Electronic Colloquium on Computational Complexity, Report No. 17 (2009)

The Maximum Communication Complexity of Multi-Party Pointer Jumping

Joshua Brody *

Department of Computer Science, Dartmouth College
Hanover, NH 03755, USA

Abstract

We study the one-way number-on-the-forhead (NOF) communication complexity of the k-layer pointer jumping problem. Strong lower bounds for this problem would have important implications in circuit complexity. All of our results apply to myopic protocols (where players see only one layer ahead, but can still see arbitrarily far behind them.) Furthermore, our results apply to the maximum communication complexity, where a protocol is charged for the *maximum* communication sent by a single player rather than the *total* communication sent by all players.

Our main result is a lower bound of $n/2$ bits for deterministic protocols, independent of the number of players. We also provide a matching upper bound, as well as an $\Omega\left(n/k \log n\right)$ lower bound for randomised protocols, improving on the bounds of Chakrabarti [Cha07]. In the non-Boolean version of the problem, we give a lower bound of $n(\log^{(k-1)} n)(1 - o(1))$ bits, essentially matching the upper bound from Damm et al. [DJS98].

1 Introduction

Communication complexity has been an important technique in proving lower bounds in a wide variety of areas, including settings that do not involve communication. Specifically, communication complexity has been used to prove lower bounds on the depth of monotone circuits for undirected connectivity [KW88], time/space tradeoffs for cell probe data structures [Ajt88, Mil94], and lower bounds on space complexity in streaming algorithms [AMS99, GM07, CJP08].

We focus on the comunication complexity of the multi-party pointer jumping problem in the number-on-the-forhead model, introduced by Chandra, Furst, and Lipton [CFL83]. A series of works [Yao90, HG91, BT94] has shown that a strong lower bound for any explicit function f in this model would imply that $f \notin \mathsf{ACC}^0$. The pointer jumping problem is widely considered to be a good candidate for such a lower bound.

1.1 The Pointer Jumping Problem and Previous Results

There are a number of variants of the pointer jumping problem, all of which involve following a series of directed edges in a graph. We study two variants of the multiplayer pointer jumping variety: a Boolean version MPJ_k and a non-Boolean version $\widehat{\text{MPJ}}_k$. In these settings, there is a graph G_k^n, which has $k+1$ layers of vertices. Layer 0 contains a single vertex v_0. Each layer $1 \leq i \leq k-1$ contains n vertices. In the Boolean version, layer k contains two vertices labelled 0 and 1. In the non-Boolean version, layer k also contains n vertices. There are directed edges in G_k^n for each vertex in layer i to each vertex in layer

*Work supported in part by NSF grant CCF-0448277

ISSN 1433-8092

$i+1$. The input to the pointer jumping problem is a subgraph where each vertex (except those in layer k) has outdegree 1, and the goal is to output the unique vertex in layer k reachable from vertex v_0. The NOF communication version of MPJ_k and $\widehat{\mathrm{MPJ}}_k$ work as follows: there are k players $\mathrm{PLR}_1, \ldots, \mathrm{PLR}_k$. The set of edges from layer $i-1$ to layer i are written on PLR_i's forhead, and the players communicate in a fixed order $\mathrm{PLR}_1, \ldots, \mathrm{PLR}_k$. PLR_k's message is the output. Note that the order of the players is important: if the players speak in an order other than $\mathrm{PLR}_1, \mathrm{PLR}_2, \ldots, \mathrm{PLR}_k$, then an easy $O(\log n)$ protocol exists. As mentioned previously, proving lower bounds for this problem would have important consequences in circuit complexity. Specifically, showing a polynomial lower bound on communication for deterministic MPJ_k protocols for any $k = \omega(\mathrm{polylog}\, n)$ would show that $\mathrm{MPJ}_k \notin \mathsf{ACC}^0$. Consult the work of Beigel and Tarui [BT94] for more details.

There are a number of other variants to the pointer jumping problem. All of them operate by following pointers on a graph similar to the multi-party version. In the *bipartite* pointer jumping problem, denoted BPJ_k, the input is a bipartite graph with directed edges between each of the parts, going in both directions. Harvey [Har08] used lower bounds for BPJ_k to show lower bounds on the number of queries needed to solve the matroid intersection problem. The graph for the *tree* pointer jumping problem, denoted TPJ_k, is a d-ary tree, with $d = O(n^{1/k-1})$. Viola and Wigderson [VW07] show lower bounds of $\Omega(n^{1/k-1}/k^{O(k)})$ for randomized protocols for TPJ_k. Note that the input to TPJ_k can be seen as a restriction of the input to MPJ_k, so this lower bound applies to MPJ_k as well.

