Maximum Matchings in Dynamic Graph Streams and the Simultaneous Communication Model

Sepehr Assadi* Sanjeev Khanna* Yang Li* Grigory Yaroslavtsev†

Abstract

We study the problem of finding an approximate maximum matching in two closely related computational models, namely, the dynamic graph streaming model and the simultaneous multi-party communication model. In the dynamic graph streaming model, the input graph is revealed as a stream of edge insertions and deletions, and the goal is to design a small space algorithm to approximate the maximum matching. In the simultaneous model, the input graph is partitioned across k players, and the goal is to design a protocol where the k players simultaneously send a small-size message to a coordinator, and the coordinator computes an approximate matching.

Dynamic graph streams. We resolve the space complexity of single-pass turnstile streaming algorithms for approximating matchings by showing that for any $\epsilon > 0$, $\Theta(n^{2-3\epsilon})$ space is both sufficient and necessary (up to polylogarithmic factors) to compute an n^{ϵ}-approximate matching; here n denotes the number of vertices in the input graph.

The simultaneous communication model. Our results for dynamic graph streams also resolve the (per-player) simultaneous communication complexity for approximating matchings in the edge partition model. For the vertex partition model, we design new randomized and deterministic protocols for k players to achieve an α-approximation. Specifically, for $\alpha \geq \sqrt{k}$, we provide a *randomized* protocol with total communication of $O(nk/\alpha^2)$ and a *deterministic* protocol with total communication of $O(nk/\alpha)$. Both these bounds are tight. Our work generalizes the results established by Dobzinski *et al.* (STOC 2014) for the special case of $k = n$. Finally, for the case of $\alpha = o(\sqrt{k})$, we establish a new lower bound on the simultaneous communication complexity which is super-linear in n.

1 Introduction

As massive datasets become more prevalent, there is a rapidly growing interest in design of sub-linear algorithms (algorithms whose resource requirements are substantially smaller than the input size) for classical problems. Over the years, different computational models for sub-linear algorithms have been studied, focusing on different types of resources. The *streaming* model of computation, formally introduced in the seminal work of [7], is one of the most classical example. In this model, an algorithm is only allowed to make a single or a few passes over the input and the target resource is the amount of *space* being used. The *(multi-party) communication* model is another classical example, in which the goal is to design algorithms (or *protocols*) for k players to compute a function on their combined input; here the target resource is the amount of *communication* between players.

The two models above, along with their seemingly different target resources, turn out to be closely related. For example, any streaming algorithm directly works as a *one-way* communication protocol: each player receives the memory state of the previous player, continues running the streaming algorithm on his own input, and sends his memory state to the next player. Based on this fact, many lower bounds on space requirement of streaming algorithms are established through lower bounds on communication complexity (see e.g. [7]). The recent result of [29] is another example, which shows that any algorithm for turnstile streams (i.e., streams that contain both insertions and deletions) can be turned into a simultaneous protocol (i.e, protocols in which players simultaneously send a message to a central party). In this reduction, the amount of communication sent by each player is essentially the same as the space requirement of the original turnstile algorithm.

Many important graph problems have been studied in both these models, including connectivity, bipartiteness, minimum spanning trees, spanners, sparsifiers, matchings, etc. [4, 5, 26, 33, 21, 22, 14] (see the survey by McGregor [32] for a summary of the results in the streaming model). However, the space/communication complexity of the fundamental problem of *maximum matching* [30] remains unresolved, and the goal of this paper is to further advance our understanding of the matching problem in these two models. Specifically, we will study the space complexity of matchings in the *dynamic graph streaming* model and the communication complexity of *simultaneous protocols* for matchings in the multi-party communication model. In the rest of this section, we formally define these two models, re-

*Department of Computer and Information Science, University of Pennsylvania. Supported in part by National Science Foundation grants CCF-1116961, CCF-1552909, and IIS-1447470.
Email: {sassadi,sanjeev,yangli2}@cis.upenn.edu

†Warren Center for Network and Data Sciences, University of Pennsylvania. Email: grigory@grigory.us.

view related work, and summarize our results.

1.1 Models and previous work

Dynamic graph streams. Two models of graph streams are mainly studied in the literature: in the *insertion-only model*, the stream contains only edge insertions, and in the *dynamic model*, the stream contains both edge insertions and deletions. The focus of this paper is on the dynamic model formally defined below.

A *dynamic graph stream* can be defined as a special case of *turnstile* streams. A turnstile stream $S = \langle a_1, a_2, \ldots, a_t \rangle$ is a sequence of *updates* that defines a d-dimensional vector $\mathbf{x}$. Each update a_k is a tuple (i_k, Δ_k) where $i_k \in [d]$ and $\Delta_k \in \{-1, +1\}$, which changes the i_k-th coordinate of $\mathbf{x}$ by an additive value of Δ_k. Given any function f over the vector $\mathbf{x}$, algorithms that compute $f(\mathbf{x})$ in turnstile streams are called *turnstile algorithms*.

A dynamic graph stream is a turnstile stream that defines edges of a *multi-graph* $G(V, E)$ on n vertices ($V = [n]$). More specifically, the vector $\mathbf{x}$ has $\binom{n}{2}$ dimensions, where each coordinate of $\mathbf{x}$, denoted by $\mathbf{x}_{i,j}$, represents the *multiplicity* of the edge between the vertices i and j. Additionally, it is standard to require that the multiplicity of every edge remains non-negative *throughout* the stream, since most graph problems are undefined when edges have negative multiplicity. We should note that a turnstile algorithm always works in dynamic graph streams, but the reverse is not true. The difference is that an algorithm that works in dynamic graph streams is allowed to abort if negative edge multiplicities are encountered, even if in the final graph, the multiplicity of every edge is non-negative. The notion of *strict turnstile algorithms* precisely captures algorithms in dynamic graph streams. Though, to the best of our knowledge, all algorithms proposed for dynamic graph streams are in fact turnstile algorithms[1].

Matchings have received a lot of attention in the graph stream literature [31, 18, 15, 16, 20, 28, 36, 5, 3, 24, 21, 25, 13, 17, 32, 2]. For single-pass *adversarially ordered* insertion-only streams[2], the best known algorithm that uses $\tilde{O}(n)$ space (i.e., the *semi-streaming model* [18]) achieves a 2-approximation, which is obtained by simply maintaining a maximal matching during the stream. On the negative side, it is shown in [20, 24] that any streaming algorithm that achieves an approximation factor of better than $e/(e-1)$ requires the storage of $n^{1+\Omega(1/\log\log n)}$ bits.

Despite the huge body of work on the matching problem in insertion-only streams, for dynamic graph streams, no non-trivial single-pass streaming algorithm using space $o(n^2)$ was known prior to this work. Resolving the space complexity of matchings in single-pass dynamic graph streams has been posed as an open problem at the Bertinoro workshop on sub-linear and streaming algorithms in 2014 [1]. To the best of our knowledge, the only previous result concerning matchings in the single-pass dynamic graph streams is the recent paper by Chitnis *et al.* [12], which provides an algorithm for computing a maximal matching of size k using $\tilde{O}(nk)$ space.

Multi-party communication. In the *multi-party communication model*, the input (in our case, an input graph) is *adversarially* partitioned across k players and the goal is to design a *protocol* such that the players can jointly compute a function of the original input (in our case, an approximate matching). We further distinguish between two possible ways of partitioning the input graph: in the *edge partition model*, each player holds a subset of edges of an input (multi-) graph while in the *vertex partition model* the input graph must be bipartite and each player holds a distinct subset of vertices on the left together with all their adjacent edges. Two measures of complexity are considered, namely the *total communication*, which is simply the total size of all messages sent to the coordinator, and the *per-player communication* which is the maximum size of the messages sent by every player. We study both *deterministic* and *randomized* protocols (with *public coins*), and solely focus on *simultaneous protocols*, where every player simultaneously sends a message to a *coordinator* who outputs the final answer. Note that simultaneous protocols, in addition to their aforementioned connection to turnstiles algorithms, are indeed more preferable in distributed settings since they are naturally round-efficient [35].

Matching in the multi-party communication model was previously studied under different variations [14, 9, 22, 21]. Huang *et al.* [22] focused on the k-party *message-passing* model (with two-way player-to-player communication) and gave a tight bound of $\Theta\left(\frac{nk}{\alpha^2}\right)$ on the total communication required to compute an α-approximate matching for both vertex partition and edge partition models. This result immediately implies a lower bound of $\Theta\left(\frac{nk}{\alpha^2}\right)$ for simultaneous protocols. However, the protocol in their upper bound is not simultaneous, and thus far it was not known if this lower bound is achievable by simultaneous protocols. The work of [14, 9] considers the bipartite matching

[1] To the best of our knowledge the only exception is [12] (when they consider a *promised problem*). However, a subsequent work [11] achieved the optimal space bound for the same problem via a turnstile algorithm.

[2] A weaker notion of *randomly ordered* streams (which is less relevant to our work) is also often considered. See [28, 25] and references therein.

problem in the n-party vertex partition model (i.e., every player holds a single vertex on the left). In particular, [14] showed that a protocol in which every player simultaneously sends a random incident edge achieves an $O(\sqrt{n})$-approximation, which matches the lower bound of [22]. The authors further studied deterministic protocols and showed an essentially tight bound on per-player communication of $\Theta(n^{1-\epsilon})$ bits for achieving an n^{ϵ}-approximation. However, these results do not directly generalize to an arbitrary number of players.

1.2 Our results and techniques

Dynamic graph streams. We resolve the space complexity of turnstile algorithms for approximating maximum matchings by proving tight upper and lower bounds on the space requirement.

THEOREM 1.1. *There is a single-pass streaming algorithm that takes any $0 < \epsilon < 1$ as input, and outputs an n^{ϵ}-approximate matching with high probability in dynamic graph streams, while using (i) $\tilde{O}(n^{2-3\epsilon})$ space for $\epsilon \le 1/2$ and (ii) $\tilde{O}(n^{1-\epsilon})$ space for $\epsilon > 1/2$. Moreover, the algorithm only maintains a linear sketch and has $\tilde{O}(1)$ update time for each edge insertion/deletion.*

In Section 3.1, we present a *sampling based* algorithm that takes advantage of the well-known linear sketching implementation of ℓ_0-sampler (see Section 2). The algorithm maintains a set of (edge) samplers that are coordinated in such a way that the sampled edges are "well-spread" across different parts of the graph, and hence contain a large matching. We point out that for weighted graphs with poly(n)-bounded weights, the standard "grouping by weight" technique can be used to obtain a similar result for approximating weighted matchings, while increasing the approximation ratio by a factor of $O(\log n)$.

Note that $\Omega(n^{1-\epsilon})$ is always a lower bound for the space requirement since an n^{ϵ}-approximate matching needs to output $\Omega(n^{1-\epsilon})$ edges whenever the optimal matching is of size $\Omega(n)$. Therefore, our algorithm immediately achieves optimal space requirement when $\epsilon \ge 1/2$. More interestingly, we also show a space lower bound of $n^{2-3\epsilon-o(1)}$ when $\epsilon < 1/2$. Since $n^{2-3\epsilon} > n^{1-\epsilon}$ for any $\epsilon < 1/2$, our algorithm in Theorem 1.1 achieves the optimal space bound for any $0 < \epsilon < 1$.

THEOREM 1.2. *For any $\epsilon < 1/2$, any randomized single-pass turnstile algorithm that outputs an n^{ϵ}-approximate matching in dynamic graph streams with a constant probability must have worst case space complexity of $n^{2-3\epsilon-o(1)}$ bits.*

Theorem 1.2 resolves an open problem posed at the Bertinoro workshop on sub-linear and streaming algorithms in 2014 [1], regarding the possibility of having a constant factor approximation algorithm for the maximum matching in $o(n^2)$ space in turnstile streams. To establish Theorem 1.2, as shown by [29], it suffices to prove the same lower bound on the per-player communication complexity of simultaneous protocols in the edge partition model for multi-graphs.

We establish the lower bound of $n^{2-3\epsilon-o(1)}$ bits for simultaneous protocols following the line of work by [20, 24] on using constructions based on Ruzsa-Szemerédi graphs (RS graphs) [34], which are graphs that can be decomposed into large-size induced matchings (see Section 2 for a formal definition). However, our focus is on simultaneous protocols (instead of *one-way* protocols studied previously) and *polynomial* approximation regime (instead of *constant*). We provide a new approach for this setting that benefits from a very dense construction of RS graphs [8] and hence bypass the $n^{1+\Omega(1/\log\log n)}$ barrier in the aforementioned work on the value of the space lower bound.

REMARK 1.1. *Very recently, [6] generalized the characterization of turnstile algorithms in [29] to* strict *turnstile algorithms. Theorems 1.1 and 1.2, combined with their result, resolve the space complexity for matchings in dynamic graph streams by extending Theorem 2 lower bound for turnstile algorithms to strict turnstile algorithms.*

Multi-party communication. The fact that our algorithm for Theorem 1.1 only maintains a linear sketch, together with composability of these sketches (sketches of subgraphs can simply be added to produce a sketch of the full graph) make our results directly applicable for different distributed settings. In particular, for the edge partition model, our result immediately gives a simultaneous protocol for achieving an n^{ϵ}-approximate matching using $\tilde{O}(n^{2-3\epsilon})$ communication per player. As mentioned above, a matching lower bound of $n^{2-3\epsilon-o(1)}$ bits on per-player communication (for multi-graphs) is obtained as a by-product of the proof of Theorem 1.2 (see Theorem 3.1).

For the vertex partition model, we show that the lower bound of $\Omega(nk/\alpha^2)$ proved for the message-passing model in [22] is achievable via (the weaker class of) simultaneous protocols, as long as $\alpha \ge \sqrt{k}$.

THEOREM 1.3. *There exists a randomized simultaneous protocol that for any $k > 1$, computes an $O(\alpha)$-approximate matching in expectation in the k-party vertex partition model, while using total communication of (i) $O(nk/\alpha^2)$ when $\sqrt{k} \le \alpha \le k$, and (ii) $\tilde{O}(n/\alpha)$ when $\alpha > k$.*

Similar to the case of Theorem 1.1, since $\Omega(n/\alpha)$ is always a lower bound on the total communication, our protocol immediately achieves the optimal communication bound for any $\alpha > k$. Moreover, when $\sqrt{k} \leq \alpha \leq k$ and in the meaningful regime where $\alpha \leq \sqrt{n}$, we have $nk/\alpha^2 \geq k$, and hence the total communication is indeed $O(nk/\alpha^2)$, which matches the lower bound of [22].

The core idea in our simultaneous protocol is to send a "random matching" from each player to the coordinator. Note that for the case where $k = n$, since each player only has one vertex on the left, a random matching degrades to a random neighbor (as is used by [14]). However, for arbitrary k, simply sending random neighbors does not result in a protocol with good approximation guarantees. Indeed, we concentrate the bulk of our efforts on both finding a proper definition of random matchings for our purpose, and exploring their underlining structures. We show that picking a random order of the vertices on the right and computing a maximal matching following this order gives a suitable definition of a random matching. Our proof is based on a proper decomposition of the random orders, which allows us to define multiple *independent* events that were originally based on the same random order.

