Noisy Boolean Hidden Matching with Applications

Michael Kapralov
EPFL, Lausanne, Switzerland

Amulya Musipatla
Carnegie Mellon University, Pittsburgh, PA, USA

Jakab Tardos
EPFL, Lausanne, Switzerland

David P. Woodruff
Carnegie Mellon University, Pittsburgh, PA, USA

Samson Zhou
Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

The Boolean Hidden Matching (BHM) problem, introduced in a seminal paper of Gavinsky et al. [STOC'07], has played an important role in lower bounds for graph problems in the streaming model (e.g., subgraph counting, maximum matching, MAX-CUT, Schatten p-norm approximation). The BHM problem typically leads to $\Omega(\sqrt{n})$ space lower bounds for constant factor approximations, with the reductions generating graphs that consist of connected components of constant size. The related Boolean Hidden Hypermatching (BHH) problem provides $\Omega(n^{1-1/t})$ lower bounds for $1+O(1/t)$ approximation, for integers $t \geq 2$. The corresponding reductions produce graphs with connected components of diameter about t, and essentially show that long range exploration is hard in the streaming model with an adversarial order of updates.

In this paper we introduce a natural variant of the BHM problem, called noisy BHM (and its natural noisy BHH variant), that we use to obtain stronger than $\Omega(\sqrt{n})$ lower bounds for approximating a number of the aforementioned problems in graph streams when the input graphs consist only of components of diameter bounded by a fixed constant.

We next introduce and study the graph classification problem, where the task is to test whether the input graph is isomorphic to a given graph. As a first step, we use the noisy BHM problem to show that the problem of classifying whether an underlying graph is isomorphic to a complete binary tree in insertion-only streams requires $\Omega(n)$ space, which seems challenging to show using either BHM or BHH.

2012 ACM Subject Classification Theory of computation → Lower bounds and information complexity

Keywords and phrases Boolean Hidden Matching, Lower Bounds, Communication Complexity, Streaming Algorithms

Digital Object Identifier 10.4230/LIPIcs.ITCS.2022.91

Related Version *Full Version*: `https://arxiv.org/pdf/2107.02578.pdf`

1 Introduction

The *streaming model* of computation has emerged as a popular model for processing large datasets. In *insertion-only* streams, sequential updates to an underlying dataset arrive over time and are permanent, while in *dynamic* or *turnstile* streams, the sequential updates to the dataset may be subsequently reversed by future updates. As many modern large datasets are most naturally represented by graphs (e.g., social networks, protein interaction networks, or communication graphs in network monitoring) there has been a substantial amount of recent interest in graph algorithms on data streams, both on insertion-only streams, e.g., [20, 21, 48, 32, 33, 17, 34, 46, 12] and dynamic streams, e.g., [1, 2, 40, 10, 5, 14, 13, 37, 45].

The Boolean Hidden Matching (BHM) problem [41, 22, 8, 23] is an important problem in communication complexity that has been a major tool for showing hardness of approximation in the streaming model for a variety of graph problems, such as triangle counting [31, 30], maximum matching [4, 19, 12], MAX-CUT [34, 42, 35], and maximum acyclic subgraph [26]. In this problem, Alice is given a binary vector x of length n and Bob is given a matching M on $[n] = \{1, 2, \ldots, n\}$ of size $\alpha n/2$ for a small positive constant $\alpha \leq 1$, as well as a binary vector w of length $\alpha n/2$ of labels for the edges of M. Under the promise that either $Mx \oplus w = 0^{\alpha n/2}$ or $Mx \oplus w = 1^{\alpha n/2}$, the goal is for Alice to send a message of minimal length, so that Bob can determine which of the two cases holds, with probability at least $\frac{2}{3}$. Here, and in the rest of the paper, we use M to denote both the matching and its edge incidence matrix, so that Mx is a vector indexed by edges $e = (u, v) \in M$ such that $(Mx)_e = x_u \oplus x_v$.

Observe that if Bob determines the parity $(Mx)_e$ of any edge e in the matching, then Bob can check whether $(Mx)_e \oplus w_e = 0$ or $(Mx)_e \oplus w_e = 1$. Thus, it suffices for Alice to send the parities of enough vertices so that with probability at least $\frac{2}{3}$, the parities of both vertices of some edge are revealed to Bob. Through a straightforward birthday paradox argument, it follows that $O(\sqrt{n})$ communication suffices. [41, 22, 23] showed that this protocol is essentially tight:

▶ **Theorem 1** ([41, 22, 23]). *Any randomized one-way protocol for the Boolean Hidden Matching problem that succeeds with probability at least $\frac{2}{3}$ requires $\Omega(\sqrt{n})$ bits of communication.*

One of the main weaknesses of the Boolean Hidden Matching problem is that its $\Omega(\sqrt{n})$ communication complexity is not strong enough to characterize the complexity of more difficult problems.

To address this shortcoming, Verbin and Yu proposed the Boolean Hidden Hypermatching problem (BHH) [48], in which Alice is given a binary vector x of length $n = kt$ and Bob is given a hypermatching M on $[n]$ in which all hyperedges contain t vertices, as well as a binary vector w of length k. Under the promise that either $Mx \oplus w = 0^k$ or $Mx \oplus w = 1^k$, the goal is for Alice to send some message of minimal length, so that Bob can determine which of the two cases holds, with probability at least $\frac{2}{3}$. Whereas BHM has complexity $\Omega(\sqrt{n})$, [48] showed that BHH has complexity $\Omega(n^{1-1/t})$:

▶ **Theorem 2** ([48]). *Any randomized protocol that succeeds with probability at least $\frac{2}{3}$ for the Boolean Hidden Hypermatching problem with hyperedges that contain t vertices requires Alice to send $\Omega(n^{1-1/t})$ bits of communication.*

Boolean Hidden Hypermatching has been used to show stronger lower bounds for cycle counting [48], maximum matching [4, 19, 12], Schatten p-norm approximation [43], MAX-CUT [34, 42, 35], and testing biconnectivity, cycle-freeness and bipartiteness [27]. The hard distributions on input graphs that are generated by these reductions typically produce a union of connected components of diameter $\Theta(t)$, where distinguishing between the **YES** and **NO** cases of the input distribution intuitively requires exploring rather long paths (of length comparable to the diameter of these components). In this work we give a new communication problem that provides stronger than $\Omega(\sqrt{n})$ lower bounds that at the same time generate graphs with connected components of bounded diameter, and therefore exploit a different source of hardness (similarly, we prove better than $\Omega(n^{1-1/t})$ lower bounds for several of the above problems on graphs with components of diameter bounded by $O(t)$).

In particular, we note that a major weakness of BHH is that this problem only yields lower bounds of $\Omega(n)$ when the size t of the hyperedges satisfies $t = \Omega(\log n)$. Consequently in the resulting reductions, quantities such as the diameter of the graph, or the size of the largest

clique, often also grow as $\Omega(\log n)$, which prevents the usage of BHH in showing hardness of approximation for specific classes of graphs, such as graphs with bounded diameter or bounded clique number. Our new communication problem overcomes this barrier.

1.1 Our Contributions

We first introduce the following natural but novel parametrization of BHM/BHH.

▶ **Definition 3** (Noisy Boolean Hidden Matching). *The p-Noisy Boolean Hidden Matching problem is a two party communication problem, with parameters $p \in [0,1]$, $\alpha \in (0,1]$, and n.*

- *Alice receives a binary string $x \in \{0,1\}^n$ of length n.*
- *Bob receives a matching M of size $\alpha n/2$ on $[n]$, as well as a binary vector of labels of length $\alpha n/2$. In the* **YES** *instance Bob's labels are the true parities of the matching edges, that is Mx. In the* **NO** *instance Bob's labels are Mx plus some independent random noise ($Ber(p)$) in each coordinate.*

Then the goal is for Alice to send a message of minimal length, so that Bob can distinguish between the **YES** *and* **NO** *cases with probability at least $2/3$.*

Observe that setting $p = 1$ recovers the original Boolean Hidden Matching problem. For significantly smaller values of p, the Hamming distance between **YES** and **NO** instance labels decreases correspondingly. Thus while BHM can be viewed as a gap promise problem, the Noisy Boolean Hidden Matching problem essentially parametrizes the gap size. Now it should be apparent that the previously discussed protocol of Alice sending the parities of $\Theta(\sqrt{n})$ vertices should fail for sufficiently small p. By birthday paradox arguments, the parities for $\Theta(\sqrt{n})$ vertices correspond to the observation of parities for roughly $\Theta(1)$ edges. However for $p = o(1)$, Alice and Bob already know the parities of most of the edges, because the vectors generated in the **YES** and **NO** cases corresponding to the possible edge labels only differ in $o(n)$ coordinates. Thus any message of length $O(\sqrt{n})$ sent by Alice is unlikely to be helpful to Bob. Indeed, we show that the communication complexity of the p-Noisy Boolean Hidden Matching problem is generally $\Omega\left(\sqrt{\frac{n}{p}}\right)$. More generally, we define the p-Noisy Boolean Hidden Hypermatching problem as a means to find tradeoffs between the noise p, the complexity of the problem, and the size of hyperedges t (see Section 3).

