) = (4+\epsilon) \cdot W_p(\mu,\nu).
\end{aligned}$$

We note that the minimum cost flow must have a cost no more than f. $\square$

Then the cost analysis for when $p = \infty$ follows in a similar manner.

Proof of Theorem 1.1 for $p = \infty$. By standard analysis of network flow algorithms, since the capacity of each edge $s \to x$ is $\mu(x)$ and the capacity of each edge $y \to t$ is $\nu(y)$, we conclude that the resulting σ is a transport plan. Moreover, given f_{i^*}, the transport plan σ can be constructed in $O(|E|)$ time since every $s - t$ path in G has constant length and at every iteration the flow along at least one edge is decremented to zero. We conclude with a cost analysis of σ.

First, we prove the lower bound. By construction of the graph G, we note that any path from s to t must be of the form $s \to a \rightsquigarrow b \to t$ where $a \rightsquigarrow b$ is a path in G from $a \in A$ to $b \in B$. By Lemma 2.3, conclude that $\sum_{e \in a \rightsquigarrow b} w_1(e) \geq \mathsf{d}_{G,1}(a,b) \geq \mathsf{d}(a,b)$ for any $a \in A, b \in B$ and any path $a \rightsquigarrow b$ in G. Moreover, every edge of the form $a \to a_C$, $b_C \to b$, $s \to a$ and $b \to t$ has cost zero. This implies

$$W_\infty(\mu,\nu) \leq w_\infty(\sigma) = \max_{a,b \in A \times B \,:\, \sigma(a,b) > 0} \mathsf{d}(a,b) \leq \max_{e \in E \,:\, f_{i^*}(e) > 0} w_1(e).$$

We now prove the upper bound. Suppose we are given the optimal transport plan σ^*. We construct a flow f with cost at most $(4+\epsilon) \cdot w_\infty(\sigma^*)$. For each $x, y \in A \times B$, by Lemma 2.3 there exists a path $x \rightsquigarrow y$ in G such that $\sum_{e \in x \rightsquigarrow y} w_1(e) \leq (4+\epsilon) \cdot \mathsf{d}(x, y)$. Let $\pi(x, y)$ denote this specific path for each pair x, y.

Initially, set f to be zero everywhere. Then for each pair x, y where $\sigma^*(x, y) > 0$, we increment flow f on every edge along $s \to x$, $y \to t$ and every edge of $\pi(x, y)$ by $\sigma^*(x, y)$. The resulting flow is a max flow since the capacity on every edge $s \to x$ leaving s is $\mu(x)$ and the capacity on every edge $y \to t$ entering t is $\nu(y)$. Moreover by construction of f and the fact that every edge $s \to x$, $y \to t$, $x \to a_C$ and $b_C \to y$ has cost zero, the cost of the flow f is

$$
\begin{aligned}
\max_{e \in E \,:\, f(e) > 0} w_1(e) &= \max_{x, y \in A \times B \,:\, \sigma^*(x,y) > 0} \mathsf{d}_{G,1}(x, y) \\
&\leq \max_{x, y \in A \times B \,:\, \sigma^*(x,y) > 0} (4+\varepsilon) \cdot \mathsf{d}(x, y) \\
&\leq (4+\epsilon) \cdot w_\infty(\sigma^*) = (4+\epsilon) \cdot W_\infty(\mu, \nu).
\end{aligned}
$$

We note that the cost of f is at most $(1+\varepsilon)$ times that of f_{i^*} since i^* is the smallest index where f_{i^*} is a max flow, i.e. has total flow 1, by definition. Rescaling ε by some constant gives the desired result. $\square$