Furthermore, we establish that achieving an $o(\sqrt{k})$ approximation actually requires the (even per-player) communication to be super-linear in n, even when the number of players (k) is a constant. Combined with Theorem 1.3, this shows that for matchings, forcing protocols to be simultaneous is *only* a limitation for achieving $o(\sqrt{k})$-approximation guarantees, and in this case simultaneous protocols require much more communication than interactive ones.

THEOREM 1.4. *For any $k > 1$, there exists a sufficiently large n such that for any simultaneous protocol (possibly randomized) that outputs an $o(\sqrt{k})$-approximate matching in the k-party vertex partition model, in the worst case, at least one player must send $n^{1+\Omega_k(\frac{1}{\log\log n})}$ bits of communication.*

We obtain this result in a similar way as Theorem 1.2 by using a construction based on RS graphs; however, since here the vertices on the left are partitioned across the players, we can only use a restricted class of RS graphs (with no known dense construction). In the proof, we cast the lower bound of the per-player communication in terms of upper bounds on the edge density for this family of RS graphs (see Lemma 4.7).

Similar to [14], we also study the power of deterministic protocols and establish a tight bound of $\Theta(nk/\alpha)$ for the total communication for the case of $\alpha \geq \sqrt{k}$ (Theorem 1.4 establishes the same barrier of $\sqrt{k}$ for deterministic protocols). This generalizes the bounds in [14] to arbitrary values of k.

THEOREM 1.5. *For* deterministic *simultaneous protocols that compute an $O(\alpha)$-approximate matching in the k-party vertex partition model for any $k > 1$, the total communication of (i) $O(nk/\alpha)$ is* sufficient *as long as $\alpha \geq \sqrt{k}$, and (ii) $\Omega(nk/\alpha)$ is* necessary *as long as $\alpha = o(k/\log k)$.*

Part (i) of Theorem 1.5 (the upper bound) is achieved by a novel protocol whereby each player repeatedly finds maximum matchings that matches distinct sets of vertices on the right, and sends all matchings to the coordinator. Part (ii) of this theorem (the lower bound) is proved using a simple combinatorial construction and a fooling set argument.

1.3 Recent related work Independently and concurrently to our work on approximating matchings in dynamic graph streams [10], Chitnis *et al.* [11] and Konrad [27] also obtained new results on this problem. Chitnis *et al.* [11] also shows an n^ϵ-approximation algorithm in dynamic graph streams using $\tilde{O}(n^{2-3\epsilon})$ space for $\epsilon \leq 1/2$ (similar to our Theorem 1.1). Although both the algorithm of [11] and our algorithm are based on sampling, our result is somewhat stronger in that (i) we also obtain an optimal space bound for the regime $\epsilon > 1/2$, and (ii) we achieve an update time of $\tilde{O}(1)$ in contrast to an update time of $O(\texttt{opt})$ achieved by [11] ($\texttt{opt}$ is the maximum matching size). Konrad [27] gives an upper bound of $\tilde{O}(n^{2-2\epsilon})$ on space for n^ϵ-approximation algorithm and a lower bound of $\Omega(n^{3/2-4\epsilon})$. Both our upper and lower bound results (Theorem 1.1 and Theorem 1.2) are stronger than the results established in [27]. We point out that while at a high level, the lower bound approach used in our paper and the one used in [27] are similar, the constructions and techniques are different.

Organization. The rest of the paper is organized as follows. After introducing some preliminaries in Section 2, we state our upper bound result (Theorem 1.1) in Section 3.1 and lower bound result (Theorem 1.2) in Section 3.2 for dynamic graph streams. Then, we present a randomized simultaneous protocol to prove Theorem 1.3 in Section 4.1, along with the lower bound in Theorem 1.4 in Section 4.2. We further present our deterministic protocol in Section 4.3, and the lower bound in Section 4.4 to prove Theorem 1.5. Finally, we conclude the paper in Section 5 with some further directions.

2 Preliminaries

ℓ_0-Samplers. We use the following powerful tool developed in the streaming literature for performing sampling in a turnstile stream.

DEFINITION 2.1. (ℓ_0-SAMPLER [19]) *An ℓ_0-sampler is an algorithm which given access to a turnstile stream on a d-dimensional vector $\boldsymbol{x}$, outputs an index $i \in [d]$, where i is chosen uniformly at random from the* non-zero *entries of the vector $\boldsymbol{x}$.*

We will use each ℓ_0-sampler to recover an edge between a pre-defined set of vertices, if one exists. Since edges are presented in dynamic graph streams, we need to use a linear sketching implementation.

LEMMA 2.1. ([23]) *For any $0 < \delta < 1$, there is a linear sketching implementation of ℓ_0-sampler for vectors in $\mathbb{R}^n$, which fails with probability δ, using $O(\log^2 n \cdot \log(\delta^{-1}))$ bits of space.*

Ruzsa-Szemerédi graphs. For any graph G, a matching M of G is an *induced matching* iff for any two vertices u v that are matched in M, if u and v are not matched to each other, then there is no edge between u and v in G.

DEFINITION 2.2. (RUZSA-SZEMERÉDI GRAPH) *A graph G is an (r, t)-*RS graph*, iff the set of edges in G consists of t pairwise disjoint induced matchings $M_1, \ldots, M_t$, each of size r.*

We refer the interested reader to [8, 20] for more information about RS graphs and their application to different areas of computer science, including proving lower bounds for streaming algorithms. In this paper, we use the construction of (r, t)-RS graphs given by [8], summarized in the following lemma.

LEMMA 2.2. ([8]) *For any sufficiently large N, there exists an (r, t)-RS graph on N vertices with $r = N^{1-o(1)}$ and $r \cdot t = \binom{N}{2} - o(N^2)$.*

We also consider lopsided RS graphs, which play an important role for proving communication lower bounds in the vertex partition model.

DEFINITION 2.3. (LOPSIDED RS GRAPHS) *We say a bipartite graph $G(L, R, E)$ is a (δ, γ, t)-lopsided RS graph, if $|L| \leq |R| \leq \gamma \cdot |L|$, and the edge set E can be partitioned into t induced matchings $M_1, \ldots, M_t$ of size $(1 - \delta)|L|$.*

We use $U(\delta, \gamma, n)$ to denote the maximum number of edges a (δ, γ, t)-lopsided RS graph $G(L, R, E)$ with $|L| = \Theta(n)$ can have. The following lemma is established by [20] (full version), which shows $U(\delta, \Theta(1/\delta), n) = n^{1+\Omega_\delta(1/\log\log n)}$.

LEMMA 2.3. ([20]) *For any* sufficiently small constant $\delta > 0$*, there exists a family of (δ, γ, t)-lopsided RS graphs with parameters $\gamma = \Theta(1/\delta)$ and $t = n^{\Omega_\delta(1/\log\log n)}$.*

3 Matchings in Dynamic Graph Streams

3.1 Space upper bound for n^ϵ-approximation

In this section, we establish Theorem 1.1 by presenting our algorithm for computing an n^ϵ-approximate matching in dynamic graph streams using $\tilde{O}(\max\{n^{2-3\epsilon}, n^{1-\epsilon}\})$ space.

Without loss of generality, we make the following assumptions. First, we assume that the input graph is bipartite; otherwise by applying the standard technique of choosing a random bipartition of the vertices upfront (using a pairwise independent hash function) and only considering edges that cross the bipartition, we can make the graph bipartite, while increasing the approximation ratio by a factor of 2. Moreover, we assume that the algorithm is provided with a value $\tilde{\texttt{opt}}$ that is a 2-approximation of $\texttt{opt}$ (i.e., the size of a maximum matching in G); we can run our algorithm for $O(\log n)$ different estimates of $\texttt{opt}$ in parallel and output the largest matching among the matchings found for all estimates. Finally, to simplify the analysis, we can assume $\texttt{opt} \geq 10^3 \cdot n^\epsilon$, since otherwise a single edge is an $O(n^\epsilon)$-approximation of the maximum matching.

At a high level, our algorithm randomly partitions the vertices on each side into $\Theta(\texttt{opt}/n^\epsilon)$ groups and for each group on the left, it chooses a subset of $\tilde{O}(\texttt{opt}/n^{2\epsilon})$ groups on the right uniformly at random and maintain one ℓ_0-sampler between the left group and each chosen group on the right. At the end of the stream the algorithm samples one edge from each ℓ_0-sampler and computes a maximum matching of these edges. Formally,

Algorithm 1. A single-pass turnstile algorithm for computing n^ϵ-approximate matching

Input: A bipartite graph $G(L, R, E)$ defined by a dynamic graph stream, a parameter $0 < \epsilon < 1$, and a 2-approximation $\tilde{\texttt{opt}}$ of the maximum matching size.
Output: A matching M with size $\Omega(\texttt{opt}/n^\epsilon)$.
- **Pre-processing:**

1. Let: $\gamma = \left\lceil \frac{\tilde{\texttt{opt}}}{n^\epsilon} \right\rceil$, $\beta = 100 \left\lceil \frac{\tilde{\texttt{opt}}}{n^{2\epsilon}} \right\rceil \cdot \log n$.
2. Create two collections $\mathcal{L}$ and $\mathcal{R}$, each containing γ sets (called *groups*). Create two *pairwise independent* hash functions $h_L : L \mapsto \mathcal{L}$ and $h_R : R \mapsto \mathcal{R}$. Each vertex $u \in L$ (resp. $v \in R$) is assigned to the group $h_L(u) \in \mathcal{L}$ (resp. $h_R(v) \in \mathcal{R}$).
3. For each $L_i \in \mathcal{L}$, assign β groups in $\mathcal{R}$ to L_i chosen *independently and uniformly at random* with replacement. For each R_j assigned to L_i,

we say R_j is an *active partner* of L_i and (L_i, R_j) form an *active pair.*

• **Streaming updates:** For each active pair (L_i, R_j), maintain an ℓ_0-sampler over the edges between the vertices assigned to L_i and R_j.

• **Post-processing:** Compute a maximum matching over the edges sampled from ℓ_0-samplers.

We first note that in the following, whenever we use ℓ_0-samplers, we always apply Lemma 2.1 with parameter $\delta = n^{-3}$. Since the number of ℓ_0-samplers used by our algorithm is bounded by $O(n^2)$, with high probability, none of them will fail. Hence we will not explicitly account for the probability of ℓ_0-samplers failure in our proofs.

Algorithm 1 stores two pairwise independent hash functions h_L and h_R to assign vertices to their groups, which requires $O(\log n)$ bits of space, the identities of all active pairs, which requires $\tilde{O}(\gamma\cdot\beta)$ bits, and $O(\gamma\cdot\beta)$ ℓ_0-samplers for the active pairs during the stream, where each requires $O(\log^3 n)$ bits (Lemma 2.1). Hence, the total space complexity of Algorithm 1 is:

$$\tilde{O}(\gamma \cdot \beta) = \begin{cases} \tilde{O}(n^{2-3\epsilon}) & \text{if } \epsilon \le 1/2 \\ \tilde{O}(n^{1-\epsilon}) & \text{otherwise} \end{cases}$$

where we used the obvious bound of $\tilde{\texttt{opt}} = O(n)$ and the fact that $\beta = O(\log n)$ when $\epsilon > 1/2$. Moreover, for any update on any edge (u, v), we apply h_L on u and h_R on v to identify the groups they belongs to, and update the ℓ_0-sampler for the edges between the groups $h_L(u)$ and $h_R(v)$ if they form an active pair. Therefore, the update time of the algorithm is $\tilde{O}(1)$.

Notation. Fix a maximum matching M^* in G (of size $\texttt{opt}$). We say a vertex v is in M^* if v is matched by M^*. For any group $L_i \in \mathcal{L}$, (resp. $R_j \in \mathcal{R}$) each edge in M^* incident on L_i (resp. R_j) is referred to as a *matching edge* of this group. We say an (L_i, R_j) pair is *matchable* if L_i and R_j share at least one matching edge. The general idea is to treat each group as a single vertex (which forms a new graph $\mathcal{G}$), and to show that the ℓ_0-samplers we stored for $\mathcal{G}$ contain an $O(1)$-approximate matching for $\mathcal{G}$ which in turn leads to an $O(n^\epsilon)$-approximate matching in G.

More specifically, we show that there exists a subset of the groups in $\mathcal{L}$ and $\mathcal{R}$ where in the subgraph of $\mathcal{G}$ induced by this subset, each vertex has bounded degree while the total number of edges is sufficiently large. Then using the following well known result (which we give a simple proof here for completeness), we can conclude that Algorithm 1 outputs a large matching in $\mathcal{G}$, which will be an $O(n^\epsilon)$-approximate matching in the original input graph G.

LEMMA 3.1. *Suppose we are given a bipartite graph $G(L, R, E)$ with m edges and maximum degree d; if for every vertex u in L, we pick one edge incident on u uniformly at random, then with probability at least $1-\exp(-m/12d)$, the sampled edges contain a matching of size $\Omega(m/d)$.*

Essentially, the bounded degree is a consequence of randomly grouping the vertices. The number of edges being large is established by arguing that the matching edges of the groups in $\mathcal{L}$ and $\mathcal{R}$ are "well-spread" across $\mathcal{G}$, implying most pair of $\mathcal{L}$ and $\mathcal{R}$ contains at least one edge. Then, we argue that picking active partners for each group in $\mathcal{L}$ (in Algorithm 1) can be interpreted as picking a random neighbor for each vertex on the left of the induced graph. We first give a simple proof for Lemma 3.1.

Proof. (Proof of Lemma 3.1) Since every vertex in L chooses one neighbor, it suffices to show that $\Omega(m/d)$ distinct vertices in R will be chosen. Since every vertex in L has degree at most d, each edge in G is picked with probability at least $1/d$. For any vertex $v \in R$, let d_v be the degree of v. The probability that v is not picked by any vertex in L is at most

$$(1 - 1/d)^{d_v} \le e^{-d_v/d} \le 1 - d_v/2d$$

where the last inequality uses the fact that $e^{-x} \le 1-x/2$ when $x \in [0, 1]$. Hence the expected number of vertices in R that are picked is at least $\sum_{v\in R} d_v/2d = m/2d$. Since a vertex in R being picked is negatively correlated with other vertices in R being picked, by Chernoff bounds, the probability that at least $m/8d$ vertices in R is matched is at least $1 - \exp(-m/12d)$ ■

We now provide a formal proof of Theorem 1.1. We start by examining the number of edges of M^* that end up in different pairs of (L_i, R_j) groups. Since we will only consider the edges in M^* (i.e., the matching edges), and the grouping leads to all edges between each (L_i, R_j) pair treated as a single edge, it is crucial that enough edges in M^* remain in distinct pair of groups.