▶ **Theorem 4.** *For $p = \Omega\left(\frac{1}{n}\right)$, any randomized one-way protocol that succeeds with probability at least $\frac{2}{3}$ for the p-Noisy Boolean Hidden Hypermatching problem on hyperedges with t vertices requires $\Omega\left(n^{1-1/t}p^{-1/t}\right)$ bits of communication.*

Through the flexibility of our p-Noisy Boolean Hidden Hypermatching problem, we show hardness of approximation for graph problems on a parametrized family of inputs, which is beyond the limits of existing BHM/BHH techinques. We first use p-Noisy Boolean Hidden Hypermatching to show hardness of approximation for MAX-CUT in the streaming model on graphs whose connected components have bounded size, which is a significant obstacle for reductions to BHH. Unlike previous reductions from BHM [34], our methods can show nearly linear lower bounds for approximation close to 1 even in this setting:

▶ **Theorem 5.** *Let $2 \le t \le n/10$ be an integer. For $p \in \left[\frac{128t}{n}, \frac{1}{2}\right]$, any one-pass streaming algorithm that outputs a $\left(1 + \frac{p}{14t}\right)$-approximation to the maximum cut with probability at least $\frac{2}{3}$ requires $\Omega\left(n^{1-1/t}p^{-1/t}\right)$ space, even for graphs with components of size bounded by $4t$.*

Similarly, we show hardness of approximation for maximum matching in the streaming model better than $\Omega(\sqrt{n})$ on graphs whose connected components have bounded size, which is again challenging for reductions from either BHH or BHM.

▶ **Theorem 6.** *Let $2 \leq t \leq n/10$ be some integer. For $p \in \left[\frac{128t}{n}, \frac{1}{2}\right]$, any one-pass streaming algorithm that outputs a $\left(1 + \frac{p}{6t}\right)$-approximation to the maximum matching with probability at least $\frac{2}{3}$ requires $\Omega\left(n^{1-1/t}p^{-1/t}\right)$ space, even for graphs with connected components of size bounded by $O(t)$.*

By comparison, for graphs with connected components with sizes bounded by a constant, existing lower bounds for both MAX-CUT and maximum matching only show that $\Omega(n^C)$ space is required, for some constant $C \in (0, 1)$ bounded away from 1.

Our third graph streaming lower bound proves hardness of approximation for maximum acyclic subgraph in the streaming model. [26] showed that an $\frac{8}{7}$-approximation requires $\Omega(\sqrt{n})$ space through a reduction from BHM, but it was not evident how their reduction could be generalized to BHH, due to its hyperedge structure. Instead, we use our p-Noisy Boolean Hidden Matching communication problem to show a fine-grained lower bound for the maximum acyclic subgraph problem with tradeoffs between approximation guarantee and space. Independently, [6] showed a lower bound that $(1 + \epsilon)$-approximation requires $\Omega(n^{1-O(\epsilon^c)})$ space for a fixed constant $c > 0$ through a reduction from their one-or-many cycles communication problem.

▶ **Theorem 7.** *Let $2 \leq t \leq n/10$ be some integer. For $p \in \left[\frac{128}{n}, \frac{1}{2}\right]$, any one-pass streaming algorithm that outputs a $\left(1 + \frac{p}{22}\right)$-approximation to the maximum acyclic subgraph problem with probability at least $\frac{2}{3}$ requires $\Omega\left(\sqrt{\frac{n}{p}}\right)$ space.*

Graph classification in data streams

Finally, we introduce and study the *graph classification problem* in data streams. The isomorphism problem is a important question in computer science and mathematics; given details describing the structure of some object, the goal of the isomorphism problem is to determine whether the object is identical to another object of interest, up to possible permutation of the elements within the object. Formally, the graph classification problem on data streams is defined as follows.

▶ **Definition 8** (Graph classification problem)**.** *In the graph classification problem on data streams the streaming algorithm must output* **YES** *if the input graph belongs to a specified isomorphism class, and output* **NO** *otherwise.*

Graph isomorphism/classification is one of the most fundamental problems in computer science; it is one of the most well-known problems whose computational complexity is unresolved. It asks whether there exists an isomorphism between two given graphs, or more specifically, whether there exists a bijection between the vertices of the two graphs that preserves edges, i.e., the image of adjacent vertices remain adjacent. Surprisingly, very little is known for the graph classification problem in the streaming model, in terms of both upper and lower bounds. We provide a more extensive exposition on graph classification in the full version of the paper [39], along with some surprising upper and lower bounds for simple families of graphs: For example, we show in the full version of the paper [39] that although classifying whether an underlying graph is a line (resp. cycle) of length n can be done using space logarithmic in the number of vertices n, classifying whether an underlying graph is a line (resp. cycle) of some arbitrary length requires $\Omega(n)$ space. Similarly, we show that we can classify whether an underlying graph is isomorphic to a star graph in sublinear space and that we can classify whether an underlying graph is isomorphic to a regular graph in sublinear space.

A special important case of graph classification is graph isomorphism on tree graphs. We show hardness of this problem for complete binary trees on insertion-only streams. In this setting, a stream of insertions and deletions of edges in an underlying graph with n vertices arrives sequentially and the task is to classify whether the resulting graph is isomorphic to a complete binary tree on n vertices. This is a class of tree graphs for which we do not know how to prove hardness of classification using any other technique, showing our communication problem may be useful for ultimately resolving the general graph classification problem on data streams. Although it is possible to produce a lower bound of $\Omega(\sqrt{n})$ space from BHM, it is not evident that reductions from BHH can produce stronger lower bounds. Instead, we use our p-Noisy Boolean Hidden Matching communication problem to show a lower bound of $\Omega(n)$ space for graph classification of complete binary trees even on insertion-only streams.

▶ **Theorem 9.** *Any randomized algorithm on insertion-only streams that correctly classifies, with probability at least $\frac{3}{4}$, whether an underlying graph is a complete binary tree uses $\Omega(n)$ space.*

In fact, we show more general parametrized space lower bounds for testing on streams whether an underlying graph is a complete binary tree or ϵ-far from being a complete binary tree, where ϵ-far is defined as follows:

▶ **Definition 10.** *We say that a graph $G = (V, E_1)$ is ϵ far from another graph $H = (V, E_2)$ on the same vertex set V if at least $\epsilon \cdot |V|$ edge insertions or deletions are required to get from G to H, i.e., $|(E_1 \setminus E_2) \cup (E_2 \setminus E_1)| \geq \epsilon \cdot |V|$.*

▶ **Theorem 11.** *For $\epsilon \in [\frac{512}{n}, \frac{1}{2}]$, any randomized algorithm on insertion-only streams that classifies, with probability at least $\frac{3}{4}$, whether an underlying graph is a complete binary tree or $\epsilon/16$-far from a complete binary tree uses $\Omega\left(\sqrt{n/\epsilon}\right)$ space.*

Related work

Several communication problems inspired by the Boolean Hidden (Hyper)Matching problem have recently been used in the literature to prove tight lower bounds for the single pass or sketching complexity of several graph problems (e.g., [35, 36] for the MAX-CUT problem, [30] for subgraph counting, in [26, 25, 16, 15] for general CSPs). The recent work of [6] gives multipass streaming lower bounds for the space complexity of the aforementioned one-or-many cycles communication problem, which is tightly connected to BHH, extending many of the abovementioned single pass lower bounds to the multipass setting. Although (to the best of our knowledge) property testing for graph isomorphism on streams has not been previously studied, there is an active line of work, e.g., [20, 21, 47, 24, 44, 27, 18, 7] studying property testing on graphs implicitly defined through various streaming models.

1.2 Overview

We first outline the analysis of the Boolean Hidden Matching problem [41, 22, 23] to describe the key differences and novelties in our approach.

A natural extension of BHM analysis and why it fails

Recall that in BHM, Alice receives a binary vector $x \in \{0,1\}^n$, and sends Bob a message m of c bits. Letting $A \subseteq \{0,1\}^n$ denote the indicator of a "typical" message, one shows that for a "typical" matching M of size αn, $\alpha \in (0, 1/2)$, Bob's posterior distribution q

on Mx conditioned on the message received from Alice is close to uniform, which in turn implies that Bob cannot distinguish between $w = Mx$ and $w = Mx \oplus 1^n$. The approach of [22] upper bounds the total variation distance from q to the uniform distribution via the ℓ_2 distance: this lets one upper bound the ℓ_2 distance between Bob's posterior distribution and the uniform distribution in the *Fourier domain*, and then use Cauchy-Schwarz to obtain the required bound of $\Omega(\sqrt{n})$ on the size c of Alice's message.