The remarkable $\Omega(n^{1/k-1}/k^{O(k)})$ bound of Viola and Wigderson is tight for TPJ_k for all constant k and is the best known lower bound for MPJ_k. Unfortunately, it says nothing when $k = \omega(\log n)$. There are several stronger lower bounds for MPJ_k in restricted settings. There are also two nontrivial *upper* bounds. In the non-Boolean case, the trivial protocol costs $O(n \log n)$ bits and has PLR_1 sending PLR_2 the input on his forhead, giving him all the input and allowing him to output the answer. Damm, Junka, and Sgall [DJS98] give a deterministic protocol for $\widehat{\mathrm{MPJ}}_k$ which has cost $O(n \log^{(k)} n)$ for $k \leq \log^* n$ and $O(n)$ for $k > \log^* n$.[1] Their protocol is particularly interesting, because it is restricted in two different ways. Firstly, players do not see the layers $1, \ldots, i-1$ "behind" them as they normally would. Instead, they see only the result of following the pointers up to layer i. Damm et al. call this a *conservative* protocol and give a deterministic lower bound for such protocols that matches their upper bound up to a constant factor. Secondly, the players in the protocol of Damm et al. are restricted in what inputs they see "ahead" of them: instead of seeing layers $i+1, \ldots, k$, PLR_i sees only layer $i+1$. Such a protocol is called *myopic*. Gronemeier [Gro06] coined this term and gave a $\Omega(n^{(1-\epsilon)/k} \log n)$ lower bound for ϵ-error protocols. Chakrabarti improved this lower bound to $\Omega(n/k)$ and proved a lower bound of $\Omega(n \log^{(k-1)} n)$ for myopic $\widehat{\mathrm{MPJ}}_k$ protocols. Both bounds apply to randomsed protocols. Chakrabarti also gives lower bounds of $\Omega(n/k^2)$ and $\Omega(n \log^{(k-1)} n)$ for randomized conservative protocols for MPJ_k and $\widehat{\mathrm{MPJ}}_k$ respectively.

For MPJ_k, Brody and Chakrabarti [BC08] give a deterministic protocol for MPJ_k with cost $O\left(n(k \log\log n/\log n)^{1-1/(k-1)}\right)$, which disproved a long-standing conjecture that essentially nothing nontrivial could be done for MPJ_k protocols. This is currently the only nontrivial protocol known for MPJ_k. Their protocol showed that MPJ_k is a deeper problem than origionally expected, and its communication complexity, even in the deterministic setting, remains an open and vexing problem. Improving either the upper or lower bounds remains an interesting and difficult task.

1.2 Our Results

The protocol of Damm et al. is both myopic and conservative, but holds only for $\widehat{\mathrm{MPJ}}_k$. The MPJ_k protocol of Brody and Chakrabarti is neither. Our main result shows that there are no nontrivial myopic protocols for MPJ_k. Specifically, we have

[1] We use $\log^{(k)}$ to denote the kth iterated logarithm of n.

Theorem 1. *In any deterministic myopic protocol for* MPJ_k*, some player* PLR_j *must communicate at least* $n/2$ *bits.*

Using this result, we provide an exact bound on the *total* communication cost of myopic MPJ_k protocols.

Corollary 2. *A deterministic myopic protocol for* MPJ_k *must communicate at least* n *bits in total.*

This shows that the best myopic MPJ_k protocol is the trivial one where PLR_{k-1} sends PLR_k the last layer of input, and other players communicate nothing. A closer inspection of the proof of Theorem 1 shows that there exists a decreasing function $\phi : \mathbb{Z}^+ \to \mathbb{R}^+$, with $\lim_{k\to\infty} \phi(k) = 1/2$, such that in any deterministic protocol for MPJ_k, some player must communicate at least $\phi(k)n$ bits. Our next result shows that this lower bound on the maximum communication of myopic protocols is essentially tight.