CLAIM 3.1. *With probability at least 0.5, the number of edges of M^* that appear in different pairs of (L_i, R_j) groups is at least $\min\{\texttt{opt}/32, \gamma^2/2\}$.*

Proof. We will consider two cases. First suppose $\texttt{opt} > 4\gamma^2$. Let $Y_{i,j}$ be the random variable counting the number of edges in M^* that appear in (L_i, R_j). The total number of distinct (L_i, R_j) pairs is γ^2, and each edge in M^* appears in any (L_i, R_j) pair with probability

$1/\gamma^2$. Hence $\mathrm{E}[Y_{i,j}] = \texttt{opt}/\gamma^2 > 4$. Since the end points of any two edges of M^* are independently assigned to the groups $\mathcal{L}$ and $\mathcal{R}$ (using pairwise independent hash functions h_L and h_R), $\mathrm{Var}[Y_{i,j}] \le \mathrm{E}[Y_{i,j}]$. By Chebyshev inequality,

$$\begin{aligned}\Pr(Y_{i,j} = 0) &\le \Pr(|Y_{i,j} - \mathrm{E}[Y_{i,j}]| \ge \mathrm{E}[Y_{i,j}]) \\ &\le \frac{\mathrm{Var}[Y_{i,j}]}{(\mathrm{E}[Y_{i,j}])^2} \le \frac{1}{\mathrm{E}[Y_{i,j}]} \le \frac{1}{4}\end{aligned}$$

Hence, the expected number of (L_i, R_j) pairs that do not contain any edge from M^* is at most $\gamma^2/4$, and by Markov inequality, with probability at least 0.5, the number of (L_i, R_j) pairs that do not contain an edge from M^* is at most $\gamma^2/2$.

Now suppose $\texttt{opt} \le 4\gamma^2$. Consider the first $\texttt{opt}/16$ edges of M^*. For any two edges e_1 and e_2 in M^*, the probability that e_1 and e_2 belong to the same (L_i, R_j) pair, for some L_i and R_j, (i.e., e_1 and e_j collide) is $1/\gamma^2$. Therefore, the expected number of collisions between the first $\texttt{opt}/16$ edges is $(\texttt{opt}/16)^2/\gamma^2 \le \texttt{opt}/64$ (since $\texttt{opt} \le 4\gamma^2$). Hence, with probability at least 0.5, the total number of collision is less than $\texttt{opt}/32$. Since all collisions can be resolved after removing $\texttt{opt}/32$ edges, at least $(\texttt{opt}/16 - \texttt{opt}/32) = \texttt{opt}/32$ edges of M^* are assigned to distinct (L_i, R_j) pairs. ■

In the following, we focus on the case where at least $\min\{\texttt{opt}/32, \gamma^2/2\}$ edges of M^* appears in distinct (L_i, R_j) pairs. By Claim 3.1, this happens with probability at least 0.5. We consider the cases for $\gamma^2/2$ edges (Lemma 3.2) and $\texttt{opt}/32$ edges (Lemma 3.3) separately, and prove that in each case, the algorithm outputs an $O(n^\epsilon)$-approximate matching, hence proving Theorem 1.1.

Lemma 3.2. *If at least $\gamma^2/2$ edges of M^* appears in distinct (L_i, R_j) pairs, then Algorithm 1 outputs a matching of size $\Omega(\texttt{opt}/n^\epsilon)$ with probability at least $1/4$.*

Proof. If at least $\gamma^2/2$ edges of M^* appears in distinct (L_i, R_j) pairs, then at least $1/4$ fraction of the groups in $\mathcal{L}$ (denoted by $\mathcal{L}'$) have at least $\gamma/3$ different matchable groups in R_j. Otherwise, the total number of edges incident on $\mathcal{L}$ is strictly less than

$$\gamma/4 \cdot \gamma + 3\gamma/4 \cdot (\gamma/3) = \gamma^2/2$$

which is a contradiction. Then, the groups $\mathcal{L}'$ and $\mathcal{R}$ forms a graph (treating each group as a singe vertex) with at least $(\gamma/3) \cdot (\gamma/4)$ edges where each vertex has degree at most γ (there are only γ groups on each side).

It remains to show that any L_i in $\mathcal{L}'$ will pick at least one matchable $R_j \in \mathcal{R}$ as an active partner with probability $1 - 1/n^2$, and moreover, the matchable R_j is chosen uniformly at random. We can then apply Lemma 3.1 to complete the argument. Each L_i is matchable to $1/3$ fraction of the groups in $\mathcal{R}$ and since L_i picks more than $6\log n$ active partners (independently and uniformly at random), by Chernoff bounds, L_i will pick a matchable R_j with probability at least $1 - 1/n^2$. By union bound, all L_i's in $\mathcal{L}'$ will pick at least one matchable $R_j \in \mathcal{R}$. Moreover, for each L_i in $\mathcal{L}'$, a matchable R_j would be picked uniformly at randomly from all groups matchable to L_i. Now, by Lemma 3.1, the edges returned by these matchable (L_i, R_j) pairs contain a matching of size $\Omega(\gamma^2/\gamma) = \Omega(\texttt{opt}/n^\epsilon)$ with probability at least $(1 - \exp(-\gamma/(3 \cdot 4 \cdot 12)))$. Since $\gamma = \lceil \tilde{\texttt{opt}}/n^\epsilon \rceil \ge 500$, the probability of failure is at most $1/2$, and the total probability that Algorithm 1 outputs a matching of size $\Omega(\texttt{opt}/n^\epsilon)$ is at least $1/4$. ■

Lemma 3.3. *If at least $\texttt{opt}/32$ edges of M^* appear in distinct (L_i, R_j) pairs, Algorithm 1 outputs a matching of size $\Omega(\texttt{opt}/n^\epsilon)$ with probability at least 0.15.*

Proof. We need some additional definition for this case. We say a group $L_i \in \mathcal{L}$ (resp. $R_j \in \mathcal{R}$) is *good* if the number of vertices in M^* that belong to L_i (resp. R_j) at least $0.999n^\epsilon$ and at most $1.001n^\epsilon$. The rest of the groups are *bad*. We first show that most groups are good.

Claim 3.2. *With probability at least 0.9, at most 0.001 fraction of the groups in $\mathcal{L}$ (resp. $\mathcal{R}$) are bad.*

Proof. We only prove for $\mathcal{L}$ and the same argument works for $\mathcal{R}$, as well. For each group $L_i \in \mathcal{L}$, let X^i be the random variable counting the number of vertices in M^* that are in L_i, we show that

$$\Pr\left(|X^i - n^\epsilon| \ge 0.001n^\epsilon\right) \le 0.0001$$

Then in expectation, at most 0.0001 fraction of the groups in $\mathcal{L}$ are bad, and by Markov inequality, with probability at most $1/10$, more than 0.001 fraction of the groups are bad, which proves the claim.

Let M_L^* be the set of vertices in L that are matched in M^*. For any vertex $u \in M_L^*$, define X_u^i to be the indicator random variable denoting whether u belongs to L_i. We have $X^i = \sum_{u \in M_L^*} X_u^i$. The expectation of X^i is

$$\begin{aligned}\mathrm{E}[X^i] &= \sum_{u \in M_L^*} \mathrm{E}[X_u^i] = |M_L^*| \cdot (1/\gamma) \\ &= \texttt{opt} \cdot (n^\epsilon/\texttt{opt}) = n^\epsilon\end{aligned}$$

Since we use a pairwise independent hash function h_L in Algorithm 1 to assign vertices in L to groups in $\mathcal{L}$,

$\mathrm{Var}[X^i] \leq \mathrm{E}\left[X^i\right]$. By Chebyshev inequality,

$$\Pr\left(|X^i - n^\epsilon| \geq 0.001n^\epsilon\right) \leq$$
$$\Pr\left(|X^i - \mathrm{E}[X^i]| \geq 100n^{\epsilon/2}\right) \leq 0.0001$$

for n being sufficiently large. ■

Consider the joint event that (i) at least $\mathtt{opt}/32$ edges of M^* appear in distinct (L_i, R_j) pairs, (ii) at most 0.0001 fraction of $\mathcal{L}$ are bad, and (iii) at most 0.0001 fraction of $\mathcal{R}$ are bad. By Claim 3.2, this event happens with probability at least $1 - 1/2 - 1/10 - 1/10 = 0.3$. Moreover, the total number of edges in M^* that are incident on good groups in $\mathcal{L}$ is at least $0.999n^\epsilon \cdot 0.999\gamma \geq 0.998\mathtt{opt}$. Therefore, removing the bad groups in $\mathcal{L}$ only removes $0.002\mathtt{opt}$ edges of M^*. Similarly, removing the bad groups in $\mathcal{R}$ only removes $0.002\mathtt{opt}$ edges of M^*. Therefore, in the worst case, the total number of edges in M^* that appear in distinct (L_i, R_j) pairs where both L_i and R_j are good groups is at least $(1/32 - 0.002 \times 2)\mathtt{opt} \geq \mathtt{opt}/40$.

Now, $\mathtt{opt}/40$ edges are incident on (at most) γ groups on each side where each group only incident on at most $1.001n^\epsilon$ of them. Hence, at least 1/80 fraction of the good $\mathcal{L}$ groups (denoted by $\mathcal{L}''$) must be incident on at least $n^\epsilon/100$ edges, since otherwise, the total number of edges incident on $\mathcal{L}$ is strictly less than

$$\gamma/80 \cdot 1.001n^\epsilon + 79\gamma/80 \cdot n^\epsilon/100 < \mathtt{opt}/40$$

For each group L_i in $\mathcal{L}''$, $n^\epsilon/100$ good $\mathcal{R}$ groups are matchable to L_i. Since L_i picks at least $\frac{100 \cdot \mathtt{opt} \log n}{n^{2\epsilon}}$ active partners and each time, the picked active partner is matchable to L_i with probability at least $(n^\epsilon/100)/\gamma = n^{2\epsilon}/(100\mathtt{opt})$, by Chernoff bounds, with high probability, we will pick at least one matchable group in $\mathcal{R}$ for L_i and the first picked matchable group is chosen uniformly at random from all matchable groups of L_i. By Lemma 3.1, Algorithm 1 outputs a matching of size $\Omega(\mathtt{opt}/n^\epsilon)$ with probability at least

$$(1 - \exp(-\mathtt{opt}/(80 \cdot 12n^\epsilon))$$

Since $\mathtt{opt} \geq 10^3 n^\epsilon$, the probability of failure is at most 1/2, and hence the total probability of success is at least $0.3 \times 0.5 = 0.15$. ■

3.2 Space lower bound for n^ϵ-approximation

In this section, we establish Theorem 1.2 which shows that any turnstile algorithm for matchings requires $n^{2-3\epsilon-o(1)}$ space in order to achieve an n^ϵ-approximation. As pointed out in Section 1.2, it suffices to prove the same result for simultaneous protocols in the edge partition model [29] as captured by the following theorem.

THEOREM 3.1. *For any $\epsilon < 1/2$, any public-coin randomized simultaneous protocol that with a constant probability outputs an n^ϵ-approximate matching in the k-party edge partition model, for $k = n^{\epsilon+o(1)}$, has to communicate $n^{2-3\epsilon-o(1)}$ bits from at least one player.*

Note that though we state Theorem 1.2 and Theorem 3.1 for general graphs, using the same reduction mentioned earlier in Section 3.1, the same lower bound also holds for bipartite graphs.

By Yao's minimax principle, it suffices to prove the lower bound on the per-player communication complexity of deterministic protocols for a fixed distribution of the inputs (known to the players). In our hard distribution, intuitively, each player will be given an (r,t)-RS graph with half of the edges discarded uniformly at random from each induced matchings. The final graph is constructed in a correlated way where for each player, only one of the induced matchings is "private" and all other edges will be incident on the same set of vertices. We carefully choose the parameters such that the coordinator has to know the edges of the private induced matchings for outputting a large matching. However, since each player is unaware of the identity of his private matchings, he has to send enough information for recovering a large fraction of the edges from *every* induced matching. We now define this distribution formally.

A hard input distribution. (for any $\epsilon > 0$ and any sufficiently large integer N)

Parameters:

$$r = N^{1-o(1)} \quad t = \frac{\binom{N}{2} - o(N^2)}{r} \quad k = \left(\frac{N^{1+\epsilon}}{r}\right)^{1/(1-\epsilon)}$$
$$n = k \cdot N \quad \alpha = n^\epsilon$$

- Fix an (r,t)-RS graph G^{RS} on N vertices.
- For each player $P^{(i)}$ independently,

1. Let G_i be the input graph of $P^{(i)}$, initialized as a copy of G^{RS} with vertices $V_i = [N]$.
2. Pick $\lambda_i \in [t]$ uniformly at random and let V_i^* be the set of vertices matched in the λ_i-th induced matching of G_i.
3. For each of the t induced matchings of G_i, drop half of the edges uniformly at random.

- Pick a random permutation π of $[n]$. For every player $P^{(i)}$, for each vertex v in $V_i \setminus V_i^*$ with label j, *relabel* v with $\pi(j)$. Enumerate the vertices in V_i^* (from the one with the smallest label to the largest), and relabel the j-th enumerated vertex with $\pi(N+(i-1)\cdot 2r+j)$. In the final graph, the vertices with the same label correspond to the same vertex.

Several remarks are in order. First, one can easily verify the following relation between the parameters,

$$k = \alpha N/r = n^{\epsilon} N^{o(1)} = n^{\epsilon+o(1)}$$

Second, for the choice of the parameters r, t, and N, by Lemma 2.2, an (r,t)-RS graph with N vertices (i.e., G^{RS}) indeed exists. Moreover, note that the labels of the vertices in $V_i \setminus V_i^*$ for all players are assigned in $\pi(1), \ldots, \pi(N)$, and the vertices in V_i^* for each player are assigned *unique* labels in $\pi(N+(i-1)\cdot 2r+1), \ldots, \pi(N+i\cdot 2r)$. Consequently, the final graph is a multi-graph with n vertices and $O(kN^2) = O(k \cdot (n/k)^2) = n^{2-\epsilon-o(1)}$ total number of edges (counting the multiplicities).

Denote by $G(V,E)$ the final graph. We say a vertex $v \in V$ is *good* if it belongs to V_i^* for some $i \in [k]$. For each player $P^{(i)}$, we call the induced matching between the good vertices (of size $r/2$) the *private matching* of $P^{(i)}$. We say that a matching M is *trivial* if M matches no more than N good vertices.