Since we are trying to obtain an $\Omega(\sqrt{n/p})$ lower bound for the p-noisy Boolean Hidden Matching problem, it becomes clear that one can no longer compare Bob's posterior to the uniform distribution (the bound of $\Omega(\sqrt{n})$ is tight here). Instead, one would like to compare the distribution of Bob's labels in the **YES** case to the same distribution in the **NO** case. A natural approach here is to compare Bob's posterior distribution q to the noisy version of the posterior, and relate these two distributions to the Fourier transform of Alice's message using the noise operator T_ρ for an appropriate choice of the parameter ρ. Interestingly, however, this approach fails: one can verify that the expected ℓ_2^2 (over the randomness of the matching M) distance is too large[1].

Our approach

Our lower bound for the complexity of p-Noisy Boolean Hidden Hypermatching is information-theoretic. We create a product distribution over $\frac{1}{p}$ instances of BHH, where the i-th instance has size roughly distributed as $\mathrm{Bin}\,(n, p)$ and is a **YES** instance with probability $\frac{1}{2}$, and a **NO** instance with probability $\frac{1}{2}$. The distribution of the sizes for each instance allows us to match the distribution of "flipped" edges in the p-Noisy BHH distribution. We would like to use the product distribution to claim that any protocol that solves p-Noisy BHH on a graph with n vertices must also solve $\frac{1}{p}$ instances of BHH on graphs with np vertices; however, communication complexity is not additive so we must instead use (conditional) information complexity.

To this end, we bound the 1-way conditional information cost of any protocol, correct on our distribution of interest, by first using a message compression result of [28] to relate the 1-way conditional information cost of a single instance to the 1-way distributional communication complexity of the instance. Our conditional information cost is conditioned on Bob's inputs. We note that recent work [4] only bounds external information cost and it does not seem immediate how to derive the same lower bound for conditional information cost from their work [3]. We then observe that any protocol which solves the p-Noisy BHH must also solve a constant fraction of the $\frac{1}{p}$ instances of BHH with size $\Omega(np)$, by distributional correctness (note it need not solve all $\frac{1}{p}$ instances). Using the conditional information cost in a direct sum argument, the communication complexity of the protocol must be at least $\frac{1}{p} \cdot \Omega\left((np)^{1-1/t}\right) = \Omega\left(n^{1-1/t}p^{-1/t}\right)$, which lower bounds the communication.

Lower bound applications

Our remaining lower bounds can be shown by generalizing existing BHM or BHH reductions to the p-Noisy BHM or BHH. For the MAX-CUT problem, [34] give a reduction from BHM that creates a connected component with eight edges for each edge m_i in the matching M. In

[1] If $f : \{0,1\}^n \to \{0,1\}$ is the characteristic function of a typical message from Alice, the ℓ_2^2 distance between the two distributions is, up to appropriate scaling factors, equal to the sum of squares of Fourier coefficients $\widehat{f}(s)$, scaled by a $(1-(1-2p)^{|s|}) \approx p \cdot |s|$ factor. While this factor gives us exactly the required p contribution for Fourier coefficients s of Hamming weight 1, the contribution of higher weight terms is much larger, precluding the analysis.

the case $(Mx)_i \oplus w_i = 0$, the resulting graph is bipartite, so that the max cut induced by the connected component is 8. In the case $(Mx)_i \oplus w_i = 1$, the resulting graph is not bipartite, so that the max cut induced by the connected component is at most 7 (see Figure 1). Thus the max cut for $Mx \oplus w = 0^n$ has size $4n$ and the max cut for $Mx \oplus w = 1^n$ has size at most $7n/2$, so any sufficiently small constant factor approximation algorithm to the max cut requires $\Omega(\sqrt{n})$ space. Observe that the same reduction from p-Noisy BHM also works, although since we set $\alpha = 1/2 < 1$, we also have to consider components corresponding to unmatched vertices. Due to the fact that only a p fraction of the components change their contribution from 8 to 7 in the **NO** case, our reduction works for $(1 + \Theta(p))$-approximation. We give a similar argument for p-Noisy BHH, which allows parametrization of the connected component size.

To show hardness of approximation for the maximum matching problem, we use a reduction similar to that of [12]. However, we reduce from p-noisy BHH. We represent each coordinate of Alice's input with a single edge, and represent each hyperedge of Bob with two disjoint cliques. The supports of the cliques are defined in such a way that the resulting connected components have even size exactly if $(Mx)_i \oplus w_i = 0$, in which case they are perfectly matchable (see Figure 2). In the noisy (**NO**) case, however, a p fraction of these components of size $O(t)$ will have odd size, which leads to an overall $(1 + \Theta(p/t))$ factor loss in the size of the maximum matching.

To show hardness of approximation for maximum acyclic subgraph, we use a reduction by [26]. For each $i \in [n]$, the case $(Mx)_i \oplus w_i = 1$ corresponds to an isolated subgraph with eight edges that contains no cycle, so that its maximum acyclic subgraph has size eight. However, the case $(Mx)_i \oplus w_i = 0$ creates an isolated subgraph with eight edges that contains a cycle, so that its maximum acyclic subgraph has size seven (see Figure 3). Thus if $Mx \oplus w = 0^n$ (the **YES** case), then all subgraphs corresponding to matching edges contribute only 7 to the maximum acyclic subgraph. However, in the **NO** case some of these contribute 8, increasing the total size of the maximum acyclic subgraph by a factor $(1 + \Theta(p))$.

To show hardness of classifying whether an underlying graph is a complete binary tree, we use a gadget by [19] that embeds BHM into the bottom layer of a binary tree. For each $i \in [n]$ where n is assumed to be a power of two, the case $(Mx)_i \oplus w_i = 0$ creates two paths of length two, which can be used to extend the binary tree to an additional layer at two different nodes. However, the case $(Mx)_i \oplus w_i = 1$ creates a path of length one and a path of length three, which results in a non-root node having degree two (see Figure 4). Thus the resulting graph is a complete binary tree if and only if $Mx \oplus w = 0^n$. While BHM requires $\Omega(\sqrt{n})$ space to distinguish whether $Mx \oplus w = 0^n$ or $Mx \oplus w = 1^n$, our p-Noisy version of BHM demands $\Omega\left(\sqrt{\frac{n}{p}}\right)$ space to distinguish between the **YES** and **NO** cases. Now for $p = \Theta\left(\frac{1}{n}\right)$, we have $Mx \oplus w \neq 0^n$ with high probability in the **NO** case. Hence, the graph classification problem also solves p-Noisy BHM in this regime of p and requires $\Omega(n)$ space.

2 Preliminaries

We use the notation $[n]$ to denote the set $\{1, 2, \ldots, n\}$. We use $\mathrm{poly}(n)$ to denote a fixed constant degree polynomial in n and $\frac{1}{\mathrm{poly}(n)}$ to denote some arbitrary degree polynomial in n corresponding to the choice of constants in the algorithms. We use $\mathrm{polylog}(n)$ to denote polylogarithmic factors of n. We say an event occurs with high probability if it occurs with

probability at least $1 - \frac{1}{\text{poly}(n)}$. For $x, y \in \{0,1\}$, we use $x \oplus y$ to denote the sum of x and y modulo 2. For $p \in [0,1]$, we use $\text{Ber}(p)$ to denote the Bernoulli distribution, so that random variable $X \sim \text{Ber}(p)$ satisfies that $\mathbf{Pr}[X = 1] = p$ and $\mathbf{Pr}[X = 0] = 1 - p$.

We define α-approximation for maximization problems for one-sided error (as opposed to two-sided errors):

▶ **Definition 12** (α-Approximation for Maximization Problems)**.** *For a parameter $\alpha \geq 1$, we say that an algorithm $\mathcal{A}$ is an α-approximation algorithm for a maximization problem with optimal value* OPT *if $\mathcal{A}$ outputs some value X with $X \leq \text{OPT} \leq \alpha X$.*

3 Noisy Boolean Hidden Hypermatching

We start with the definition of the p-noisy Boolean Hidden Hypermatching problem (p-noisy BHH), then given a simple protocol for it and show that this protocol is asympototically tight.

▶ **Definition 13.** *A t-hypermatching on $[n]$ is a collection of disjoint subsets of $[n]$, each of size t, which we call hyperedges.*

The noisy Boolean Hidden Hypermatching problem is:

▶ **Definition 14.** *For $p \in [0,1]$ and integer $t \geq 2$ the p-Noisy Boolean Hidden t-Hypermatching (p-noisy BHH) is a one way two party communication problem:*

- *Alice gets $x \in \{0,1\}^n$ uniformly at random.*
- *Bob gets M, a t-hypermatching of $[n]$ with $\alpha n/t$ hyperedges, for a constant $\alpha \in (0,1]$. M is considered to be the $\alpha n/t \times n$ incidence matrix of the matching hyperedges. That is, the i^{th} row of M has ones corresponding to the vertices of the i^{th} matching hyperedge, and zeros elsewhere.*
- *Bob also receives labels for each hyperedge m_i. In the* **YES** *case Bob receives edge labels $w = Mx$ (the true parities of each the matching hyperedges with respect to x). In the* **NO** *case (the noisy case) Bob receives labels $w \oplus z \in \{0,1\}^M$, where each z_i is an independent $Ber(p)$ variable.*
- *Output: Bob must determine whether the communication game is in the* **YES** *or* **NO** *case.*

A protocol for p-noisy BHH

We begin by presenting a simple protocol for solving Boolean hidden hypermatching, which turns out to be nearly asymptotically optimal: for some $c \geq t$ Alice sends a set S of c random bits of x to Bob. Bob then takes each hyperedge that is fully supported on S, and verifies that his label is the true parity. If all such labels reflect the true parity, Bob guesses the **YES** case, while if there is a discrepancy Bob guesses the **NO** case.