Theorem 3. *For all* $k \geq 3$*, there exists a deterministic myopic protocol for* MPJ_k *in which each player sends* $(1 + o(1))\phi(k)n$ *bits.*

Our technique uses a round elimination lemma on a generalized version of MPJ_k in which there are $m \leq n$ vertices in the first layer of the graph. This method can also be applied to $\widehat{\text{MPJ}}_k$ protocols. Recall that Damm et al. gave a deterministic myopic protocol for $\widehat{\text{MPJ}}_k$ where each player sends at most $n \log^{(k-1)} n$ bits. Our technique gives a lower bound that nearly matches this.

Theorem 4. *In any deterministic myopic protocol for* $\widehat{\text{MPJ}}_k$*, some player must communicate at least* $n(\log^{(k-1)} n - \log^{(k)} n)$ *bits.*

Finally, we give a randomized bound on the maximum communication of randomized myopic MPJ_k protocols. Chakrabarti gave a lower bound of $\Omega(n/k)$ on the *total* communication of randomized MPJ_k protocols. This immediately yields a lower bound of $\Omega(n/k^2)$ on the *maximum* communication. We give a similar but incomparable result.

Theorem 5. *In any randomized myopic protocol for* MPJ_k*, some player must communicate at least* $\Omega(n/k \log n)$ *bits.*

While this improves on the bound of Chakrabarti only for $k \geq \log n$ players, we emphasize that this is precisely the setting which would yield lower bounds in circuit complexity.

1.3 Organization

The rest of the paper is organized as follows. In Section 2 we introduce notation and formally define the pointer jumping problem. In Section 3 we prove Theorems 1 and 4 and Corollary 2. We prove Theorem 3 in Section 4 and Theorem 5 in Section 5.

2 Preliminaties and Notation

For the rest of the paper, "protocols" will be assumed to be deterministic one-way NOF protocols unless otherwise qualified. Let $\mathcal{P}$ be a k-player protocol in which player i's message has length ℓ_i. Most of our results concern the *maximum* communication of a protocol. We define $\text{cost}(\mathcal{P}) := \max_{1\leq i\leq k} \ell_i$. A γ-bit protocol is a protocol $\mathcal{P}$ with $\text{cost}(\mathcal{P}) = \gamma$. We also define $\text{tcost}(\mathcal{P}) := \ell_1 + \ldots + \ell_k$ to be the *total* communication cost of a protocol.

We now formally define the problems $\text{MPJ}_{m,k}$ and $\widehat{\text{MPJ}}_{m,k}$ in a recursive fashion. We define $\text{MPJ}_{m,2} : [m] \times \{0,1\}^m \to \{0,1\}$ as $\text{MPJ}_{m,2}(i,x) := x_i$, where x_i denotes the ith bit of the string x. In a similar fashion, we define $\widehat{\text{MPJ}}_{m,2} : [m] \times [n]^{[m]} \to [n]$ as $\widehat{\text{MPJ}}_{m,2}(i, f_2) := f_2(i)$. For $k \geq 3$ we then define

$\text{MPJ}_{m,k} : [m] \times [n]^{[m]} \times \left([n]^{[n]}\right)^{k-3} \times \{0,1\}^n \to \{0,1\}$ and $\widehat{\text{MPJ}}_{m,k} : [m] \times [n]^{[m]} \times \left([n]^{[n]}\right)^{k-2} \to [n]$ as follows:

$$\text{MPJ}_{m,k}(i, f_2, f_3, \ldots, f_{k-1}, x) := \text{MPJ}_{n,k-1}(f_2(i), f_3, \ldots, f_{k-1}, x)\,, \text{ for } k \geq 3$$
$$\widehat{\text{MPJ}}_{m,k}(i, f_2, f_3, \ldots, f_k) := \widehat{\text{MPJ}}_{n,k-1}(f_2(i), f_3, \ldots, f_k)\,, \text{ for } k \geq 3\,.$$

It will be helpful, at times, to view strings in $\{0,1\}^n$ as functions from $[n]$ to $\{0,1\}$ and use functional notation accordingly. Unrolling the recursion in the above definitions, we see that, for $k \geq 2$,

$$\text{MPJ}_{m,k}(i, f_2, \ldots, f_{k-1}, x) \;=\; x \circ f_{k-1} \circ \cdots \circ f_2(i)\,; \quad \widehat{\text{MPJ}}_{m,k}(i, f_2, \ldots, f_k) \;=\; f_k \circ \cdots \circ f_2(i)\,. \quad (1)$$

The most natural formulation of this problem has $m = n$. In this case, we drop n from the notation. Previous work on multiplayer pointer jumping considered only MPJ_k and $\widehat{\text{MPJ}}_k$. In the next section, we prove Theorem 1 by performing round elimination on $\text{MPJ}_{m,k}$ and shrinking m at each step.