Claim 3.3. *Let M^* be a* maximum matching *in G and M be any* trivial matching*, then*

$$\frac{|M|}{|M^*|} \leq \frac{4}{\alpha}$$

Proof. Since M^* is a maximum matching, it contains at least $k\cdot(r/2)$ edges (just by using all private matchings). On the other hand, since M is a trivial matching, its size is at most the number of vertices shared by all players plus the number of good vertices matched in M, which is at most $2N$. Since $k = \alpha N/r$,

$$\frac{|M|}{|M^*|} \leq \frac{2N}{k\cdot r/2} = \frac{2N}{\alpha N/2} = \frac{4}{\alpha}$$

∎

Our goal from now on is to show that if the players do not communicate enough information, all that the coordinator can do is to output a trivial matching. Fix a player $P^{(i)}$. According to the distribution, the input to $P^{(i)}$ is a graph G_i (which is a subgraph of the final graph G) obtained by dropping edges from a copy of G^{RS} with vertices labeled by $[N]$ (where we denote by H_i the graph after dropping edges), and relabeling each vertex from $[N]$ to $[n]$ (where we denote the *relabeling function* by σ_i). Therefore, we can formally define the input to $P^{(i)}$ as a pair $(H_i, \sigma_i) \in \mathcal{G} \times \Sigma_i$ where $\mathcal{G}$ is the family of all possible graphs after dropping half of the edges from each induced matching of G^{RS}, and Σ_i is the set of all possible *relabeling functions*. Note that each relabeling function σ_i is defined by a permutation π of $[n]$ and $\lambda_i \in [t]$ chosen in the distribution. A crucial observation for our analysis is that the input to each player $P^{(i)}$ is chosen uniformly at random from the product distribution $\mathcal{G} \times \Sigma_i$, *independent* of the value of λ_i. However, note that the relabeling functions given to different players are indeed correlated.

In what follows, we prove two general lemmas required for the proof of main theorem. Since these lemmas are also required for the proof of Theorem 1.4, they are stated in a slightly more general way. In particular, they work for any family $\mathcal{G}$ that is obtained from dropping half of the edges from each induced matching of *any fixed RS graph* (rather than just G^{RS}).

For any subset $F \subseteq \mathcal{G}$, we define the graph G_F as the *intersection graph* of all graphs in F, i.e., an edge belongs to G_F iff it belongs to every graph in F. The following lemma states that if F is large enough, then most of the t induced matchings in G_F are small.

Lemma 3.4. *For any subset $F \subseteq \mathcal{G}$, and any two integers $a \leq r, b \leq r/2a$, let $I_b \subseteq [t]$ be the set of indices such that for all $j \in I_b$, the intersection graph G_F contains at least $(b\cdot r)/a$ edges from the j-th induced matching; if $|F| \geq 2^{\left(-\frac{r.t}{4a\log n}\right)}|\mathcal{G}|$, then $|I_b| \leq \frac{t}{4b\log n}$.*

Proof. First, notice that $|\mathcal{G}| = \binom{r}{\frac{r}{2}}^t$. Let $\eta = |I_b|$; we can upper bound the size of F as follows:

$$\begin{aligned} |F| &\leq \binom{r - \frac{b\cdot r}{a}}{\frac{r}{2}}^{\eta} \cdot \binom{r}{\frac{r}{2}}^{t-\eta} \\ &\leq \left(2^{-\frac{br}{a}} \cdot \binom{r}{\frac{r}{2}}\right)^{\eta} \cdot \binom{r}{\frac{r}{2}}^{t-\eta} \\ &= 2^{-\frac{br\eta}{a}} \cdot \binom{r}{\frac{r}{2}}^{t} = 2^{-\frac{br\eta}{a}} \cdot |\mathcal{G}| \end{aligned}$$

Therefore, $\eta > \frac{t}{4b\log n}$ implies $|F| < 2^{\left(-\frac{r.t}{4a\log n}\right)}|\mathcal{G}|$; a contradiction. ∎

Lemma 3.5. *For any $a \leq r/100$, suppose every player $P^{(i)}$ sends a message of size at most*

$$s = \frac{r\cdot t}{5a\cdot\log n}$$

bits to the coordinator; then, the expected number of good vertices that are matched in the matching computed by the coordinator is at most $\frac{k\cdot r}{2a}$.

Proof. Fix a player $P^{(i)}$. Let X_i denote the random variable counting the number of good vertices that are matched by the coordinator from the graph G_i provided to the player $P^{(i)}$. In the following, we prove that

$$\underset{G_i}{\mathrm{E}}[X_i] = \underset{(H_i,\sigma_i)}{\mathrm{E}}[X_i] \leq \frac{r}{2a} \tag{3.1}$$

Having this, for $X := \sum_{i\in[k]} X_i$, by linearity of expectation, we have $\mathrm{E}[X] \leq kr/2a$, implying that the expected number of good vertices matched by the coordinator is at most $kr/2a$.

In order to prove $\mathrm{E}[X_i] \leq \frac{r}{2a}$, we show that for any fixed $\sigma \in \Sigma_i$, $\mathrm{E}_{H_i\sim\mathcal{G}}[X_i \mid \sigma_i = \sigma] \leq \frac{r}{2a}$, i.e, conditioning on the relabeling function being σ_i, the coordinator can still only match $\frac{r}{2a}$ good vertices in $P^{(i)}$. Fix a $\sigma \in \Sigma_i$; suppose the coordinator knows all inputs to the players except for the graph H_i given to the player $P^{(i)}$. Note that this is the maximum information the coordinator can obtain from other players. Define $\psi_i : \mathcal{G} \times \Sigma_i \mapsto \{0,1\}^s$ as the deterministic mapping used by the player $P^{(i)}$ to map the input to a s-bit message and send it to the coordinator. Note that since σ is fixed, we can define a function $\phi_i : \mathcal{G} \mapsto \{0,1\}^s$ where $\phi_i(H) = \psi(H,\sigma)$ for any $H \in \mathcal{G}$, and assume that ϕ_i is the function used by the player $P^{(i)}$ when $\sigma_i = \sigma$. We further define the function $\Gamma_i : \mathcal{G} \mapsto 2^{\mathcal{G}}$ such that for any $H \in \mathcal{G}$, $\Gamma_i(H) = \{H' \in \mathcal{G} \mid \phi_i(H') = \phi_i(H)\}$.

The important observation is that since the protocol is deterministic, the coordinator can output an edge $e = (u,v) \in G_i$ as a matching edge for the player $P^{(i)}$, only if $e' = (\sigma^{-1}(u), \sigma^{-1}(v))$ is part of *every* graph in $\Gamma_i(H_i)$, i.e, it belongs to the intersection graph $G_{\Gamma_i(H_i)}$.

We define $\mathcal{E}_i$ to be the event that for the graph H_i, $|\Gamma_i(H_i)| < 2^{(-\frac{r.t}{4a\cdot\log n})}|\mathcal{G}|$. The following claim can be proved using a simple counting argument (proof is deferred to the end of this section).

Claim 3.4. *For any $i \in [k]$, $\Pr(\mathcal{E}_i \mid \sigma_i = \sigma) < \frac{1}{n}$.*

We can write the expected value of X_i as,

$$\begin{aligned}\mathrm{E}[X_i \mid \sigma_i = \sigma] =& \mathrm{E}[X_i \mid \mathcal{E}_i, \sigma_i = \sigma]\cdot\Pr(\mathcal{E}_i \mid \sigma_i = \sigma)\\ &+\mathrm{E}[X_i \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma]\cdot\Pr(\bar{\mathcal{E}}_i \mid \sigma_i = \sigma)\end{aligned} \tag{3.2}$$

By Claim 3.4, the first term in this equation is less than 1. Since $a \leq r/100$ by lemma assumption and our goal is to show that $E[X_i \mid \sigma_i = \sigma] \leq r/2a$, we can safely ignore this additive value of 1. We now bound the second term. We have:

$$\begin{aligned}\mathrm{E}[X_i \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma] &= \sum_{j=1}^{n} j\cdot\Pr(X_i = j \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma)\\ &\leq \sum_{\ell=1}^{\log n}\frac{2^{\ell+1}r}{a}\Pr(X_i \geq \frac{2^{\ell}r}{a} \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma)\end{aligned} \tag{3.3}$$

We can upper bound $\Pr(X_i \geq \frac{2^{\ell}r}{a} \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma)$ for any $\ell \geq 0$ as follows. Let $F = \Gamma_i(H_i)$; the event $\bar{\mathcal{E}}_i$ implies that $|F| \geq 2^{(-\frac{r.t}{4a\cdot\log n})}|\mathcal{G}|$. By applying Lemma 3.4 on the family $\mathcal{G}$ and $F \subseteq \mathcal{G}$ with parameters a and $b = 2^{\ell}$, we have that for I_b defined as in the lemma statement, $|I_b| \leq \frac{t}{4b\log n} = \frac{t}{2^{\ell+2}\log n}$. In the input distribution, it is an easy calculation to see that for any $\lambda \in [t]$, $\Pr(\lambda_i = \lambda \mid \sigma_i = \sigma) = 1/t$. Moreover, since λ_i is chosen independent of H_i (and hence independent of $\bar{\mathcal{E}}_i$), $\Pr\left(\lambda_i = \lambda \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma\right) = \Pr(\lambda_i = \lambda \mid \sigma_i = \sigma) = 1/t$. Hence,

$$\begin{aligned}\Pr(X_i \geq \frac{2^{\ell}r}{a} \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma) &= \Pr(\lambda_i \in I_b \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma)\\ &= \frac{|I_b|}{t} \leq \frac{1}{2^{\ell+2}\log n}\end{aligned} \tag{3.4}$$

By plugging in inequality (3.4) in (3.3) we obtain,

$$\begin{aligned}\mathrm{E}[X_i \mid \bar{\mathcal{E}}_i, \sigma_i = \sigma] &\leq \sum_{\ell=1}^{\log n}\frac{2^{\ell+1}r}{a}\cdot\frac{1}{2^{\ell+2}\log n}\\ &= \sum_{\ell=1}^{\log n}\frac{r}{2a\log n} = \frac{r}{2a}\end{aligned}$$

As $\mathrm{E}[X_i \mid \sigma_i = \sigma] \leq \frac{r}{2a}$ for every $\sigma \in \Sigma_i$, we obtain inequality (3.1). ■

Proof. (Proof of Theorem 3.1) For $\epsilon < 1/2$, we have $\alpha = o(r)$ and hence by Lemma 3.5 with parameter $a = \alpha$, if no player communicates a message of size $\Omega(\frac{r\cdot t}{\alpha\cdot\log n})$ bits, then the expected number of good vertices matched in the matching output by the coordinator is $kr/2\alpha = N/2$ and hence by Markov inequality, the output matching is a trivial matching with probability at least $1/2$. By Claim 3.3, any trivial matching is at most an $(\alpha/4)$-approximation to the maximum matching.

Since $\alpha = n^{\epsilon}$, $k = n^{\epsilon+o(1)}$, $N = n/k$, and $r\cdot t = \Omega(N^2)$ (by Lemma 2.2), we have that any simultaneous protocol that obtains a better than $(n^{\epsilon}/4)$-approximation to the maximum matching with constant probability, has to communicate $n^{2-3\epsilon-o(1)}$ bits from at least one player. Note that by a slight change in the parameters, we obtain the same result for n^{ϵ}-approximation also. ■

Here we provide the deferred proof of Claim 3.4 in Lemma 3.5.

Proof. (Proof of Claim 3.4) Let $o \in \{0,1\}^s$ be the output of the function ϕ_i, and with a slight abuse of notation, we let $\Gamma_i(o) = \Gamma_i(H)$ for any H that $\phi_i(H) = o$. We say o is *light* iff $|\Gamma_i(o)| < 2^{(-\frac{r.t}{4a\cdot\log n})}|\mathcal{G}|$.

We have

$$\begin{aligned}\Pr(\mathcal{E}_i \mid \sigma_i = \sigma) &= \sum_{o \text{ is light}} \Pr_{H\sim\mathcal{G}}(\phi_i(H) = o \mid \sigma_i = \sigma) \\ &= \sum_{o \text{ is light}} \frac{|\Gamma_i(o)|}{|\mathcal{G}|} \\ &\leq 2^{s - \frac{r.t}{4a\cdot\log n}} < \frac{1}{n}\end{aligned}$$

■

4 Matchings in the Simultaneous Communication Model

4.1 Upper bounds for randomized protocols In this section, we establish Theorem 1.3 by presenting a simultaneous protocol for k players to compute an $O(\alpha)$-approximate matching for any $\alpha \geq \sqrt{k}$ using $O(nk/\alpha^2)$ total communication in the vertex partition model. We should note that technically, similar to the case for dynamic graph streams, since n/α could be the size of the target matching, the lower bound of the total communication should be $\Omega(\max\{nk/\alpha^2, n/\alpha\})$. Consequently, when $\alpha \geq k$, $\Omega(n/\alpha)$ becomes the lower bound and when $\alpha \leq k$, $\Omega(nk/\alpha^2)$ is the lower bound. We give a protocol for each regime. The first regime is easier since at least one player contains $1/k$ fraction (which is at least $1/\alpha$ fraction) of any fixed optimum matching. The latter case is much more challenging and it is indeed the main contribution of this section.

4.1.1 A protocol with $\mathbf{O(nk/\alpha^2)}$ communication for $\sqrt{\mathbf{k}} \leq \alpha \leq \mathbf{k}$ We introduce the following protocol $\mathcal{P}^{rand}$.

Protocol $\mathcal{P}^{rand}$. A randomized $O(\alpha)$-approximation simultaneous protocol (for $\sqrt{k} \leq \alpha \leq k$).

1. Let L_i be the vertices in L that belong to the i-th player $P^{(i)}$, and let $l_i = |L_i|$.
2. For each player $P^{(i)}$ independently:
   (a) Pick a random permutation $\pi^{(i)}$ of the vertices in R.
   (b) Use $\pi^{(i)}$ to construct a matching M_i as follows: Enumerate the vertices v in R according to the order $\pi^{(i)}$, and match v with any unmatched neighbor if one exists.
   (c) Send the first $\lceil \frac{l_i \cdot k}{\alpha^2} \rceil$ edges of M_i to the coordinator.
3. The coordinator finds a maximum matching M among all received edges.

Since each player only sends a matching of size at most $\lceil \frac{l_i \cdot k}{2\alpha^2} \rceil$, the total communication is $O(nk/\alpha^2 + k)$. Note that in the meaningful regime where $\alpha \leq \sqrt{n}$, $nk/\alpha^2 \geq k$, and the total communication is indeed $O(nk/\alpha^2)$. In the rest of this section, we show that the protocol $\mathcal{P}^{rand}$ outputs an α-approximate matching, and hence prove Theorem 1.3.

Fix a maximum matching M^* of G (of size $\texttt{opt}$). Let $\texttt{opt}_i$ be the number of edges in M^* that belong to the i-th player $P^{(i)}$. A vertex $v \in R$ is said to be *good* for $P^{(i)}$ if v is matched in M^* by an edge in $P^{(i)}$. The vertices in R that are *not good* for $P^{(i)}$ are said to be *bad* for $P^{(i)}$. A few remarks are in order.