It is clear that the only way the protocol can fail is if in the **NO** case no hyperedge is simultaneously supported on S and mislabeled. Let us call the event that the i^{th} hyperedge, m_i is supported on S and mislabeled $\mathcal{E}_i$. For each hyperedge of M, the probability of being supported on S is $\binom{c}{t}/\binom{n}{t} \geq c^t/e^t n^t$, while the probability of being mislabeled is independently p. Therefore, $\mathbb{P}(\mathcal{E}_i) \geq p \cdot \left(\frac{c}{en}\right)^t$. Since the events $\mathcal{E}_i$ are negatively correlated we have that

$$\mathbb{P}\left(\bigcup_{i=1}^{\alpha n/t} \mathcal{E}_i\right) \geq 1 - \prod_{i=1}^{\alpha n/t} (1 - \mathbb{P}(\mathcal{E}_i)) \geq 1 - \left(1 - p \cdot \left(\frac{c}{en}\right)^t\right)^{\alpha n/t} \geq 1 - \exp\left(-\frac{p\alpha n}{t} \cdot \left(\frac{c}{en}\right)^t\right)$$

Therefore, if $t \leq n/10$ and $c = \Omega(n^{1-1/t}(p\alpha)^{-1/t})$, this probability is an arbitrarily large constant, and Bob can distinguish between the **YES** and **NO** cases with high constant probability.

Information Complexity of Noisy BHH.

In this section, we lower bound the communication complexity of Noisy BHH through an information-theoretic argument. The core of the proof is a reduction to Boolean Hidden Hypermatching, whose complexity we defined in Theorem 2. We present the proof for the case $\alpha = 1$, where Bob receives a perfect hypermatching. This is the more challenging setting, as we need to be careful about the parity of the noise applied to the labels in the **NO** case, as described below. The easier setting of $\alpha < 1$ does not require this modification to the noise, and we omit the proof.

▶ **Definition 15.** *We define the distribution $\mathcal{Z}_p^n$, as identical to the independent vector of $Ber(p)$ variables of length n, but with the condition that the number of ones is even:*

$$V \sim Ber(p)^n \implies V\Big|2|w_H(V) \sim \mathcal{Z}_p^n$$

where w_H denotes the Hamming weight. More formally, if $Z \sim \mathcal{Z}_p^n$ we have

$$\mathbb{P}(Z = z) = \frac{\mathbb{1}(2|w_H(z))p^{w_H(z)}(1-p)^{n-w_H(z)}}{\sum_{z' \in \{0,1\}^n} \mathbb{1}(2|w_H(z'))p^{w_H(z')}(1-p)^{n-w_H(z')}}.$$

Furthermore, let $|\mathcal{Z}_p^n|$ denote the distribution of the Hamming weight of a variable from $\mathcal{Z}_p^n$.

We are now able to define the distributional communication problem of p-Noisy Perfect Boolean Hidden Hypermatching:

▶ **Definition 16.** *For $p \in [0,1]$ and integer $t \geq 2$ the p-Noisy Perfect Boolean Hidden t-Hypermatching (p-Noisy PBHH) is a one-way two-party communication problem:*

- *Alice gets $x \in \{0,1\}^n$.*
- *Bob gets M, a perfect t-hypermatching of $[n]$ (that is n/t hyperedges).*
- *Bob also receives edge labels $w \in \{0,1\}^{n/t}$. With probability $1/2$, we are in the* **YES** *case, and the edge labels satisfy $Mx = w$. With probability $1/2$, we are in the* **NO** *case and the edge labels satisfy $Mx \oplus z = w$, where $Z \sim \mathcal{Z}_p^{n/t}$.*
- *Output: Bob must determine whether the communication game is in the* **YES** *or* **NO** *case.*

We prove a lower bound on the communication and information complexity of p-Noisy BHH by considering the uniform input distribution: We are in the **YES** and **NO** cases with probability $1/2$ each; x is sampled uniformly at random from all n-length bit strings and M is sampled uniformly at random from all t-hypermatchings of size $\alpha n/t$ on $[n]$. We replace the variables x, w, and z with the random variables X, W and Z respectively. We first define quantities from information theory in order to show a direct sum theorem for internal information.

▶ **Definition 17** (Entropy and conditional entropy)**.** *The* entropy *of a random variable X is defined as $H(X) := \sum_x p(x) \log \frac{1}{p(x)}$, where $p(x) = \mathbf{Pr}[X = x]$. The* conditional entropy *of X with respect to a random variable Y is defined as $H(X|Y) = \mathop{\mathbb{E}}_y [H(X|Y = y)]$.*

▶ **Definition 18** (Mutual information and conditional mutual information)**.** *The* mutual information *between random variables A and B is defined as $I(A;B) = H(A) - H(A|B) = H(B) - H(B|A)$. The* conditional mutual information *between A and B conditioned on a random variable C is defined as $I(A;B|C) = H(A|C) - H(A|B,C)$.*

▶ **Definition 19** (Communication Complexity). *Given a distributional one-way communication problem on inputs in $\mathcal{X} \times \mathcal{Y}$ distributed according to $\mathcal{D}$, and a one-way protocol Π, let $\Pi(X)$ denote the message of Alice under input X. Then the* communication complexity *of Π is the maximum length of $\Pi(X)$, over all inputs X and private randomness r:*

$$\mathsf{CC}(\Pi) := \max_{X \in supp(\mathcal{D}),\ private\ randomness\ r} |\Pi_r(X)|,$$

where $\Pi_r(X)$ is the length of Alice's message on input X with private randomness r.

▶ **Definition 20** (Internal Information Cost). *Given a distributional one-way communication problem on inputs in $\mathcal{X} \times \mathcal{Y}$ distributed according to $\mathcal{D}$, and a one-way protocol Π, let $\Pi(X)$ denote the message of Alice under input X. Then we define the* internal information cost *of Π with respect to some other distribution $\mathcal{D}'$ to be $\mathsf{IC}_{\mathcal{D}'}(\Pi) := I_{\mathcal{D}'}(\Pi(X); X|Y, R)$, where $\Pi(X)$ denotes the message sent from Alice to Bob in the protocol Π on input X. Here R is the public randomness.*

Here $\mathcal{D}$ is the correctness distribution for the protocol Π and $\mathcal{D}'$ is a distribution for measuring information.

The following well-known theorem (e.g., Lemma 3.14 in [11]) shows that the communication complexity of any protocol is at least the (internal) information cost of the protocol.

▶ **Theorem 21.** *Given a distributional one-way communication problem on inputs in $\mathcal{X} \times \mathcal{Y}$ distributed according to $\mathcal{D}$, and a one-way protocol Π, let $\Pi(X)$ denote the message of Alice under input X. Then $supp(\mathcal{D}') \subseteq supp(\mathcal{D})$ implies $\mathsf{CC}(\Pi) \geq \mathsf{IC}_{\mathcal{D}'}(\Pi)$.*

Note that we include the condition $\mathrm{supp}(\mathcal{D}') \subseteq \mathrm{supp}(\mathcal{D})$ since our distributional communication problem is defined as the maximum message length over inputs in the support of $\mathcal{D}$, and thus $I_{\mathcal{D}'}(\Pi(X); X|Y) \leq H(\Pi(X)|Y) \leq H(\Pi(X)) \leq \mathbf{E}_{X \sim \mathcal{D}'}|\Pi(X)|$, where $|\Pi(X)|$ is the length of Alice's message on random input X, which in turn is at most $\max_{x \in \mathrm{supp}(\mathcal{D}'),\ \text{private randomness } r} |\Pi_r(x)| \leq \max_{x \in \mathrm{supp}(\mathcal{D}),\ \text{private randomness } r} |\Pi_r(x)| = \mathsf{CC}(\Pi)$. Here $|\Pi_r(x)|$ denotes the length of Alice's message on input x with private randomness r.

The core of our proof is to show that any protocol that succeeds in solving the p-Noisy PBHH with high constant probability has high internal information cost. This statement is then reduced to a statement about the information cost of a distributional version of the standard BHH problem, through a method similar to that of [9].