For many of our results, we shall make use of the following sequences of numbers, all of which are parameterized by some $\delta \in \mathbb{R}^+$ (possibly dependent on n and k) to be specified later. Let $a_0 := 0$, and for $\ell > 0$, let $a_\ell := \delta 2^{a_{\ell}-1}$. For all $\ell \geq 0$, let $m_\ell := n2^{-a_\ell}$. Note that $m_0 = n$. Also, let $\phi(k)$ be the least δ such that $a_{k-1} \geq 1$.

3 Proof of the Main Theorem

We now prove the lower bound on myopic MPJ_k protocols. We repeat the main theorem here for convenience:

Theorem 6. *(Precise restatement of Theorem 1). Let $\mathcal{P}$ be a myopic protocol for MPJ_k. Then, $\text{cost}(\mathcal{P}) > n/2$.*

We prove this theorem by viewing MPJ_k as a special instance of $\text{MPJ}_{m,k}$ and by using a round elimination lemma. First, we note that $\text{MPJ}_{m,2}$ is just the INDEX problem on m bits. The one-way communication complexity of INDEX is well known; we state it here in terms of $\text{MPJ}_{m,2}$.

Fact 7. *If $\mathcal{P}$ is a protocol for $\text{MPJ}_{m,2}$, then $\text{cost}(\mathcal{P}) \geq m$.*

The structure of our proof is as follows. We assume the existence of a protocol for MPJ_k in which each player sends at most δn bits. In the round elimination step, we show how to turn a protocol for $\text{MPJ}_{m,k}$ into a protocol for $\text{MPJ}_{m',k-1}$ with the same cost, and with $m' < m$. Repeating this step $k-2$ times, transforms the δn-bit protocol for MPJ_k into a δn-bit protocol for $\text{MPJ}_{m,2}$ with $m > \delta n$, contradicting Fact 7.

The following simple definition and lemma provide the combinatorial hook that will permits the round elimination step.

Definition 1. Let $i \in [\ell]$ and $\mathcal{F} \subseteq [n]^\ell$ be given. The *range* of i in $\mathcal{F}$, denoted $\text{Range}(i, \mathcal{F})$, is defined as:

$$\text{Range}(i, \mathcal{F}) := \bigcup_{f \in \mathcal{F}} \{f(i)\}$$

Lemma 8. *Let $\mathcal{F} \subseteq [n]^\ell$ be given. If $|\mathcal{F}| \geq m^\ell$, then there exists $i \in [\ell]$ with $|\text{Range}(i, \mathcal{F})| \geq m$.*

Proof. We prove the contrapositive of this statement. Suppose that $|\text{Range}(i, \mathcal{F})| < m$ for all $i \in [\ell]$. Without loss of generality, assume that $\text{Range}(i, \mathcal{F}) \subseteq [m-1]$ for each i, and let $\mathcal{G} := \{f : f(i) \leq m-1 \text{ for all } i \in [\ell]\}$. Its clear that $\mathcal{F} \subseteq \mathcal{G}$. Furthermore, $|\mathcal{G}| = (m-1)^\ell$. Hence, $|\mathcal{F}| \leq |\mathcal{G}| < m^\ell$. ☐

Lemma 9 (**Round Elimination Lemma**). *Let $k \geq 3$. If there is a δn-bit protocol $\mathcal{P}$ for $\text{MPJ}_{m,k}$, then there is a δn-bit protocol $\mathcal{Q}$ for $\text{MPJ}_{m',k-1}$ with $m' = n \cdot 2^{-\delta n/m}$.*

Proof. In $\text{MPJ}_{m,k}$, PLR_1's input is a function $f_2 : [m] \to [n]$. There are n^m such functions. Since PLR_1 sends at most δn bits, he must send the same message M on $n^m/2^{\delta n}$ distinct f_2. Let $\mathcal{F}$ be the set of inputs for which PLR_1 sends M. It follows that $|\mathcal{F}| \geq n^m/2^{\delta n} = 2^{m \log n - \delta n} = 2^{m(\log n - \delta n/m)} = 2^{m \log m'} = (m')^m$. By Lemma 8, we must have $i \in [m]$ with $|\operatorname{Range}(i, \mathcal{F})| \geq m'$. Fix such an i, and let $S := \operatorname{Range}(i, \mathcal{F})$. Without loss of generality, assume $S = [m']$. [2]