Remark 4.1. (*a*) *Each player $P^{(i)}$ will send at least $\left\lceil \frac{\texttt{opt}_i \cdot k}{2\alpha^2} \right\rceil$ ($\leq \left\lceil \frac{\texttt{opt}_i}{2} \right\rceil$ since $\alpha \geq \sqrt{k}$) edges. This is because $\mathcal{P}^{rand}$ can find a maximal matching in $P^{(i)}$, where the size of a maximum matching in $P^{(i)}$ is at least $\texttt{opt}_i$. As it turns out, it suffices for us to only consider the first $\left\lceil \frac{\texttt{opt}_i \cdot k}{2\alpha^2} \right\rceil$ edges sent by $P^{(i)}$.* (*b*) *Without loss of generality, assume $\texttt{opt}_i < n/100$ for each player, since otherwise $P^{(i)}$ will send a matching of size $\frac{\texttt{opt}_i \cdot k}{2\alpha^2} \geq \frac{\texttt{opt}}{200\alpha}$ (since $k \geq \alpha$), which is an $O(\alpha)$-approximation.*

One key component of our analysis is to decompose picking a random permutation $\pi^{(i)}$ into three *independent* components. $\pi^{(i)}_{pos}$: randomly pick $\texttt{opt}_i$ positions in $[n]$ for placing the good vertices. We will refer to the picked positions as the *good positions* and the rest as *bad positions*; $\pi^{(i)}_b$: pick a random permutation of the bad vertices; and $\pi^{(i)}_g$: pick a random permutation of the good vertices. Then, placing the good/bad vertices in the good/bad positions following the orders $\pi^{(i)}_g/\pi^{(i)}_b$ gives the random permutation $\pi^{(i)}$. Observe that the three components $\pi^{(i)}_{pos}$, $\pi^{(i)}_b$, and $\pi^{(i)}_g$ are independent of each other, and hence events defined on different components are independent, which significantly simplifies the analysis. Moreover, we should note that, of course, each player does not know which vertices are good or bad, and hence the decomposition is only for the sake of analysis. We are now ready to prove that $\mathcal{P}^{rand}$ achieves an $O(\alpha)$-approximation.

Proof. (Proof of Theorem 1.3) Define $\mathcal{E}^*_i$ to be the event (on $\pi^{(i)}_b$) that, between L_i and the first n/α (bad) vertices in $\pi^{(i)}_b$, the maximum matching size is at least $\frac{\texttt{opt}_i \cdot k}{3\alpha^2}$. We partition the players into two types based probability of this event: Type 1 are players with $\Pr(\mathcal{E}^*_i) \geq 1/2$ and Type 2 are the rest. Let T_1 (resp. T_2) be the set of players that are Type 1 (resp. Type 2).

Note that the type of each player only depends on the structure of his input graph and not the protocol.

In the following, we consider the case where players in T_1 contain at least $\texttt{opt}/2$ edges of M^* (Lemma 4.1) and the complement case where players in T_2 contain at least $\texttt{opt}/2$ edges of M^* (Lemma 4.3), separately.

LEMMA 4.1. *If* $\sum_{i \in T_1} \texttt{opt}_i \geq \texttt{opt}/2$, *then* $\mathrm{E}[|M|] = \Omega(\texttt{opt}/\alpha)$.

Proof. Let $k_1 = |T_1|$. Without loss of generality, assume that the Type 1 players are $P^{(1)}, P^{(2)}, \ldots, P^{(k_1)}$ and the protocol $\mathcal{P}^{rand}$ is executed for these k_1 players following this specific order. Define S_i (for any $i \in [k_1]$) to be the set of the distinct vertices in R that are matched (and sent) by at least one of the first i players. To simplify the presentation, we further define $S_0 = \emptyset$. Then $\mathrm{E}[|M|] \geq \mathrm{E}[|S_{k_1}|]$ since each vertex in S_{k_1} is matched with a distinct vertex in L. We will prove that for any $i \in [k_1]$, if the size of S_{i-1} is at most $\frac{\texttt{opt}}{30\alpha}$, then the i-th player will match a large number of new vertices in R in expectation. Formally,

LEMMA 4.2. *For any integer* $i \in [k_1]$ *we have,* $\mathrm{E}\left[|S_i \setminus S_{i-1}| \,\middle|\, |S_{i-1}| \leq \frac{\texttt{opt}}{30\alpha}\right] \geq 0.49 \cdot (\frac{\texttt{opt}_i \cdot k}{6\alpha^2} - \frac{\texttt{opt}}{15\alpha^2})$.

Suppose we have Lemma 4.2 and define $\mathcal{E}'$ as the event that there exists an $i \in [k_1]$ where $|S_{i-1}| > \frac{\texttt{opt}}{30\alpha}$. Note that if $\mathcal{E}'$ happens then $|S_{k_1}| > \frac{\texttt{opt}}{30\alpha}$. Hence,

$$
\begin{aligned}
&\mathrm{E}[|S_{k_1}|] = \mathrm{E}\left[|S_{k_1}| \,\middle|\, \mathcal{E}'\right] \cdot \Pr(\mathcal{E}') + \mathrm{E}\left[|S_{k_1}| \,\middle|\, \bar{\mathcal{E}}'\right] \cdot \Pr(\bar{\mathcal{E}}') \\
&\geq \frac{\texttt{opt}}{30\alpha} \cdot \Pr(\mathcal{E}') \\
&\quad + \sum_{i \in [k_1]} \mathrm{E}\left[|S_i \setminus S_{i-1}| \,\middle|\, |S_{i-1}| \leq \frac{\texttt{opt}}{30\alpha}\right] \cdot \Pr(\bar{\mathcal{E}}') \\
&\text{(by Lemma 4.2)} \\
&\geq \frac{\texttt{opt}}{30\alpha} \cdot \Pr(\mathcal{E}') + 0.49 \sum_{i \in [k_1]} \left(\frac{\texttt{opt}_i \cdot k}{6\alpha^2} - \frac{\texttt{opt}}{15\alpha^2}\right) \cdot \Pr(\bar{\mathcal{E}}') \\
&\text{(since } \textstyle\sum_{i \in [k_1]} \texttt{opt}_i \geq \texttt{opt}/2) \\
&\geq \frac{\texttt{opt}}{30\alpha} \cdot \Pr(\mathcal{E}') + 0.49 \left(\frac{\texttt{opt} \cdot k}{12\alpha^2} - \frac{\texttt{opt} \cdot k}{15\alpha^2}\right) \cdot \Pr(\bar{\mathcal{E}}') \\
&\text{(since } k \geq \alpha) \\
&= \frac{\texttt{opt}}{30\alpha} \cdot \Pr(\mathcal{E}') + \Omega\left(\frac{\texttt{opt}}{\alpha}\right) \Pr(\bar{\mathcal{E}}') = \Omega\left(\frac{\texttt{opt}}{\alpha}\right)
\end{aligned}
$$

Hence $\mathrm{E}[|M|] = \Omega(\texttt{opt}/\alpha)$. We now prove Lemma 4.2.

Proof. (Proof of Lemma 4.2) We need to lower bound the expected number of new vertices in R that are matched by $P^{(i)}$ compare to S_{i-1}. For any $\beta \geq 1$, define $g^{(i)}_\beta$ to be the random variable (on $\pi^{(i)}_{pos}$) counting the number of good positions that appear in the first $1/\beta$ fraction of $[n]$. We will consider the joint event that $\mathcal{E}^*_i$ happens and the number of good positions that appear in the first $2/\alpha$ fraction is at most n/α (i.e., $g^{(i)}_{\alpha/2} \leq n/\alpha$).

$$
\begin{aligned}
&\mathrm{E}\left[|S_i \setminus S_{i-1}| \,\middle|\, |S_{i-1}| \leq \frac{\texttt{opt}}{30\alpha}\right] \\
&\geq \mathrm{E}\left[|S_i \setminus S_{i-1}| \,\middle|\, |S_{i-1}| \leq \frac{\texttt{opt}}{30\alpha}, \mathcal{E}^*_i, g^{(i)}_{\alpha/2} \leq n/\alpha\right] \\
&\cdot \Pr\left(\mathcal{E}^*_i, g^{(i)}_{\alpha/2} \leq n/\alpha\right)
\end{aligned}
$$

Since $\mathcal{E}^*_i$ is defined on $\pi^{(i)}_b$ and $g^{(i)}_{\alpha/2}$ is defined on $\pi^{(i)}_{pos}$, they are independent (due to the decomposition of $\pi^{(i)}$). We know that Type 1 players have $\Pr(\mathcal{E}^*_i) \geq 1/2$, so we only need to bound $\Pr\left(g^{(i)}_{\alpha/2} \leq n/\alpha\right)$. Since $\texttt{opt}_i < n/100$ (by Remark 4.1(b)), $\mathrm{E}\left[g^{(i)}_{\alpha/2}\right] < (2/\alpha) \cdot (n/100) = n/50\alpha$. By Markov inequality, $\Pr\left(g^{(i)}_{\alpha/2} \geq n/\alpha\right) \leq 1/50$. Hence,

$$
\begin{aligned}
&\Pr\left(\mathcal{E}^*_i, g^{(i)}_{\alpha/2} \leq n/\alpha\right) \\
&= \Pr(\mathcal{E}^*_i) \cdot \Pr\left(g^{(i)}_{\alpha/2} \leq n/\alpha\right) \geq 1/2 \cdot 49/50 = 0.49
\end{aligned}
$$

It remains to lower bound

$$
\mathrm{E}\left[|S_i \setminus S_{i-1}| \,\middle|\, |S_{i-1}| \leq \frac{\texttt{opt}}{30\alpha}, \mathcal{E}^*_i, g^{(i)}_{\alpha/2} \leq n/\alpha\right]
$$

We only consider the first $2n/\alpha$ vertices of $\pi^{(i)}$, denoted by $\pi^{(i)}[2n/\alpha]$, and analyze two quantities. x: the expected number of vertices in $\pi^{(i)}[2n/\alpha]$ that are matched, and y: the expected number of vertices in $\pi^{(i)}[2n/\alpha]$ that belong to S_{i-1}. We will show that $x \geq \frac{\texttt{opt}_i \cdot k}{6\alpha^2}$ and $y \leq \frac{\texttt{opt}}{15\alpha^2}$. Since $x - y$ is a lower bound of the expected number of new vertices in R that are matched in $P^{(i)}$, this will complete the proof.

For the quantity x, $g^{(i)}_{\alpha/2} \leq n/\alpha$ implies that there are at least n/α bad vertices in $\pi^{(i)}[2n/\alpha]$, and by $\mathcal{E}^*_i$, there is a matching of size at least $\frac{\texttt{opt}_i \cdot k}{3\alpha^2}$ between L_i and the first n/α bad vertices (and hence between L_i and $\pi^{(i)}[2n/\alpha]$). Since $P^{(i)}$ will find a maximal matching, at least $\frac{\texttt{opt}_i \cdot k}{6\alpha^2}$ vertices in $\pi^{(i)}[2n/\alpha]$ will be matched.

For the quantity y, since $|S_{i-1}| \leq \frac{\texttt{opt}}{30\alpha}$, and each vertex in S_{i-1} belongs to $\pi^{(i)}[2n/\alpha]$ with probability $2/\alpha$, the expected number of vertices in S_{i-1} that belong to $\pi^{(i)}[2n/\alpha]$ is at most $\frac{\texttt{opt}}{15\alpha^2}$. Therefore,

$$
\mathrm{E}\left[|S_i \setminus S_{i-1}| \,\middle|\, |S_{i-1}| \leq \frac{\texttt{opt}}{30\alpha}\right] \geq 0.49 \cdot (\frac{\texttt{opt}_i \cdot k}{6\alpha^2} - \frac{\texttt{opt}}{15\alpha^2})
$$

■

We now analyze the case where $\texttt{opt}/2$ edges of M^* belong to the players in T_2.

LEMMA 4.3. *If* $\sum_{i \in T_2} \texttt{opt}_i \geq \texttt{opt}/2$, *then* $\mathrm{E}[|M|] = \Omega(\texttt{opt}/\alpha)$.

Proof. We will show for the Type 2 players that $\Omega(1/\alpha)$ fraction of the good vertices will be matched in expectation. Recall that for any $\beta \geq 1$, $g_\beta^{(i)}$ is the random variable (on $\pi_{pos}^{(i)}$) counting the number of good positions that appear in the first $1/\beta$ fraction of $[n]$. Then $\mathrm{E}\left[g_\beta^{(i)}\right] = \texttt{opt}_i/\beta$. Define $gm^{(i)}$ to be the random variable (on $\pi^{(i)}$) for the number of good vertices that are matched and sent to the coordinator by $P^{(i)}$. Since the size of M is at least the sum of $gm^{(i)}$, our goal is to lower bound $\mathrm{E}\left[gm^{(i)}\right]$. We establish the following key lemma.

LEMMA 4.4. *For any player* $P^{(i)}$ *in* T_2, *any integer* $r \geq 1$, *and any* $\pi_{pos}^{(i)}$ *with* $g_\alpha^{(i)} = r$, $\mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} = r\right] \geq \frac{1}{4}\min\left\{r, \left\lceil \frac{\texttt{opt}_i \cdot k}{6\alpha^2} \right\rceil\right\}$, *where the expectation is taken over* $\pi_b^{(i)}$ *and* $\pi_g^{(i)}$.

Note that $\frac{1}{4}\min\left\{r, \left\lceil \frac{\texttt{opt}_i \cdot k}{6\alpha^2} \right\rceil\right\} \geq \frac{1}{4}$, and sometimes, we will directly use $\frac{1}{4}$ as a lower bound of the target expectation when applying Lemma 4.4. We first demonstrate how to use Lemma 4.4 to prove $\mathrm{E}[|M|] = \Omega(\texttt{opt}/\alpha)$. To see this, we need to further partition the Type 2 players into two sub-types, where for Type $2a$: $\texttt{opt}_i/\alpha \geq 1$, and for Type $2b$: $\texttt{opt}_i/\alpha < 1$. Let the set of players that are in Type $2a$ (resp. Type $2b$) be T_{2a} (resp. T_{2b}). We consider these two sub-types separately.

LEMMA 4.5. *If* $\sum_{i \in T_{2a}} \texttt{opt}_i \geq \texttt{opt}/4$, *then* $\mathrm{E}[|M|] = \Omega(\texttt{opt}/\alpha)$.