For completeness we state this distributional version of BHH:

▶ **Definition 22.** *For an* even *integer n, and integer $t \geq 2$, Boolean Hidden t-Hypermatching (BHH) is a one-way two-party communication problem:*

- *Alice gets $X \in \{0,1\}^n$ uniformly at random.*
- *Bob gets M, a perfect t-hypermatching of $[n]$ (that is, n/t hyperedges) uniformly at random.*
- *Bob also receives edge labels $W \in \{0,1\}^{n/t}$. With probability $1/2$, we are in the* **YES** *case, and Bob's edge labels satisfy $MX = W$. With probability $1/2$, we are in the* **NO** *case and Bob's edge labels satisfy $MX \oplus W = 1^{n/t}$.*
- *Output: Bob must determine whether the communication game is in the* **YES** *or* **NO** *case.*

We call this input distribution of BHH $\mathcal{D}$, and call the distributions when conditioning on the **YES** *and* **NO** *cases $\mathcal{D}_{\mathbf{YES}}$ and $\mathcal{D}_{\mathbf{NO}}$, respectively.*

▶ **Lemma 23** (Lemma 3.4 in [28] with $t = 1$). *Given a distributional one-way communication problem on inputs in $\mathcal{X} \times \mathcal{Y}$ distributed according to $\mathcal{D}'$. Suppose there exists a one-way communication protocol Π' succeeding with probability $1 - \delta$ with $IC_{\mathcal{D}'}(\Pi') \leq c$. Then for any $\epsilon > 0$ there exists some other one-way communication protocol Π with*

$$CC(\Pi) \leq \frac{c+5}{\epsilon} + O\left(\log \frac{1}{\epsilon}\right),$$

and succeeding with probability $1 - \delta - 6\epsilon$.

▶ **Corollary 24.** *Any randomized protocol Π that succeeds with probability at least $\frac{2}{3}$ over the distribution $\mathcal{D}$ for the Boolean Hidden Hypermatching problem with hyperedges that contain t vertices requires internal information cost $\Omega(n^{1-1/t})$ when measured on the D distribution.*

Proof. It is known, e.g. [4], that the communication complexity of this distributional version of BHH is $\Omega(n^{1-1/t})$ for any protocol that succeeds with probability bounded away from $1/2$. However, any protocol with internal information cost $o(n^{1-1/t})$, when combined with Lemma 23 for a small constant ϵ, would result in a better protocol for BHH; a contradiction. ◀

▶ **Corollary 25** (Conditional information lower bounds for BHH). *Any randomized protocol Π that succeeds with probability at least $\frac{2}{3}$ over the distribution $\mathcal{D}$ for the Boolean Hidden Hypermatching problem with hyperedges that contain t vertices requires internal information cost $\Omega(n^{1-1/t})$ when measured on the $\mathcal{D}_{\mathbf{YES}}$ distribution. That is:*

$$I_{\mathcal{D}_{\mathbf{YES}}}(X; \Pi(X) | M, W) = \Omega(n^{1-1/t}).$$

Proof. By Corollary 24 we know that

$$I_{\mathcal{D}}(X; \Pi(X) | M, W) = \Omega(n^{1-1/t})$$

We decompose this into information complexity with respect to the $\mathcal{D}_{\mathbf{YES}}$ and $\mathcal{D}_{\mathbf{NO}}$ distributions. Let b be the single bit denoting whether the input is in the **YES** or **NO** case. Since b has entropy 1, adding or removing the conditioning on b amounts to at most a change of 1 in the mutual information.

$$\begin{aligned} I_{\mathcal{D}}(X; \Pi(X) | M, W) &\leq 1 + I_{\mathcal{D}}(X; \Pi(X) | M, W, b) \\ &= 1 + \tfrac{1}{2} I_{\mathcal{D}_{\mathbf{YES}}}(X; \Pi(X) | M, W, b) + \tfrac{1}{2} I_{\mathcal{D}_{\mathbf{NO}}}(X; \Pi(X) | M, W, b) \end{aligned}$$

Note that X (and consequently $\Pi(X)$) follow the same distribution in both the **YES** and **NO** cases. Furthermore, note that BHH flips each bit of the edge labels deterministically in the **NO** case. Therefore, in both the **YES** and **NO** cases, revealing b and W amounts to revealing the true edge parities. Thus, two terms ($I_{\mathcal{D}_{\mathbf{YES}}}$ and $I_{\mathcal{D}_{\mathbf{NO}}}$) are equal:

$$\begin{aligned} I_{\mathcal{D}}(X; \Pi(X) | M, W) &\leq 1 + I_{\mathcal{D}_{\mathbf{YES}}}(X; \Pi(X) | M, W, b) \\ &\leq 2 + I_{\mathcal{D}_{\mathbf{YES}}}(X; \Pi(X) | M, W). \end{aligned}$$

The statement of the corollary then follows. ◀

We define one more distributional one-way communication problem to bridge the gap between BHH and p-Noisy PBHH. It consists of $1/(2p)$ instances of BHH, of which exactly one or none are in the **NO** case. Furthermore all the instances of BHH are of variable size:

▶ **Definition 26.** *Let $p \in \left(\frac{t}{2n}, \frac{1}{2}\right]$, where we assume for simplicity that $Q := 1/2p$ is an integer. For $t \geq 2$ integer, the Variable Size, Q-Copy Boolean Hidden t-Hypermatching ($\overline{VBHH}$) is a one-way two-party communication problem:*

- *$S_1, \ldots, S_Q$ are drawn independently from $|\mathcal{Z}_p^{n/t}|$ (recall Definition 15), and r is drawn independently, uniformly from $[Q]$.*
- *In the* **YES** *case, Alice and Bob get Q independent copies of BHH distributed according to $\mathcal{D}_{\mathbf{YES}}$, where the i^{th} copy is on $t \cdot S_i$ vertices.*
- *In the* **NO** *case, Alice and Bob again get Q independent copies of BHH, where the i^{th} copy has size $t \cdot S_i$. All copies are distributed according to $\mathcal{D}_{\mathbf{YES}}$, except the r^{th} copy, which is distributed according to $\mathcal{D}_{\mathbf{NO}}$.*
- *Output: Bob must determine whether the communication game is in the* **YES** *or* **NO** *case.*

We call this input distribution of $\overline{VBHH}$ $\overline{\mathcal{D}}$, and call the distributions when conditioning on the **YES** *and* **NO** *cases $\overline{\mathcal{D}}_{\mathbf{YES}}$ and $\overline{\mathcal{D}}_{\mathbf{NO}}$ respectively.*

▶ **Theorem 27.** *Any randomized protocol Π that succeeds with probability at least $\frac{4}{5}$ over the distribution $\overline{\mathcal{D}}$ for the $\overline{VBHH}$ parameters n, $2 \leq t \leq n/100$ and $p \in \left(\frac{t}{2n}, \frac{1}{2}\right]$, has internal information cost $\Omega(n^{1-1/t}p^{-1/t})$ when measured on the $\overline{\mathcal{D}}_{\mathbf{YES}}$ distribution. That is:*

$$I_{\overline{\mathcal{D}}_{\mathbf{YES}}}(X; \Pi(X)|M, W) = \Omega(n^{1-1/t}p^{-1/t}).$$

Finally, we lower bound the communication complexity of p-Noisy PBHH, by a reduction from $\overline{\text{VBHH}}$.

▶ **Theorem 28.** *For $t \leq n/10$, and $p \in \left(\frac{t}{2n}, \frac{1}{2}\right]$, the communication complexity of p-Noisy Perfect Boolean Hidden Hypermatching with success probability $5/6$ is $\Omega(n^{1-1/t}p^{-1/t})$ bits.*

4 Applications

In this section, we give a number of applications for the p-Noisy BHM and p-Noisy BHH problems. Through a reduction from p-Noisy BHH, we first show hardness of approximation for max cut in the streaming model on graphs whose connected components have bounded size, which is a significant obstacle for reductions to BHM. Unlike reductions from BHM, our reduction from p-Noisy BHM can still show nearly linear lower bounds in this setting.

We then show hardness of approximation for maximum matching in the streaming model better than $\Omega(\sqrt{n})$ on graphs whose connected components have bounded size. Again our reduction displays flexibility beyond what is offered by either BHM or BHH for both parametrization of component size and tradeoffs between approximation guarantee and space complexity.

Finally, we show hardness of approximation for maximum acyclic subgraph. [26] show a $\Omega(\sqrt{n})$ lower bound for $\frac{8}{7}$-approximation, but the reduction does not readily translate into a stronger lower bound from BHH, due to the hyperedge structure of BHH. We use our p-Noisy BHM problem to give a fine-grained lower bound that provides tradeoffs between the approximation guarantee and the required space.

4.1 MAX-CUT

Recall the following definition of the MAX-CUT problem.