We are now ready to construct a protocol for $\text{MPJ}_{m',k-1}$. Label the players $\text{PLR}_2, \ldots, \text{PLR}_k$. For each $j \in [m']$, the players agree on a $g_j \in \mathcal{F}$ such that $g_j(i) = j$. Then, on input $(j, f_3, \ldots, f_{k-1}, x)$, players simulate $\mathcal{P}$ on input $(i, g_j, f_3, \ldots, f_{k-1}, x)$. Clearly, $\operatorname{cost}(\mathcal{Q}) = \operatorname{cost}(\mathcal{P})$, and since $g_j(i) = j$, we must have $\text{MPJ}_{m,k}(i, g_j, f_3, \ldots, f_{k-1}, x) = \text{MPJ}_{m',k-1}(j, f_3, \ldots, f_{k-1}, x)$. □

Note that the reduction step in the round elimination lemma uses only the first two layers of input, so the lemma can be applied to a much wider range of problems than just $\text{MPJ}_{m,k}$. In particular, the lemma applies to $\widehat{\text{MPJ}}_{m,k}$ exactly as stated.

Corollary 10. *Let $k \geq 3$. If there is a δn-bit protocol $\mathcal{P}$ for $\widehat{\text{MPJ}}_{m,k}$, then there is a δn-bit protocol $\mathcal{Q}$ for $\widehat{\text{MPJ}}_{m',k-1}$ with $m' = n \cdot 2^{-\delta n/m}$.*

Proof of Theorem 6. The main theorem follows by careful application of the round elimination lemma. Suppose $\mathcal{P}$ is a δn-bit myopic protocol for $\text{MPJ}_k = \text{MPJ}_{m_0,k}$. By the Round Elimination Lemma, a δn-bit protocol for $\text{MPJ}_{m_\ell,z}$ yields a δn-bit protocol for $\text{MPJ}_{m',z-1}$, where $m' = n \cdot 2^{-\delta n/m_\ell} = n \cdot 2^{-\delta n/(n 2^{-a_\ell})} = n \cdot 2^{-\delta 2^{a_\ell}} = n \cdot 2^{-a_{\ell+1}} = m_{\ell+1}$. Applying the lemma $k-2$ times, we transform $\mathcal{P}$ into a δn-bit protocol for $\text{MPJ}_{m_{k-2},2}$. By Fact 7, we must have $\delta n \geq m_{k-2} = n 2^{-a_{k-2}}$, hence $1 \leq \delta 2^{a_{k-2}} = a_{k-1}$. Therefore, cost$(\mathcal{P}) \geq \phi(k) n$. (Recall that $\phi(k)$ is precisely the least δ such that $a_{k-1} \geq 1$.)

We complete the proof by showing that $\phi(k) > 1/2$. Specifially, we claim that if $\delta \leq 1/2$, then $a_\ell < 1$ for all $\ell > 0$. We prove this claim by induction. In the base case, $a_1 = \delta 2^{a_0} \leq 1/2 < 1$, and if $a_\ell < 1$, then $a_{\ell+1} = \delta 2^{a_\ell} < (1/2) \cdot 2^1 = 1$. □

Next, we show how to extend this to an exact lower bound for the total communication of myopic protocols.

Corollary 11. *For all $m \leq n$, any myopic protocol $\mathcal{P}$ for $\text{MPJ}_{m,k}$ must have $\operatorname{tcost}(\mathcal{P}) \geq m$.*

Proof. We prove this by induction on k. The base case $\text{MPJ}_{m,2}$ is trivial. For the general case, assume that for all $m \leq n$, any protocol for $\text{MPJ}_{m,k-1}$ requires m bits, and suppose there is a protocol $\mathcal{P}$ for $\text{MPJ}_{m,k}$ where PLR_1 sends m_1 bits. The reduction in Lemma 9 gives a protocol $\mathcal{Q}$ for $\text{MPJ}_{m',k-1}$ where $m' = n \cdot 2^{-\delta n/m} = n \cdot 2^{-m_1/m}$. By the induction hypothesis, $\operatorname{tcost}(\mathcal{Q}) \geq m'$. Therefore, $\operatorname{tcost}(\mathcal{P}) \geq m_1 + m'$. Next, note that