Proof. Fix any player $P^{(i)}$ in T_{2a}. We can lower bound the expectation of $gm^{(i)}$ as follows.

$$\mathrm{E}\left[gm^{(i)}\right] \geq \mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} \geq \frac{\texttt{opt}_i}{2\alpha}\right] \cdot \Pr\left(g_\alpha^{(i)} \geq \frac{\texttt{opt}_i}{2\alpha}\right)$$

Since a good position appearing in the first n/α fraction of $[n]$ is negatively correlated to other good position appearing in the first n/α fraction of $[n]$, by Chernoff bounds, $\Pr\left(g_\alpha^{(i)} < \frac{\texttt{opt}_i}{2\alpha}\right) \leq \frac{1}{e^{1/12}}$. Denote by c the constant $\left(1 - \frac{1}{e^{1/12}}\right)$. Hence

$$\mathrm{E}\left[gm^{(i)}\right] \geq \mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} \geq \frac{\texttt{opt}_i}{2\alpha}\right] \cdot c$$

By Lemma 4.4,

$$\begin{aligned}\mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} \geq \frac{\texttt{opt}_i}{2\alpha}\right] \cdot c &\geq \frac{1}{4}\min\left\{\frac{\texttt{opt}_i}{2\alpha}, \left\lceil \frac{\texttt{opt}_i \cdot k}{6\alpha^2} \right\rceil\right\} \cdot c \\ &\geq \frac{c}{4}\frac{\texttt{opt}_i}{6\alpha} = \Omega\left(\frac{\texttt{opt}_i}{\alpha}\right)\end{aligned}$$

Therefore, summing over all players in T_{2a},

$$\begin{aligned}\mathrm{E}\left[\sum_{i \in T_{2a}} gm^{(i)}\right] &= \sum_{i \in T_{2a}} \mathrm{E}\left[gm^{(i)}\right] \\ &= \Omega\left(\sum_{i \in T_{2a}} \frac{\texttt{opt}_i}{\alpha}\right) = \Omega\left(\frac{\texttt{opt}}{\alpha}\right)\end{aligned}$$

■

LEMMA 4.6. *If* $\sum_{i \in T_{2b}} \texttt{opt}_i \geq \texttt{opt}/4$, *then* $\mathrm{E}[|M|] = \Omega(\texttt{opt}/\alpha)$.

Proof. For any player $P^{(i)}$ in T_{2b},

$$\mathrm{E}\left[gm^{(i)}\right] \geq \mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} \geq 1\right] \cdot \Pr\left(g_\alpha^{(i)} \geq 1\right)$$

Since $\texttt{opt}_i/\alpha < 1$, the probability that no good position appears in the first n/α is

$$\begin{aligned}\Pr\left(g_\alpha^{(i)} = 0\right) &= \binom{n - n/\alpha}{\texttt{opt}_i} \Big/ \binom{n}{\texttt{opt}_i} \\ &= \frac{(n - n/\alpha)! \cdot (n - \texttt{opt}_i)!}{n! \cdot (n - n/\alpha - \texttt{opt}_i)!} \\ &= \prod_{j=0}^{n/\alpha - 1} \frac{n - \texttt{opt}_i - j}{n - j} \\ &\leq \left(\frac{n - \texttt{opt}_i}{n}\right)^{n/\alpha} \leq \exp\left(-\frac{\texttt{opt}_i}{n} \cdot \frac{n}{\alpha}\right) \\ &= e^{-\texttt{opt}_i/\alpha} \leq 1 - \frac{\texttt{opt}_i}{2\alpha}\end{aligned}$$

where the last inequality is because $e^{-x} \leq 1 - x/2$ for any $x \in [0, 1]$. Therefore,

$$\begin{aligned}\mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} \geq 1\right] &\cdot \Pr\left(g_\alpha^{(i)} \geq 1\right) \\ &\geq \mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} \geq 1\right] \cdot \frac{\texttt{opt}_i}{2\alpha}\end{aligned}$$

By Lemma 4.4,

$$\mathrm{E}\left[gm^{(i)} \middle| g_\alpha^{(i)} \geq 1\right] \cdot \frac{\texttt{opt}_i}{2\alpha} \geq \frac{1}{4} \cdot \frac{\texttt{opt}_i}{2\alpha} = \Omega\left(\frac{\texttt{opt}_i}{\alpha}\right)$$

Summing over all players in T_{2b},

$$\mathrm{E}\left[\sum_{i \in T_{2b}} gm^{(i)}\right] \geq \Omega\left(\sum_{i \in T_{2b}} \frac{\texttt{opt}_i}{\alpha}\right) = \Omega\left(\frac{\texttt{opt}}{\alpha}\right)$$

■

Proof. (Proof of Lemma 4.4) We need to lower bound the expected number of good vertices that are matched. It suffices for us to only consider this expectation when the event $\mathcal{E}_i^*$ does not happen, i.e., the first n/α bad vertices only have a matching of size less than $\frac{\texttt{opt}_i \cdot k}{3\alpha^2}$ to L_i.

$$\mathrm{E}\left[gm^{(i)} | g_\alpha^{(i)} = r\right] \geq \mathrm{E}\left[gm^{(i)} | g_\alpha^{(i)} = r, \bar{\mathcal{E}_i^*}\right] \cdot \Pr\left(\bar{\mathcal{E}_i^*}\right)$$
$$\geq \mathrm{E}\left[gm^{(i)} | g_\alpha^{(i)} = r, \bar{\mathcal{E}_i^*}\right] \cdot \frac{1}{2}$$

In the following, we claim that when enumerating each of the first $\min\left\{\left\lceil \frac{\texttt{opt}_i \cdot k}{6\alpha^2}\right\rceil, r\right\}$ good positions in the first n/α, (a) $P^{(i)}$ still has the budget to send one more edge, and moreover, (b) with probability at least $1/2$, $\pi_g^{(i)}$ picks a good vertex that has an unmatched neighbor.

To see property (a), when enumerating any of these good positions, the number of vertices in R that are matched is strictly less than $\frac{\texttt{opt}_i \cdot k}{3\alpha^2}$ (which is an upper bound of the number of matched bad vertices) plus $\left\lceil \frac{\texttt{opt}_i \cdot k}{6\alpha^2}\right\rceil - 1$ (which is an upper bound of the number of good vertices that have appeared). Since $\left\lceil \frac{\texttt{opt}_i \cdot k}{6\alpha^2}\right\rceil - 1 \leq \frac{\texttt{opt}_i \cdot k}{6\alpha^2}$, the total number of vertices in R that are matched is strictly less than $\frac{\texttt{opt}_i \cdot k}{2\alpha^2}$. Since $P^{(i)}$ can send at least $\left\lceil \frac{\texttt{opt}_i \cdot k}{2\alpha^2}\right\rceil$ edges, the number of matching edges is strictly less than the budget, and hence $P^{(i)}$ can send at least one more edge.

To see property (b), since, again, at most $\texttt{opt}_i k/3\alpha^2$ bad vertices are matched, and $\texttt{opt}_i k/6\alpha^2$ good vertices have appeared, at least $\texttt{opt}_i/2$ good vertices that have not appeared have the property that the vertices they are matched with in M^* are still unmatched. Hence $\pi_g^{(i)}$ assign a good vertex that can be matched with probability at least $1/2$. Therefore,

$$\mathrm{E}\left[gm^{(i)} | g_\alpha^{(i)} = r, \bar{\mathcal{E}_i^*}\right] \cdot \frac{1}{2} \geq \frac{1}{4} \cdot \min\left\{\left\lceil \frac{\texttt{opt}_i \cdot k}{6\alpha^2}\right\rceil, r\right\}$$

■

4.1.2 A protocol with $\mathbf{O(n/\alpha)}$ communication for $\alpha \geq \mathbf{k}$

We introduce the following protocol $\mathcal{P}_2^{rand}$.

Protocol $\mathcal{P}_2^{rand}$. A randomized $O(\alpha)$-approximation simultaneous protocol (for $\alpha \geq k$).

1. For each player $P^{(i)}$ independently,
   (a) Let l_i be the number of vertices in L that are in $P^{(i)}$.
   (b) Guess the size of a maximum matching, denoted by $\texttt{opt}$, in the input graph G from $\{n/\alpha, n/(2\alpha), n/(4\alpha), \ldots, n\}$.
   (c) For each guessed value of $\texttt{opt}$, denoted by $\tilde{\texttt{opt}}$, toss a biased coin and with probability $\min\{2l_i \log n/\tilde{\texttt{opt}}, 1\}$, find a maximum matching M_i and send the first (at most) $\tilde{\texttt{opt}}/\alpha$ edges of M_i to the coordinator.
2. The coordinator finds a maximum matching among all received edges.

Claim 4.1. *The protocol $\mathcal{P}_2^{rand}$ uses $\tilde{O}(n/\alpha)$ communication.*

Proof. Since there are only $O(\log n)$ different guesses of $\texttt{opt}$, we only need to show that for each guess, $\tilde{\texttt{opt}}$, the total communication is $\tilde{O}(n/\alpha)$. Fix an $\tilde{\texttt{opt}}$, since a player will send at most $\tilde{\texttt{opt}}/\alpha$ edges to the coordinator, we only need to show that only $\tilde{O}(n/\tilde{\texttt{opt}})$ players will pass the coin toss and send a matching to the coordinator. The expected number of players that passed the coin toss is

$$\sum_{i \in [k]} 2l_i \log n/\tilde{\texttt{opt}} = 2n \log n/\tilde{\texttt{opt}} \geq 2 \log n$$

By Chernoff bounds, with high probability, at most $\tilde{O}(n/\tilde{\texttt{opt}})$ players passes the coin toss, and hence the total communication is $\tilde{O}(n/\alpha)$. ■

Claim 4.2. *The protocol $\mathcal{P}_2^{rand}$ outputs a matching of size $\Omega(\texttt{opt}/\alpha)$.*

Proof. If $\texttt{opt} \leq n/\alpha$, since at least one player (say $P^{(i)}$) contains a matching of size at least $\texttt{opt}/k$ (which is at least $\texttt{opt}/\alpha$), it suffices to show that $P^{(i)}$ will send a matching of size at least n/α^2 (which is $\geq \texttt{opt}/\alpha$). When $P^{(i)}$ guesses $\tilde{\texttt{opt}} = n/\alpha$, the probability that $P^{(i)}$ passes the coin toss is at least $2l_i \log n/\tilde{\texttt{opt}} \geq 1$, and $P^{(i)}$ will send a matching of size $\tilde{\texttt{opt}}/\alpha = n/\alpha^2$.

If $\texttt{opt} > n/\alpha$, we argue that when every player guesses an $\tilde{\texttt{opt}} \in [\texttt{opt}/2, \texttt{opt}]$, a matching of size $\Omega(\tilde{\texttt{opt}}/\alpha)$ (which is also $\Omega(\texttt{opt}/\alpha)$) will be sent to the coordinator. Fix a matching M^* in G of size $\tilde{\texttt{opt}}$ and consider the vertices in M that belong to L, denoted by L_{M^*}. L_{M^*} is partitioned across k players. We refer to any player that contains more than $\tilde{\texttt{opt}}/(2\alpha)$ vertices in L_{M^*} as *good*. Since the size of a maximum matching in any good player $P^{(i)}$ is at least the number of vertices in L_M that belong to $P^{(i)}$, any good player that passes the coin toss will send a matching of size at least $\tilde{\texttt{opt}}/(2\alpha)$ to the coordinator, and $\mathcal{P}_2^{rand}$ would then output a matching of size $\Omega(\tilde{\texttt{opt}}/\alpha)$. We only need to show at least one good player will pass the coin toss.

Since the total number of vertices in L_{M^*} that belong to the players that are not good is at most

$k \cdot \tilde{\texttt{opt}}/(2\alpha) \leq \tilde{\texttt{opt}}/2$, at least $\tilde{\texttt{opt}}/2$ vertices in L_{M^*} belongs to the good players. The expected number of good players that passed the coin toss is at least

$$\sum_{P^{(i)} \text{ is good}} 2l_i \log n/\tilde{\texttt{opt}} \geq (\tilde{\texttt{opt}}/2) 2 \log n/\tilde{\texttt{opt}} = \log n$$

By Chernoff bounds, with high probability, at least one of the good players will pass the coin toss. ■

4.2 A lower bound for randomized protocols when $\alpha = \mathbf{o}(\sqrt{\mathbf{k}})$ In this section, we provide a lower bound on the per-player communication required for any α-approximation protocol when $\alpha = o(\sqrt{k})$ in the k-party simultaneous vertex partition model. Our lower bound construction is based on lopsided RS graphs defined in Section 2. We will prove the following general lemma, which will leave Theorem 1.4 as a corollary. Recall that $U(\delta, \gamma, n)$ denotes the maximum number of edges a (δ, γ, t)-lopsided RS graph $G(L, R, E)$ with $|L| = \Theta(n)$ can have. Since typically γ is much larger than δ, for simplicity, in the following we will assume that γ is an integer multiple of δ.

Lemma 4.7. *For any $0 < \delta < 1/2$ and $\gamma > 1$, let $\alpha = \frac{1}{12\delta}$ and $k = \frac{\gamma}{\delta}$. There exists a sufficiently large n such that any protocol that outputs an α-approximate matching in the k-party simultaneous vertex partition model has to communicate:*

$$\Omega\left(\frac{U(\delta, \gamma, \Theta(n/k))}{\alpha \cdot \log n}\right)$$

bits from at least one player.

Before proving Lemma 4.7, we show how it implies Theorem 1.4. Recall that by Lemma 2.3, for any sufficiently small constant $\delta > 0$, there exists a $\gamma = \Theta(1/\delta)$ such that $U(\delta, \gamma, n) = n^{1+\Omega_\delta(\frac{1}{\log \log n})}$. Suppose $\gamma = c/\delta$ (for some constant $c \geq 1$) and let $k = \gamma/\delta$. Consequently, $\delta = \sqrt{\frac{c}{k}}$ and hence $\alpha = \frac{1}{12\delta} = \frac{\sqrt{k}}{c'}$ (for some other constant $c' > 1$). By Lemma 4.7, any protocol that computes an $\frac{\sqrt{k}}{c'}$-approximate matching in the vertex partition model with k players, has to communicate:

$$\left(\frac{(\frac{n}{k})^{1+\Omega_k(\frac{1}{\log \log n})}}{\alpha \cdot \log n}\right) = n^{1+\Omega_k(\frac{1}{\log \log n})}$$

bits from at least one player. Note that in the above equality, we used the fact that n is chosen sufficiently large, *after* fixing the value of k. We now prove Lemma 4.7.

Similar to the proof of Theorem 3.1, in order to prove Lemma 4.7, it suffices to provide a hard input distribution for deterministic protocols that approximate the maximum matching to within a factor of α.

A hard input distribution (for any $\delta, \gamma > 0$, and sufficiently large integer N)

Parameters:

$$\alpha = \frac{1}{12\delta} \quad k = \gamma/\delta \quad r = (1-\delta)N$$
$$n = 2rk + \gamma N \quad t = U(\delta, \gamma, N)/r$$

- Fix a lopsided (δ, γ, t)-RS graph $G^{RS}(L, R, E)$ with $|L| = N$.
- For each player $P^{(i)}$ independently:

1. Let G_i be the input graph of $P^{(i)}$, initialized as a copy of G^{RS} with vertices $V_i = L_i \cup R_i$ such that $L_i = [N]$ and $R_i = [\gamma \cdot N]$.
2. Pick $\lambda_i \in [t]$ uniformly at random and let $V_i^* = L_i^* \cup R_i^*$ be the set of vertices matched in the λ_i-th induced matching of G_i.
3. For each of the t induced matchings of G_i, drop half of the edges uniformly at random.

- Pick two random permutations π_L and π_R of $[n]$. For every player $P^{(i)}$:

1. For each vertex $u \in L_i$ with label j, *relabel* u with $\pi_L((i-1) \cdot N + j)$.
2. For each vertex $v \in R_i \setminus R_i^*$, with label j, relabel v with $\pi_R(j)$.
3. Enumerate the vertices in R_i^* (from the one with the smallest label to the largest) and relabel the j-th enumerate vertex with $\pi_R(\gamma N + (i-1) \cdot 2r + j)$.