▶ **Problem 29** (MAX-CUT). *Given an unweighted graph $G = (V, E)$, the goal is to output the maximum of the number of edges of G that cross a bipartition, over all bipartitions of V, i.e., $\max_{P \cup Q = V, P \cap Q = \emptyset} |E \cap (P \times Q)|$.*

To show hardness of approximation of MAX-CUT in the streaming model, where the edges of the underlying graph G arrive sequentially, we use a reduction similar to [34], who gave a reduction from BHM that creates a connected component with 8 edges for each edge m_i in the input matching M from BHM. The key property of the reduction of [34] is that for $(Mx)_i \oplus w_i = 0$, the connected component corresponding to m_i is bipartite and its induced max cut has size 8, but for $(Mx)_i \oplus w_i = 1$, the connected component is not bipartite and its induced max cut has size at most 7 (see Figure 1). Therefore, the max cut for $Mx \oplus w = 0^n$ has size $8n$ and the max cut for $Mx \oplus w = 1^n$ has size at most $7n$, which gives the desired separation for a constant factor approximation algorithm.

We instead reduce from Noisy Boolean Hidden Hypermatching. Suppose Alice is given a binary vector x of length $n = 2kt$ and Bob is a given a hypermatching M of size $n/2t = k$ on n vertices, where each edge contains t vertices, so that $\alpha = \frac{1}{2}$. Bob also receives a vector w of length k, generated according to the **YES** or **NO** case of the p-noisy BHH (see Definition 14). To distinguish between the two cases:

- Alice creates the four vertices a_i, b_i, c_i, and d_i for each $i \in [n]$ corresponding to a coordinate of x.
- If $x_i = 0$, then Alice adds the edges (a_i, b_i), (c_i, d_i), and (a_i, d_i), but if $x_i = 1$, then Alice instead adds the edges (a_i, b_i), (c_i, d_i), and (a_i, c_i).
- For each hyperedge $m_i = (j_{i,1}, \ldots, j_{i,t})$ of M, with $j_{i,s} \leq j_{i,s+1}$, if $w_i = 0$ Bob adds the edges $(d_{j_{i,s}}, a_{j_{i,s+1}})$ for $s \in [t-1]$, and the edge $(d_{j_{i,t}}, a_{j_{i,1}})$. Otherwise if $w_i = 1$, then Bob instead adds the edges $(d_{j_{i,s}}, a_{j_{i,s+1}})$ for $s \in [t-1]$ and the edge $(d_{j_{i,t}}, b_{j_{i,1}})$.

By design, the connected component of the graph corresponding to m_i is bipartite if and only if $(Mx)_i \oplus w_i = 0$. Hence, the max cut has size $4t$ if $(Mx)_i \oplus w_i = 0$ and size at most $4t - 1$ otherwise, i.e., when $(Mx)_i \oplus w_i = 1$.

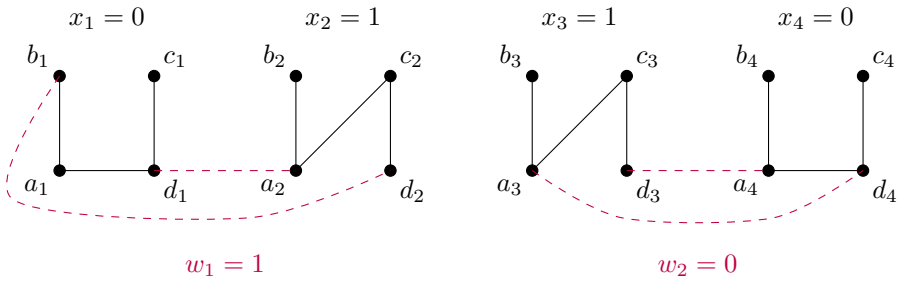


Figure 1 Example of reduction from Noisy Boolean Hidden Matching (Noisy BHH with $t = 2$) to MAX-CUT. Solid lines added by Alice, dashed purple lines added by Bob. Here we have $m_1 = (1, 2)$ and $m_2 = (3, 4)$. Note that $x_1 \oplus x_2 \oplus w_1 = 0$ and has a cycle of even length while $x_3 \oplus x_4 \oplus w_2 = 1$ has a cycle of odd length.

4.2 MAX-MATCHING

Recall the following definition of the MAX-MATCHING problem.

▶ **Problem 30** (Maximum Matching). *Given an unweighted graph $G = (V, E)$, the goal is to output the maximum size of a matching, that is a set of disjoint edges.*

To show hardness of approximation of MAX-MATCHING in the streaming model, we use a reduction similar to [12]. We use a reduction from noisy BHH. In the reduction, each matching hyperedge m_i corresponds to a gadget consisting of two connected components of

size $O(t)$. The key observation is that if $m_i x \oplus w_i = 0$, then the two components are of even size, and can be perfectly matched, but if $m_i x \oplus w_i = 1$, then the two components are of odd size, and cannot be perfectly matched.

We reduce from Noisy Boolean Hidden Hypermatching with $\alpha = 1/2$. Suppose Alice is given a binary vector x of length n and Bob is a given a hypermatching M of size $n/2t$ on the vertex-set $[n]$, where each edge contains t vertices. Bob also receives a vector w of length $n/2t$, generated according to the **YES** or **NO** case of the p-noisy BHH (see Definition 14). The protocol to distinguish between the two cases slightly differs based on the parity of t, but the core idea is the same. We begin by considering odd t.

- Alice creates the four vertices a_i, b_i, c_i, and d_i for each $i \in [n]$ corresponding to a coordinate of x.
- If $x_i = 0$, then Alice adds the single edges (a_i, b_i), but if $x_i = 1$, then Alice instead adds the edge (c_i, d_i).
- For each hyperedge $m_i = (j_{i,1}, \ldots, j_{i,t})$ of M, Bob adds a single vertex e_i. Then, if $w_i = 0$ Bob adds a clique between $(b_{j_i,1}, b_{j_{i,2}}, \ldots, b_{j_{i,t}})$ and another clique between $(d_{j_i,1}, d_{j_{i,2}}, \ldots, d_{j_{i,t}}, e_i)$. If $w_i = 1$, Bob adds the same two cliques, but moving e_i from the second clique to the first. Formally Bob adds the cliques $(b_{j_i,1}, b_{j_{i,2}}, \ldots, b_{j_{i,t}}, e_i)$ and $(d_{j_i,1}, d_{j_{i,2}}, \ldots, d_{j_{i,t}})$.

If t is even, the protocol is very similar, but we state it here for completeness:

- Alice does the same thing as in the previous case: she creates the four vertices a_i, b_i, c_i, and d_i for each $i \in [n]$ corresponding to a coordinate of x.
- If $x_i = 0$, then Alice adds the single edges (a_i, b_i), but if $x_i = 1$, then Alice instead adds the edge (c_i, d_i).
- For each hyperedge $m_i = (j_{i,1}, \ldots, j_{i,t})$ of M, Bob adds two new vertices e_i and f_i. Then, if $w_i = 0$ Bob adds a clique between $(b_{j_i,1}, b_{j_{i,2}}, \ldots, b_{j_{i,t}})$ and Bob adds another clique between $(d_{j_i,1}, d_{j_{i,2}}, \ldots, d_{j_{i,t}}, e_i, f_i)$. If $w_i = 1$, Bob adds the same two cliques, but moving e_i from the second clique to the first. Formally Bob adds the cliques $(b_{j_i,1}, b_{j_{i,2}}, \ldots, b_{j_{i,t}}, e_i)$ and $(d_{j_i,1}, d_{j_{i,2}}, \ldots, d_{j_{i,t}}, f_i)$.

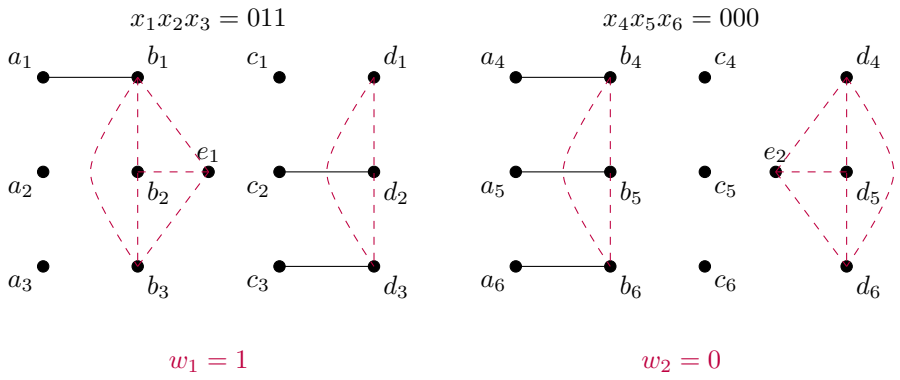


Figure 2 An illustration of the graph construction for the maximum matching reduction for $t = 3$. Solid lines added by Alice, dashed lines added by Bob. On the left side, we see the components corresponding to $m_1 = \{1, 2, 3\}$. Since $m_1 x \oplus w_1 = 1$ this subgraph contributes 4 to the maximum matching. On the right side, we see the components corresponding to $m_2 = \{4, 5, 6\}$. Since $m_2 x \oplus w_2 = 0$ this subgraph contributes 5 to the maximum matching.