$$m_1 + m' < m \quad \Leftrightarrow \quad m_1 + n \cdot 2^{-m_1/m} < m \tag{2}$$

$$\Leftrightarrow \quad n < 2^{m_1/m}(m - m_1) \tag{3}$$

$$\Leftrightarrow \quad n < 2^{\alpha} m(1 - \alpha). \tag{4}$$

where $\alpha = m'/m \in [0, 1]$. The function $f(x) = 2^x(1 - x)$ is decreasing on all $x \in [0, 1]$, so it achieves its maximal value at $f(0) = 1$. Hence inequality (4) becomes $n < m$. However, by assumption, $m \leq n$, so this cannot be true. Therefore, $m_1 + m' \geq m$, completing the proof. □

[2]Specifically, if $S \neq [m']$, then fix a permutation $\pi \in S_n$ that maps (a subset of) S to $[m']$. In $\mathcal{Q}$, players agree on g_j such that $\pi(g_j(i)) = j$ and simulate $\mathcal{P}$ on input $(i, g_j, f_3 \circ \pi, \ldots, f_{k-1}, x)$. $f_3(j) = f_3(\pi(g_j(i))) = f_3 \circ \pi(g_j(i))$, and the rest of the proof follows.

Our main theorem shows that no matter how many players are involved, someone must send at least $\phi(k)n > n/2$ bits. For specific k, the constant factor can be improved. For example, a δn-bit protocol for MPJ_3 gives a δn-bit protocol for $\text{MPJ}_{m,2}$ with $m = n \cdot 2^{-\delta}$. By Lemma 7, we must have $n \cdot 2^{-\delta} \leq \delta n$, or $\delta 2^{\delta} \geq 1$. Solving for δ gives a lower bound of $\approx 0.6412n$.

Next we give a similar theorem for $\widehat{\text{MPJ}}_k$.

Theorem 12. *(Restatement of Theorem 4). Fix $2 \leq k < \log^* n$, and let $\mathcal{P}$ be a myopic protocol for $\widehat{\text{MPJ}}_k$. Then, $\text{cost}(\mathcal{P}) \geq n(\log^{(k-1)} n - \log^{(k)} n)$ bits.*

As in the lower bound proof for MPJ_k, we begin with an easy lower bound for $\widehat{\text{MPJ}}_{m,2}$.

Fact 13. *In any deterministic protocol for $\widehat{\text{MPJ}}_{m,2}$, PLR_1 communicates at least $m \log n$ bits.*

Theorem 12 is a direct consequence of the following lemma:

Lemma 14. *If $\delta = \log^{(k-1)} n - \log^{(k)} n$, then $a_j \leq \log^{(k-j)} n - \log^{(k+1-j)} n$ for all $1 \leq j < k$. In particular, $a_{k-1} \leq \log n - \log\log n$.*

Proof. (by induction) For $j = 1$, $a_j = a_1 = \delta = \log^{(k-1)} n - \log^{(k)} n = \log^{(k-j)} n - \log^{(k+1-j)} n$. For the induction step, we have

$$\begin{aligned} a_{j-1} &\leq \log^{(k+1-j)} n - \log^{(k+2-j)} n \\ &= \log\left(\frac{\log^{(k-j)} n}{\log^{(k+1-j)} n}\right) \end{aligned}$$

Therefore, $2^{a_{j-1}} \leq \frac{\log^{(k-j)} n}{\log^{(k+1-j)} n}$, and

$$\begin{aligned} a_j &= \delta 2^{a_{j-1}} \\ &\leq \left(\log^{(k-1)} n - \log^{(k)} n\right)\left(\frac{\log^{(k-j)} n}{\log^{(k+1-j)} n}\right) \\ &= \frac{\log^{(k-1)} n \log^{(k-j)} n}{\log^{(k+1-j)} n} - \frac{\log^{(k)} n \log^{(k-j)} n}{\log^{(k+1-j)} n} \\ &\leq \log^{(k-j)} n - \log^{(k+1-j)} n \end{aligned}$$

where the last inequality is because the positive term is less than $\log^{(k-j)} n$, and the negative term is greater than $\log^{(k+1-j)} n$, for all $2 \leq j < k$. □