The vertices with the same label correspond to the same vertex in the final graph. Note that the vertices in L_i's are pairwise disjoint.

Denote by $G(L, R, E)$ the final graph. Similar to the Section 3.2, we define *good* vertices as vertices in V_i^* for some $i \in [k]$ and *trivial* matchings as matchings which match no more than γN good vertices (instead of N as chosen in Section 3.2). The following claim is analogous to Claim 3.3 for this new distribution.

Claim 4.3. *Let M^* be a* maximum matching *in G and M be any* trivial matching*, then*

$$\frac{|M|}{|M^*|} < \frac{1}{\alpha}$$

Proof. Since M^* is a maximum matching, it contains at least $\frac{k \cdot r}{2}$ edges (just using the induced matching between the good vertices of each player). On the other hand, since M is a trivial matching, its size is at most:

$$\gamma N + k \cdot \delta N + \gamma N$$

where the terms respectively correspond to the number of vertices in R shared by all players, the number of vertices in L that are not good, and the number of good vertices matched in M. Since $k = \gamma/\delta$,

$$\begin{aligned}\frac{|M|}{|M^*|} &\le \frac{\gamma N + k \cdot \delta N + \gamma N}{k \cdot r/2} \\ &\le \frac{2\gamma N + k \cdot \delta N}{k \cdot (1-\delta)N/2} \\ &= \frac{4\delta}{1-\delta} + \frac{2\delta}{1-\delta} < 12\delta = 1/\alpha\end{aligned}$$

where the last inequality is by $\delta < 1/2$. ∎

We are now ready to prove Lemma 4.7.

Proof. (Proof of Lemma 4.7) Similar to the analysis in proof of Theorem 3.1, we can define the input to each player as a pair $(H_i, \sigma_i) \in \mathcal{G} \times \Sigma_i$ where $\mathcal{G}$ is the family of graphs obtained from G^{RS} (this time a lopsided RS graph) and Σ_i is the set of all relabeling functions (defined as before, this time using permutation π_L and π_R). As stated, Lemma 3.4 and Lemma 3.5 hold for any family of graph obtained from a fixed RS graph, and hence we can apply Lemma 3.5 with parameter $a = 12\alpha$, to obtain that if no player communicates a message of size:

$$\Omega\left(\frac{r.t}{a \log n}\right) = \Omega\left(\frac{U(\delta, \gamma, \Theta(n/k))}{\alpha \log n}\right)$$

bits, then the expected number of good vertices matched by the coordinator is $\frac{kr}{2a} = \frac{\gamma}{\delta} \cdot \frac{r}{2} \cdot \frac{1}{12\alpha} \le \frac{N\gamma}{2}$. Consequently, with probability at least $1/2$, the coordinator is only able to output a trivial matching which does not achieve an α-approximation (by Claim 4.3). ∎

4.3 An $\mathbf{O(nk/\alpha)}$ upper bound for deterministic protocols When $\alpha \ge k$, a simple protocol where every player finds a maximum matching and send the first n/α edges will achieve the required approximation and communication. Hence, here we consider the $\alpha < k$ case and introduce the following deterministic protocol $\mathcal{P}^{det}$.

Protocol $\mathcal{P}^{det}$. A deterministic simultaneous protocol for an $O(\alpha)$-approximation.

1. For each player $P^{(i)}$ independently:
   (a) Find a maximum matching M_1 and remove the vertices in R that are matched in M_1; find a maximum matching M_2 in the remaining graph and remove the vertices in R that are matched in M_2; repeat for $\lceil k/\alpha \rceil$ times. We refer to the set of the edges found in this process as a *matching-cover.*
   (b) Send the matching-cover to the coordinator.
2. The coordinator outputs a maximum matching of all received edges.

If we denote by L_i the set of vertices in L that belong to the player $P^{(i)}$ and by l_i the size of L_i, $P^{(i)}$ will send at most $\lceil k/\alpha \rceil \cdot l_i$ edges to the coordinator. Hence the total amount of communication is $(\sum_{i \in [k]} l_i)\lceil k/\alpha \rceil = O(nk/\alpha)$. We now show that the protocol $\mathcal{P}^{det}$ outputs an $O(\alpha)$-approximate matching and hence proving part (i) in Theorem 1.5.

Proof. (Proof of Theorem 1.5, part (i)) Let M be a maximum matching of the edges received by the coordinator and $\texttt{opt}$ be the size of a maximum matching in G. Denote the sets of vertices matched in M by A (in L) and B (in R). No edge between $L \setminus A$ (denoted by $\bar{A}$) and $R \setminus B$ (denoted by $\bar{B}$) is received by the coordinator. We argue that size of M is at least $\texttt{opt}/(12\alpha)$. Suppose, by contradiction, that $|M| = |A| = |B| < \texttt{opt}/(12\alpha)$.

Let M^* be a maximum matching between $\bar{A}$ and $\bar{B}$ in G. Since G has a matching of size $\texttt{opt}$ while A and B each only have at most $\texttt{opt}/(12\alpha)$ vertices, the size of M^* is at least $\texttt{opt} - |A| - |B| \ge \texttt{opt} - 2\frac{\texttt{opt}}{12\alpha} > 5\texttt{opt}/6 > 3\texttt{opt}/4$. Denote the sets of vertices matched in M^* by A^* (in L) and B^* (in R). The vertices in A^* are partitioned between the k players. Denote the vertices in A^* that belong to the player $P^{(i)}$ by A^*_i, and denote by n_i the size of A^*_i. Hence $\sum_{i \in [k]} n_i = |A^*| \ge 3\texttt{opt}/4$. Denote the number of vertices in A that belong to $P^{(i)}$ by a_i. To simplify the presentation, we further assume for each player $P^{(i)}$, if $a_i \ne 0$, n_i is an integer multiple of a_i[3]. Let B^*_i be the set of vertices matched to A^*_i in M^*. We first make the following observation.

Claim 4.4. *For any player $P^{(i)}$, the j-th maximum matching found by $P^{(i)}$, for any $1 \le j \le \lceil k/\alpha \rceil$, must match at least $(n_i - ja_i)$ vertices in $A^*(i)$.*

Proof. Fix a player $P^{(i)}$. Since no edge between A^*_i and B^*_i is sent to the coordinator, if a vertex $v \in B^*_i$ is matched by the matching-cover of $P^{(i)}$, v must be matched to a vertex in A. Since the number of vertices in A that belong to $P^{(i)}$ is a_i, each maximum matching

[3]This can be achieved by removing at most a_i vertices from A^*_i for each player $P^{(i)}$. Since $\sum_{i \in [k]} a_i \le |A|$, the size of the remaining matching in M^* is still at least $\texttt{opt} - 2|A| - |B| > 3\texttt{opt}/4$.

found by $P^{(i)}$ can only match at most a_i of the vertices in B_i^*. Hence when $P^{(i)}$ is finding a maximum matching for the j-th time, at least $(n_i - ja_i)$ vertices in B_i^* are unmatched. Consider the corresponding $(n_i - ja_i)$ vertices in A_i^* that are matched with these $(n_i - ja_i)$ vertices in B_i^* in the matching M^* (denoted by A''). Since A'' is not matched with any vertex in $\bar{B}$, $P^{(i)}$ must match A'' with a set of $(n_i - ja_i)$ vertices in B. ■

As an immediate consequence of Claim 4.4, we can establish the following connection between n_i and a_i for each player.

Lemma 4.8. *For each player $P^{(i)}$, $\min\{n_i/a_i, \lceil k/\alpha\rceil\} \cdot (n_i - a_i)/2 \le |B|$.*

Proof. In the rest of the proof we assume that $n_i \ge 1$ since otherwise the inequality holds trivially. Using Claim 4.4, $P^{(i)}$ matches at least $(n_i - ja_i)$ vertices in $A^*(i)$ in the j-th matching. Since once a vertex v in B is matched, v will be deleted from the graph, these $(n_i - ja_i)$ vertices in B must be distinct for each j.

Let $\beta = \min\{n_i/a_i, \lceil k/\alpha\rceil\}$. Then for the first β times of finding maximum matching,

$$\begin{aligned} |B| &\ge \sum_{j\in[\beta]} (n_i - ja_i) \\ &= \frac{((n_i - a_i) + (n_i - \beta a_i))\,\beta}{2} \\ &\ge \frac{(n_i - a_i)\beta}{2} \end{aligned}$$

where the last inequality is due to $\beta \le n_i/a_i$. ■

To use Lemma 4.8, we partition the players into two groups. The group $\mathcal{G}_1$ contains players $P^{(i)}$ where $n_i/a_i \le \lceil k/\alpha\rceil$, and the group $\mathcal{G}_2$ contains the players where $n_i/a_i > \lceil k/\alpha\rceil$. Since at least one of the two groups contains at least half of the edges in M^*, we consider $\mathcal{G}_1$ containing half of M^* and $\mathcal{G}_2$ containing half of M^* separately and show that for either case $|M| \ge \texttt{opt}/(12\alpha)$.

If $\mathcal{G}_1$ contains half of the edges in M^*, i.e., $\sum_{i\in\mathcal{G}_1} n_i \ge |M^*|/2 \ge 3\texttt{opt}/8 > \texttt{opt}/4$, using Lemma 4.8, we have

$$(n_i/a_i)(n_i - a_i)/2 \le |B| (= |A|)$$

which implies $a_i \ge n_i^2/(2|A| + n_i)$. Note that $n_i \le |A|$ since otherwise the player $P^{(i)}$ contains a matching of size larger than $|A|$, and the edges sent by $P^{(i)}$ alone must contain a matching of size larger than $|A|$, which contradicts the fact that the coordinator outputs a maximum matching of size $|A|$. Therefore, $a_i \ge n_i^2/(3|A|)$. Summing over all players in $\mathcal{G}_1$, we have

$$\begin{aligned} (|A| \ge) \sum_{i\in\mathcal{G}_1} a_i \ge \frac{\sum n_i^2}{3|A|} &= \frac{\sum n_i^2 |\mathcal{G}_1|}{3|A||\mathcal{G}_1|} \\ &\ge \frac{(\sum n_i)^2}{3k|A|} \ge \frac{\texttt{opt}^2}{48k|A|} \end{aligned}$$

where the second inequality is by the Cauchy-Schwarz. Hence, $|A| \ge \texttt{opt}/(\sqrt{48k}) \ge \texttt{opt}/(4\sqrt{3}\alpha) > \texttt{opt}/(12\alpha)$, a contradiction.

If $\mathcal{G}_2$ contains half of the edges in M^*, i.e., $\sum_{i\in\mathcal{G}_2} n_i \ge |M^*|/2 \ge 3\texttt{opt}/8$, Using Lemma 4.8, we have:

$$\lceil k/\alpha\rceil (n_i - a_i)/2 \le |B| (= |A|)$$

which implies $a_i \ge n_i - 2\alpha|A|/k$. Summing over all players in $\mathcal{G}_2$, we have

$$(|A| \ge) \sum_{i\in\mathcal{G}_2} a_i \ge \sum_{i\in\mathcal{G}_2} (n_i - 2\alpha|A|/k) \ge 3\texttt{opt}/8 - 2\alpha|A|$$

which implies $|A|(1 + 2\alpha) \ge 3\texttt{opt}/8$, and hence $|A| \ge \texttt{opt}/(8\alpha)$, a contradiction. ■

4.4 An $\Omega(nk/\alpha)$ lower bound for deterministic protocols In this section, we establish the lower bound part of Theorem 1.5 which shows that any simultaneous deterministic protocol that achieves an α-approximation to the maximum matching must communicate $\Omega(nk/\alpha)$ bits in total.

To simplify the presentation, we prove that achieving an α/c-approximation (for some fixed constant $c > 0$), requires $\Omega(nk/\alpha)$ bits of communication. For an α-approximation, one can simply use $\alpha' = c \cdot \alpha$ to replace α in the following presentation.

Let $\ell = n/\alpha$ and define $\mathcal{F}$ to be the family of all sets $S \subseteq [n]$ with $|S| = \ell$. Suppose the vertices on each side of the input graph are labeled by $[n]$. For any tuple $T = (S_1, \ldots, S_k)$ where each $S_i \in \mathcal{F}$, define G_T to be a bipartite graph partitioned between the k players in a way that the i-th player $P^{(i)}$ is given a complete bipartite graph between a set of distinct vertices L_i of size n/k on the left, and a set of vertices R_i represented by S_i on the right.

The intuition behind the proof is the following. Consider the graph G_T for $T = (S, S, \ldots, S)$, for some $S \in \mathcal{F}$. The maximum matching size in this graph is $O(n/\alpha)$ since all edges are incident on the same set of vertices of size n/α in R. Now, since a "typical" message sent by each player is small (significantly less than $\log|\mathcal{F}|$), each player will send the same message for lots of different subsets $S' \in \mathcal{F}$. Consequently, the coordinator would receive the same exact message for graph G_T and graph $G_{T'}$ for some $T' = (S_1, \ldots, S_k)$

and hence is forced to output a matching of size $O(n/\alpha)$ also for $G_{T'}$. The goal now is to create a graph $G_{T'}$ where each player $P^{(i)}$ would send the same message for S and S_i, and $G_{T'}$ has a matching of size $\Omega(n)$. In the remainder of this section, we make this intuition formal.

Suppose each player $P^{(i)}$ uses a function $\phi_i : \mathcal{F} \mapsto \{0,1\}^s$ (for some $s = O(nk/\alpha)$) to compute the message to send to the coordinator. We define $\Gamma_i(S)$ for any $S \in \mathcal{F}$ to be the set of all subsets $S' \in \mathcal{F}$ where ϕ_i maps S and S' to the same message, i.e., $\Gamma_i(S) = \{S' \in \mathcal{F} \mid \phi_i(S') = \phi_i(S)\}$. We say that a set $S \in \mathcal{F}$ is *heavy* for the player $P^{(i)}$ iff:

$$|\Gamma_i(S)| \geq \frac{\binom{n}{\ell}}{2^\ell}$$

If a set is not heavy, we call it *light*. The following lemma states the main property of heavy sets that we use in our proof.

Lemma 4.9. *Suppose $S \in \mathcal{F}$ is such that S is heavy for at least p players; then there exists a tuple $T = (S_1, \ldots, S_k)$ such that each $S_i \in \Gamma_i(S)$ and the graph G_T has a matching of size $\frac{p \cdot n}{4ek}$.*

Proof. Without loss of generality assume S is heavy for the players $P^{(1)}, \ldots, P^{(p)}$. For any graph G, let $\texttt{opt}(G)$ be the size of the maximum matching in G. Consider a sequence of tuples $T^{(0)}, \ldots, T^{(p)}(= T)$, where $T^{(0)} = (S, \ldots, S)$ and each $T^{(i)}$ is obtained by changing only the i-th index of $T^{(i-1)}$ to be the set $S_i \in \Gamma_i(S)$ that maximizes $\texttt{opt}(G_{T^{(i)}})$.