4.3 Maximum Acyclic Subgraph

In this section, we study the hardness of approximation for the maximum acyclic subgraph on insertion-only streams.

▶ **Problem 31** (Maximum acyclic subgraph). *Given a directed graph $G = (V, E)$, the goal is to output the size of the largest acyclic subgraph of G, where the size of a graph is defined to be the number of edges in it.*

To show hardness of approximation of maximum acyclic subgraph in the streaming model, where the directed edges of G arrive sequentially, we consider the reduction of [26], who created a subgraph with 8 edges for each edge m_i in the input matching M from BHM. The key property of the reduction is that $(Mx)_i \oplus w_i = 1$ corresponds to a subgraph with no cycles but $(Mx)_i \oplus w_i = 0$ corresponds to a subgraph with a cycle (see Figure 3). Hence, the former case allows 8 edges in its maximum acyclic subgraph while the latter case only allows for 7 edges. We use the same reduction from p-Noisy BHM with $\alpha = 1/2$, so that there are $n/4$ matching edges.

In particular, suppose Alice is given a string $x \in \{0,1\}^{2n}$ and Bob is given a matching of $[2n]$ of size αn with $\alpha = \frac{1}{2}$. To distinguish between the two cases:

- Alice and Bob consider a graph on $4(2n) = 8n$ vertices and associate four vertices a_i, b_i, c_i, and d_i to each $i \in [2n]$, i.e., through a predetermined ordering of the vertices.
- If $x_i = 0$, then Alice creates the directed edges (a_i, b_i) and (d_i, c_i), where we use the convention that (v_1, v_2) represents the directed edge $v_1 \to v_2$. Otherwise if $x_i = 1$, then Alice adds the directed edges (b_i, a_i) and (c_i, d_i).
- If $w_i = 0$, Bob adds the four directed edges (b_{y_i}, a_{z_i}), (b_{z_i}, a_{y_i}), (d_{y_i}, c_{z_i}), and (d_{z_i}, c_{y_i}) for each matching edge $m_i = (y_i, z_i)$ of M from the p-Noisy Boolean Hidden Matching problem. Otherwise if $w_i = 1$, then Bob adds the four directed edges (b_{y_i}, b_{z_i}), (a_{z_i}, a_{y_i}), (d_{y_i}, d_{z_i}), and (c_{z_i}, c_{y_i}).

By design, the subgraph corresponding to m_i has eight edges, but consists of exactly one cycle when $x_{y_i} \oplus x_{z_i} \oplus w_i = 0$ and zero cycles when $x_{y_i} \oplus x_{z_i} \oplus w_i = 1$. Finally, any unmmatched subgraph contributes four edges due to Alice's construction.

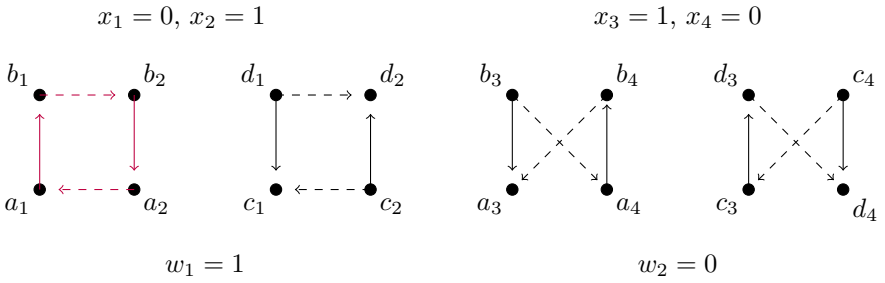
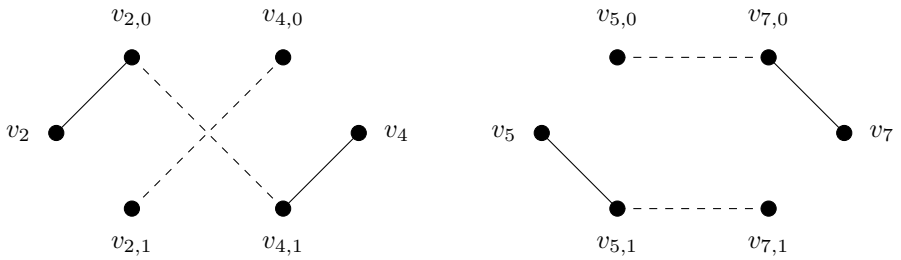


Figure 3 Example of reduction from (Noisy) Boolean Hidden Matching to Maximum Acyclic Subgraph. Solid lines added by Alice, dashed lines added by Bob, and purple lines represent a cycle. Here we have $m_1 = (1, 2)$ and $m_2 = (3, 4)$. Note that $x_1 \oplus x_2 \oplus w_1 = 0$ has a cycle while $x_3 \oplus x_4 \oplus w_2 = 1$ has no cycles.

4.4 Streaming Binary Tree Classification

In this section, we study the hardness of tree classification on insertion-only streams. In this setting, a stream of edge-insertions in an underlying graph with n vertices arrive sequentially and the task is to classify whether the resulting graph is isomorphic to a complete binary tree on n vertices, or δ-far from one.



(a) Suppose $(m_4, m_4') = (2, 3)$, $x_2 = 0$, $x_3 = 1$, and $w_4 = 0$. Then $x_2 \oplus x_3 \oplus w_4 = 1$ and the construction forms a path of length one and a path of length three. Solid lines added by Alice, dashed lines added by Bob.

(b) Suppose $(m_6, m_6') = (5, 7)$, $x_5 = 1$, $x_7 = 0$, and $w_6 = 1$. Then $x_5 \oplus x_7 \oplus w_6 = 0$ and the construction forms two paths of length two. Solid lines added by Alice, dashed lines added by Bob.

Figure 4 Construction of graph from instance of Boolean Hidden Matching. Observe that in the case $Mx \oplus w = 1^n$, $v_{5,1}$ and $v_{7,0}$ can be embedded into the bottom layer of a complete binary tree to form another complete binary tree, while in the $Mx \oplus w = 0^n$ case, $v_{2,0}$ and $v_{4,1}$ will induce a cycle in the graph.

▶ **Problem 32** (Graph Classification). *Given a stream for the set of edges between n vertices, determine whether the resulting graph induced by the stream is isomorphic to a complete binary tree on n vertices. Here we assume that the number of vertices is $2^k - 1$ for some integer k.*

[19] used the following construction to show hardness of approximation for maximum matching size in the streaming model. Given an instance of Boolean Hidden Matching with an input vector $x \in \{0,1\}^{2n}$, a binary string $w \in \{0,1\}^n$, and a matching $M = \{(m_1, m_1'), (m_2, m_2'), \ldots, (m_n, m_n')\}$ of $[2n]$, let G be a graph with $6n$ vertices. Each bit x_i is associated with vertices $v_i, v_{i,0}, v_{i,1}$ in G. Alice connects vertex v_i to v_{i,x_i} in G, e.g., if $v_i = 0$ then Alice connects v_i to $v_{i,0}$ and if $v_i = 1$ then Alice connects v_i to $v_{i,1}$. If $w_i = 0$, then Bob creates an edge between $v_{m_i,0}$ and $v_{m_i',1}$, as well as an edge between $v_{m_i,1}$ and $v_{m_i',0}$. Otherwise if $w_i = 1$, then Bob creates an edge between $v_{m_i,0}$ and $v_{m_i',0}$, as well as an edge between $v_{m_i,1}$ and $v_{m_i',1}$. Under this construction, if $x_{m_i} \oplus x_{m_i'} \oplus w_i = 1$, then the six vertices $v_{m_i}, v_{m_i,0}, v_{m_i,1}, v_{m_i'}, v_{m_i',0}, v_{m_i',1}$ form a path of length one and a path of length three. Otherwise, if $x_{m_i} \oplus x_{m_i'} \oplus w_i = 0$, then the six vertices form two paths of length two. We call this the EHLMO construction. See Figure 4 for an illustration.

To show hardness of classifying whether an underlying graph is a complete binary tree, we use the EHLMO construction to embed an instance of p-Noisy BHM of size $2n$ into the bottom layer of a binary tree with $4n - 1$ total nodes. For each $i \in [2n]$, the case $(Mx)_i \oplus w_i = 0$ creates two paths of length two. We use these two paths of length two to extend the binary tree to an additional layer at two different nodes. On the other hand, the case $(Mx)_i \oplus w_i = 1$ creates a path of length one and a path of length three. By using the same construction, the case $(Mx)_i \oplus w_i = 1$ thus results in a non-root node in the tree having degree two; hence the resulting graph is a complete binary tree if and only if $Mx \oplus w = 0^n$.

Formally, let n be a power of 2 and let T be a complete binary tree with $2n$ leaves. Consider the EHLMO construction to embed an instance of the above Noisy Boolean Hidden Matching problem with noise $p = \frac{4}{n}$ into the bottom layer of a tree, so that in the case where $Mx \oplus w = 0^n$, the resulting graph is a complete binary tree, while in the case $Mx \oplus w \neq 0^n$, the resulting graph contains a cycle. More formally:

- Alice creates a complete $6n$ vertices $v_i, v_{i,0}, v_{i,1}$ for each $i \in [2n]$.
- Alice creates a complete binary tree with leaves v_{i,x_i} for $i \in [2n]$ (and new non-leaf vertices, distinct from any previously created).

- Alice also creates an edge between vertices v_i and v_{i,x_i} for each $i \in [2n]$.
- For each edge $m_i = (y_i, z_i)$ of the matching M with $i \in [n]$, Bob creates an edge between $v_{y_i,0}$ and $v_{z_i,w_i \oplus 1}$ as well as an edge between $v_{y_i,1}$ and v_{z_i,w_i}.

We note that Theorem 9 also follows by a reduction from the standard INDEX communication problem. The following more general Theorem 11, however, does not. It follows by the same reduction we use for Theorem 11 and follows from a nearly identical proof, with the only change being that the variable p is set to ϵ in the reduction from p-Noisy BHM.

References

1 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.

2 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.

3 Sepehr Assadi. Personal communication, 2021.

4 Sepehr Assadi, Sanjeev Khanna, and Yang Li. On estimating maximum matching size in graph streams. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1723–1742, 2017.

5 Sepehr Assadi, Sanjeev Khanna, Yang Li, and Grigory Yaroslavtsev. Maximum matchings in dynamic graph streams and the simultaneous communication model. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1345–1364, 2016.

6 Sepehr Assadi, Gillat Kol, Raghuvansh R. Saxena, and Huacheng Yu. Multi-pass graph streaming lower bounds for cycle counting, max-cut, matching size, and other problems. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS*, pages 354–364, 2020.

7 Sepehr Assadi and Vishvajeet N. Graph streaming lower bounds for parameter estimation and property testing via a streaming XOR lemma. In *STOC: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 612–625, 2021.

8 Ziv Bar-Yossef, T. S. Jayram, and Iordanis Kerenidis. Exponential separation of quantum and classical one-way communication complexity. *SIAM J. Comput.*, 38(1):366–384, 2008.

9 Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.*, 68(4):702–732, 2004.

10 Sayan Bhattacharya, Monika Henzinger, Danupon Nanongkai, and Charalampos E. Tsourakakis. Space- and time-efficient algorithm for maintaining dense subgraphs on one-pass dynamic streams. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC*, pages 173–182, 2015.

11 Mark Braverman and Anup Rao. Information equals amortized communication. *IEEE Trans. Inf. Theory*, 60(10):6058–6069, 2014.

12 Marc Bury, Elena Grigorescu, Andrew McGregor, Morteza Monemizadeh, Chris Schwiegelshohn, Sofya Vorotnikova, and Samson Zhou. Structural results on matching estimation with applications to streaming. *Algorithmica*, 81(1):367–392, 2019.

13 Rajesh Chitnis, Graham Cormode, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Andrew McGregor, Morteza Monemizadeh, and Sofya Vorotnikova. Kernelization via sampling with applications to finding matchings and related problems in dynamic graph streams. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1326–1344, 2016.

14 Rajesh 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.

15 Chi-Ning Chou, Alexander Golovnev, Madhu Sudan, and Santhoshini Velusamy. Approximability of all boolean csps in the dynamic streaming setting. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS*, 2021. (to appear).

16 Chi-Ning Chou, Sasha Golovnev, and Santhoshini Velusamy. Optimal streaming approximations for all boolean max-2csps and max-ksat. In *IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 330–341, 2020.

17 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.

18 Artur Czumaj, Hendrik Fichtenberger, Pan Peng, and Christian Sohler. Testable properties in general graphs and random order streaming. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, 2020.

19 Hossein Esfandiari, MohammadTaghi Hajiaghayi, Vahid Liaghat, Morteza Monemizadeh, and Krzysztof Onak. Streaming algorithms for estimating the matching size in planar graphs and beyond. *ACM Trans. Algorithms*, 14(4):48:1–48:23, 2018.

20 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.

21 Joan Feigenbaum, Sampath Kannan, Andrew McGregor, Siddharth Suri, and Jian Zhang. Graph distances in the data-stream model. *SIAM J. Comput.*, 38(5):1709–1727, 2008.

22 Dmitry Gavinsky, Julia Kempe, and Ronald de Wolf. Exponential separation of quantum and classical one-way communication complexity for a boolean function. *CoRR*, abs/quant-ph/0607174, 2006.

23 Dmitry Gavinsky, Julia Kempe, Iordanis Kerenidis, Ran Raz, and Ronald de Wolf. Exponential separation for one-way quantum communication complexity, with applications to cryptography. *SIAM J. Comput.*, 38(5):1695–1708, 2008.

24 Sudipto Guha, Andrew McGregor, and David Tench. Vertex and hyperedge connectivity in dynamic graph streams. In *Proceedings of the 34th ACM Symposium on Principles of Database Systems, PODS*, pages 241–247, 2015.

25 Venkatesan Guruswami and Runzhou Tao. Streaming hardness of unique games. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, pages 5:1–5:12, 2019.

26 Venkatesan Guruswami, Ameya Velingker, and Santhoshini Velusamy. Streaming complexity of approximating max 2csp and max acyclic subgraph. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, volume 81, pages 8:1–8:19, 2017.

27 Zengfeng Huang and Pan Peng. Dynamic graph stream algorithms in o(n) space. *Algorithmica*, 81(5):1965–1987, 2019.

28 Rahul Jain, Attila Pereszlényi, and Penghui Yao. A direct product theorem for bounded-round public-coin randomized communication complexity. *CoRR*, abs/1201.1666, 2012.

29 Jeff Kahn, Gil Kalai, and Nathan Linial. The influence of variables on boolean functions (extended abstract). In *29th Annual Symposium on Foundations of Computer Science*, pages 68–80. IEEE Computer Society, 1988.

30 John Kallaugher, Michael Kapralov, and Eric Price. The sketching complexity of graph and hypergraph counting. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 556–567, 2018. doi:10.1109/FOCS.2018.00059.

31 John Kallaugher and Eric Price. A hybrid sampling scheme for triangle counting. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1778–1797, 2017.

32 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.

33 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.

34 Michael Kapralov, Sanjeev Khanna, and Madhu Sudan. Streaming lower bounds for approximating MAX-CUT. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1263–1282, 2015.

35 Michael Kapralov, Sanjeev Khanna, Madhu Sudan, and Ameya Velingker. $(1 + \omega(1))$-approximation to MAX-CUT requires linear space. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1703–1722, 2017.

36 Michael Kapralov and Dmitry Krachun. An optimal space lower bound for approximating MAX-CUT. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 277–288, 2019. doi:10.1145/3313276.3316364.

37 Michael Kapralov, Yin Tat Lee, Cameron Musco, Christopher Musco, and Aaron Sidford. Single pass spectral sparsification in dynamic streams. *SIAM J. Comput.*, 46(1):456–477, 2017.

38 Michael Kapralov, Slobodan Mitrovic, Ashkan Norouzi-Fard, and Jakab Tardos. Space efficient approximation to maximum matching size from uniform edge samples. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1753–1772, 2020.

39 Michael Kapralov, Amulya Musipatla, Jakab Tardos, David P. Woodruff, and Samson Zhou. Noisy boolean hidden matching with applications. *CoRR*, abs/2107.02578, 2021.

40 Michael Kapralov and David P. Woodruff. Spanners and sparsifiers in dynamic streams. In *ACM Symposium on Principles of Distributed Computing, PODC*, pages 272–281, 2014.

41 Iordanis Kerenidis and Ran Raz. The one-way communication complexity of the boolean hidden matching problem. *CoRR*, abs/quant-ph/0607173, 2006.

42 Dmitry Kogan and Robert Krauthgamer. Sketching cuts in graphs and hypergraphs. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science, ITCS*, pages 367–376, 2015.

43 Yi Li and David P. Woodruff. On approximating functions of the singular values in a stream. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pages 726–739, 2016.

44 Morteza Monemizadeh, S. Muthukrishnan, Pan Peng, and Christian Sohler. Testable bounded degree graph properties are random order streamable. In *44th International Colloquium on Automata, Languages, and Programming, ICALP*, pages 131:1–131:14, 2017.

45 Jelani Nelson and Huacheng Yu. Optimal lower bounds for distributed and streaming spanning forest computation. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1844–1860, 2019.

46 Ami Paz and Gregory Schwartzman. A $(2 + \epsilon$-approximation for maximum weight matching in the semi-streaming model. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 2153–2161, 2017.

47 Xiaoming Sun and David P. Woodruff. Tight bounds for graph problems in insertion streams. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, pages 435–448, 2015.

48 Elad Verbin and Wei Yu. The streaming complexity of cycle counting, sorting by reversals, and other problems. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 11–25, 2011.