We argue that at each step $i \in [p]$, if $\texttt{opt}(G_{T^{(i-1)}}) \leq \frac{pn}{ck} \leq \frac{n}{c}$ (for some constant $c > 0$ to be determined later), then $\texttt{opt}(G_{T^{(i)}}) \geq \texttt{opt}(G_{T^{(i-1)}}) + \frac{n}{ck}$. Consequently, the maximum matching size increases by at least $\frac{n}{ck}$ at each step and hence would be at least $\frac{pn}{ck}$ after p steps, proving the lemma. We now prove this claim.

Suppose by contradiction, at some step $i \in [p]$, no set $S' \in \Gamma_i(S)$ can be added to the tuple $T^{(i-1)}$ that increases the matching size in the new graph by $\frac{n}{ck}$. It implies that for $P^{(i)}$, every set $S' \in \Gamma_i(S)$ intersects the set of vertices that are already matched in $G_{T^{(i-1)}}$ (whose size is at most n/c) by at least $\ell - \frac{n}{ck}$. Therefore, we can bound the size of $\Gamma_i(S)$ as follows:

$$|\Gamma_i(S)| \leq \sum_{t=0}^{\frac{n}{ck}} \binom{(1-\frac{1}{c})n}{t} \cdot \binom{\frac{n}{c}}{\ell - t} \leq \frac{n}{ck}\binom{n}{\frac{n}{ck}} \cdot \binom{\frac{n}{c}}{\ell}$$

Since $\alpha = o\left(\frac{k}{\log k}\right)$, we have:

$$\frac{n}{ck}\binom{n}{\frac{n}{ck}} \leq \binom{n}{\frac{n}{ck}}^2 \leq (e \cdot ck)^{\frac{n}{ck}} = 2^{\log{(e \cdot ck)}\frac{n}{ck}} < 2^\ell$$

Moreover, by choosing $c = 4e$ we have,

$$|\Gamma_i(S)| < 2^\ell \cdot \binom{\frac{n}{c}}{\ell} < 2^\ell \cdot \left(\frac{en}{c \cdot \ell}\right)^\ell = \frac{2^\ell}{4^\ell}\left(\frac{n}{\ell}\right)^\ell \leq \frac{2^\ell}{4^\ell}\binom{n}{\ell} = \frac{\binom{n}{\ell}}{2^\ell}$$

contradicting the fact that S was heavy for the player $P^{(i)}$. ■

We now show that under the given constraint on the total communication of the protocol, there exists a heavy set for a sufficiently large number of players.

Lemma 4.10. *Suppose that the total communication of the players to the coordinator is at most*

$$s = \frac{nk}{8\alpha}$$

bits; then there exists a set $S_h \in \mathcal{F}$ that is heavy for at least $k/6$ players.

Proof. We prove this lemma using a simple probabilistic argument. For any $i \in [k]$, let X_i be a random variable indicating the number of bits sent by $P^{(i)}$ when its input set S is chosen uniformly at random from $\mathcal{F}$. Let K be a subset of the players such that for any $i \in K$, $\mathrm{E}[X_i] \leq n/4\alpha$. We first claim that $|K|$ is at least $k/2$. Otherwise, more than $k/2$ players are not in K; define $X := \sum_{i \notin K} X_i$, and we have $\mathrm{E}[X] > nk/8\alpha = s$. Hence there exists an input where the total communication is more than s; a contradiction.

Now, fix any player $i \in K$. We partition the sets in $\mathcal{F}$ into 2^s groups w.r.t. the outputs of ϕ_i. We call these groups *buckets* and for any $x \in \{0,1\}^s$, we use B_x to denote the bucket where for every $S \in B_x$, $\phi_i(S) = x$. We say that a bucket B_x is *light* iff the number of sets in B_x is at most $\frac{\binom{n}{\ell}}{2^\ell}$ (i.e., all sets in B_x are light).

For any set $S \in \mathcal{F}$, define $\mathcal{E}_S$ as the event that $X_i > n/2\alpha = \ell/2$. Since $\mathrm{E}[X_i] \leq n/4\alpha$, by Markov inequality, $\Pr(\mathcal{E}_S) \leq 1/2$. If we pick a set $S \in \mathcal{F}$ uniformly at random, then

$$\begin{aligned}\Pr(S \text{ is light for } P^{(i)}) &= \Pr(S \text{ is light} \mid \mathcal{E}_S) \cdot \Pr(\mathcal{E}_S) \\ &\quad + \Pr(S \text{ is light} \mid \overline{\mathcal{E}_S}) \cdot \Pr(\overline{\mathcal{E}_S}) \\ &\leq \Pr(\mathcal{E}_S) + \Pr(S \text{ is light} \mid \overline{\mathcal{E}_S}) \\ &\leq \frac{1}{2} + \sum_{B_j \text{ is light}} \Pr(S \in B_j \mid \overline{\mathcal{E}_S})\end{aligned}$$

If the player $P^{(i)}$ is communicating only $\ell/2$ bits (event $\overline{\mathcal{E}_S}$), S could only be chosen from $2^{\ell/2}$ different possible buckets. Since the size of a light bucket is at most $\frac{\binom{n}{\ell}}{2^\ell}$, even if all of the $2^{\ell/2}$ buckets are light, we have

$$\begin{aligned}\Pr(S \text{ is light for } P^{(i)}) &\leq \frac{1}{2} + \sum_{B_j \text{ is light}} \Pr(S \in B_j \mid \overline{\mathcal{E}_S}) \\ &\leq \frac{1}{2} + \sum_{B_j \text{ is light}} \frac{\Pr(S \in B_j)}{\Pr(\overline{\mathcal{E}_S})} \\ &\leq \frac{1}{2} + 2^{\ell/2} \left(\frac{\binom{n}{\ell}}{2^\ell} \frac{2}{\binom{n}{\ell}} \right) \leq \frac{2}{3}\end{aligned}$$

Therefore, a uniformly at random chosen set S is heavy for a player $i \in K$ with probability at least $1/3$. Since the number of player in K is at least $k/2$, the expected number of players that S is heavy for is at least $k/6$. Consequently, there exists a set that is heavy for at least $k/6$ players. ■

Proof. (Proof of Theorem 1.5, part (*ii*)) Suppose the total communication of the protocol is $o(nk/\alpha)$; then by Lemma 4.10, there exists a set $S_h \in \mathcal{F}$ such that S_h is heavy for at least $k/6$ players. Consider the graph G_T defined by the k-tuple $T = (S_h, \ldots, S_h)$; all edges are incident on the vertices in S_h and hence the matching size in G_T is at most $|S_h| = \frac{n}{\alpha}$. Therefore, the coordinator, given the message $\langle \phi_1(S_h), \ldots, \phi_k(S_h) \rangle$, should only output a matching of size no more than $\frac{n}{\alpha}$.

However, by Lemma 4.9, there is a tuple $T' = (S_1, \ldots, S_k)$, where each $S_i \in \Gamma_i(S_h)$ and the size of the maximum matching in $G_{T'}$ is at least $\frac{p \cdot n}{4ek} = \frac{n}{24e}$ (here $p = k/6$). Since $S_i \in \Gamma_i(S_h)$ for each i, the messages that the coordinator receives for the graph G'_T is the same as the messages for G_T. Hence the coordinator again can only output a matching of size $\frac{n}{\alpha}$, which is an $\frac{\alpha}{24e}$-approximation to the maximum matching in G'_T. ■

5 Conclusions

In this paper, we resolved the space complexity of single-pass turnstile algorithms for approximating matchings by showing that for any $\epsilon > 0$, $\Theta(n^{2-3\epsilon})$ space is both sufficient and necessary (up to polylogarithmic factors) to compute an n^ϵ-approximate matching.

Our results for dynamic graph streams also resolve the per-player simultaneous communication complexity for approximating matchings in the edge partition model. For the vertex partition model, we established tight bound of $\Theta(nk/\alpha^2)$ (resp. $\Theta(nk/\alpha)$) total communication for simultaneous randomized (resp. deterministic) protocols for k players to compute an $O(\alpha)$-approximate matching, in the regime where $\alpha \geq \sqrt{k}$.

We showed that fundamentally new techniques are required to achieve an $o(\sqrt{k})$-approximation with limited communication since in this regime, any simultaneous protocol must use communication that is super-linear in n, even when k is a constant. We should note that protocols with improved performance in this regime would have immediate implications on the density of lopsided RS graphs.

Acknowledgments

We would like to thank Michael Kapralov and David Woodruff for helpful discussions.

References

[1] Bertinoro workshop 2014, problem 64. `http://sublinear.info/index.php?title=Open_Problems:64`. Accessed: 2015-05-1.

[2] Kook Jin Ahn and Sudipto Guha. Access to data and number of iterations: Dual primal algorithms for maximum matching under resource constraints. *CoRR*, abs/1307.4359, 2013.

[3] Kook Jin Ahn and Sudipto Guha. Linear programming in the semi-streaming model with application to the maximum matching problem. *Inf. Comput.*, 222:59–79, 2013.

[4] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Analyzing graph structure via linear measurements. In *Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 459–467, 2012.

[5] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Graph sketches: sparsification, spanners, and subgraphs. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS*, pages 5–14, 2012.

[6] Yuqing Ai, Wei Hu, and David P. Woodruff. Additive error norm approximation and new characterizations in turnstile streams. *Personal communication*, 2015.

[7] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. In *STOC*, pages 20–29. ACM, 1996.

[8] Noga Alon, Ankur Moitra, and Benny Sudakov. Nearly complete graphs decomposable into large induced matchings and their applications. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC*, pages 1079–1090, 2012.

[9] Noga Alon, Noam Nisan, Ran Raz, and Omri Weinstein. Welfare maximization with limited interaction. *CoRR, abs/1504.01780. To appear in Symposium on Foundations of Computer Science (FOCS)*, 2015.

[10] Sepehr Assadi, Sanjeev Khanna, Yang Li, and Grigory Yaroslavtsev. Tight bounds for linear sketches of approximate matchings. *CoRR*, abs/1505.01467, 2015.

[11] Rajesh Hemant Chitnis, Graham Cormode, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Andrew McGregor, Morteza Monemizadeh, and Sofya Vorotnikova. Kernelization via sampling with applications to dynamic graph streams. *CoRR*, abs/1505.01731, 2015.

[12] Rajesh Hemant Chitnis, Graham Cormode, Mohammad Taghi Hajiaghayi, and Morteza Monemizadeh. Parameterized streaming: Maximal matching and vertex cover. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1234–1251, 2015.

[13] Michael Crouch and Daniel S. Stubbs. Improved streaming algorithms for weighted matching, via unweighted matching. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, pages 96–104, 2014.

[14] Shahar Dobzinski, Noam Nisan, and Sigal Oren. Economic efficiency requires interaction. In *Symposium on Theory of Computing, STOC*, pages 233–242, 2014.

[15] Sebastian Eggert, Lasse Kliemann, and Anand Srivastav. Bipartite graph matchings in the semi-streaming model. In *Algorithms - ESA 2009, 17th Annual European Symposium*, pages 492–503, 2009.

[16] Leah Epstein, Asaf Levin, Julián Mestre, and Danny Segev. Improved approximation guarantees for weighted matching in the semi-streaming model. *SIAM J. Discrete Math.*, 25(3):1251–1265, 2011.

[17] Hossein Esfandiari, Mohammad Taghi Hajiaghayi, Vahid Liaghat, Morteza Monemizadeh, and Krzysztof Onak. Streaming algorithms for estimating the matching size in planar graphs and beyond. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1217–1233, 2015.

[18] Joan Feigenbaum, Sampath Kannan, Andrew McGregor, Siddharth Suri, and Jian Zhang. On graph problems in a semi-streaming model. *Theor. Comput. Sci.*, 348(2-3):207–216, 2005.

[19] Gereon Frahling, Piotr Indyk, and Christian Sohler. Sampling in dynamic data streams and applications. *International Journal of Computational Geometry & Applications*, 18(01n02):3–28, 2008.

[20] Ashish Goel, Michael Kapralov, and Sanjeev Khanna. On the communication and streaming complexity of maximum bipartite matching. In *Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 468–485. SIAM, 2012.

[21] Venkatesan Guruswami and Krzysztof Onak. Superlinear lower bounds for multipass graph processing. In *Proceedings of the 28th Conference on Computational Complexity, CCC*, pages 287–298, 2013.

[22] Zengfeng Huang, Bozidar Radunovic, Milan Vojnovic, and Qin Zhang. Communication complexity of approximate matching in distributed graphs. In *32nd International Symposium on Theoretical Aspects of Computer Science, STACS*, pages 460–473, 2015.

[23] Hossein Jowhari, Mert Sağlam, and Gábor Tardos. Tight bounds for lp samplers, finding duplicates in streams, and related problems. In *Proceedings of the thirtieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, PODS*, pages 49–58, 2011.

[24] Michael Kapralov. Better bounds for matchings in the streaming model. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1679–1697, 2013.

[25] Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Approximating matching size from random streams. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 734–751, 2014.

[26] Michael Kapralov, Yin Tat Lee, Cameron Musco, Christopher Musco, and Aaron Sidford. Single pass spectral sparsification in dynamic streams. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS*, pages 561–570, 2014.

[27] Christian Konrad. Maximum matching in turnstile streams. In *Algorithms - ESA 2015 - 23rd Annual European Symposium*, pages 840–852, 2015.

[28] Christian Konrad, Frédéric Magniez, and Claire Mathieu. Maximum matching in semi-streaming with few passes. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 15th International Workshop, APPROX, and 16th International Workshop, RANDOM*, pages 231–242, 2012.

[29] Yi Li, Huy L. Nguyen, and David P. Woodruff. Turnstile streaming algorithms might as well be linear sketches. In *Symposium on Theory of Computing, STOC*, pages 174–183, 2014.

[30] L. Lovász and D. Plummer. *Matching Theory*. AMS Chelsea Publishing Series. American Mathematical Soc., 2009.

[31] Andrew McGregor. Finding graph matchings in data streams. In *Approximation, Randomization and Combinatorial Optimization, Algorithms and Techniques, 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX and 9th InternationalWorkshop on Randomization and Computation, RANDOM*, pages 170–181, 2005.

[32] Andrew McGregor. Graph stream algorithms: a survey. *SIGMOD Record*, 43(1):9–20, 2014.

[33] Jeff M. Phillips, Elad Verbin, and Qin Zhang. Lower bounds for number-in-hand multiparty communication complexity, made easy. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 486–501, 2012.

[34] Imre Z Ruzsa and Endre Szemerédi. Triple systems with no six points carrying three triangles. *Combinatorics (Keszthely, 1976), Coll. Math. Soc. J. Bolyai*, 18:939–945, 1978.

[35] David P. Woodruff and Qin Zhang. When distributed computation is communication expensive. In *Distributed Computing - 27th International Symposium, DISC*, pages 16–30, 2013.

[36] Mariano Zelke. Weighted matching in the semi-streaming model. *Algorithmica*, 62(1-2):1–20, 2012.