Proof of Theorem 12. Let $\delta = \log^{(k-1)} n - \log^{(k)} n$. Suppose we have a protocol for $\widehat{\text{MPJ}}_k$ in which each player sends δn bits. By Corollary 10, we have a δn-bit protocol for $\widehat{\text{MPJ}}_{m_{k-2},2}$. By Fact 13, such a protocol costs at least $m_{k-2} \log n$ bits. Hence, we must have

$$\begin{aligned} \delta n \geq m_{k-2} \log n \quad &\Leftrightarrow \quad \delta n \geq n 2^{-a_{k-2}} \log n \\ &\Leftrightarrow \quad \delta 2^{a_{k-2}} \geq \log n \\ &\Leftrightarrow \quad a_{k-1} \geq \log n \end{aligned}$$

However, we know by Lemma 14 that $a_{k-1} \leq \log n - \log\log n < \log n$, so we have a contradiction. □

4 An upper bound for Myopic protocols

The analysis for the lower bound in the previous section also gives insight as to what myopic protocols *can* do. Specifically, in a protocol for $\text{MPJ}_{m,k}$, we'd like PLR_1's message to give PLR_2 enough information so that $\text{PLR}_2, \ldots, \text{PLR}_k$ can run a protocol for $\text{MPJ}_{m',k-1}$ for some $m' < m$. To do this, we need PLR_1's messages to partition his input space so that for each of his messages M_j and for each $1 \leq i \leq m$, the range size $|\operatorname{Range}(i, M_1)|$ is small.

It turns out that just such a protocol is possible, and that the communication cost matches our lower bound up to $1 + o(1)$ factors. To aid in the analysis of this protocol, we need the following *covering lemma*.

Definition 2. We say a subset $T \subseteq [m]^d$ is isomorphic to $[m']^d$ and write $T \cong [m']^d$ if $T = T_1 \times \cdots \times T_d$ for sets $T_1, \ldots, T_d \subseteq [m]$, each of size m'.

Lemma 15. ***(Covering Lemma).*** *For integers $d, m, m' < m \in \mathbb{Z}_{>0}$, let $\mathcal{U}_{m,d} := [m]^d$, and $\mathcal{S}_{m',d} := \{T \subseteq \mathcal{U}_{m,d} : T \cong [m']^d\}$. Then there exists a set $\mathcal{C} \subseteq \mathcal{S}_{m',d}$ of size $|\mathcal{C}| \leq (m/m')^d \cdot d \ln m + 1$ such that $\cup_{T \in \mathcal{C}} T = \mathcal{U}_{m,d}$. We say that $\mathcal{C}$* covers *$\mathcal{U}_{m,d}$ and call $\mathcal{C}$ an m'-covering of $\mathcal{U}_{m,d}$.*

Proof. We use the probabilistic method. Fix $r > (m/m')^d d \ln m$, and pick $T_1, \ldots, T_r$ independently and uniformly at random from $\mathcal{S}_{m',d}$. Note that picking T in this way amounts to picking d $[m']$-subsets of $[m]$ independently and uniformly at random. Therefore, for any $p \in \mathcal{U}_{m,d}$, we have $\Pr[p \in T] = (m'/m)^d$. For each $p \in \mathcal{U}_{m,d}$, let $BAD_p := \bigwedge_{1 \leq j \leq r} (p \notin T_j)$ be the event that p is not covered by any set T_j. Also, let $BAD := \bigvee_{p \in \mathcal{U}_{m,d}} BAD_p$ be the event that *some* p is not covered. From the probability calculation above, and using the fact that $1 + x \leq e^x$, we have $\Pr[BAD_p] = \left(1 - (m'/m)^d\right)^r \leq e^{-r(m'/m)^d}$. By the union bound, we have $\Pr[BAD] \leq m^d \Pr[BAD_p] \leq e^{d \ln m - r(m'/m)^d}$. Recall that $r > (m/m')^d \cdot d \ln m$, so $d \ln m - r(m'/m)^d < d \ln m - d \ln m = 0$. Hence, $\Pr[BAD] < e^0 = 1$. Therefore, there must exist a set $\{T_1, \ldots, T_r\}$ of sets isomorphic to $[m']^d$ that cover $\mathcal{U}_{m,d}$. ☐

Theorem 16. *For all $k \geq 3$, there exists a deterministic protocol for MPJ_k in which each player sends $\phi(k) n (1 + o(1))$ bits.*

Proof. We prove this by construction. As a warmup, we give a $(0.65n)$-bit max-communication protocol for MPJ_3. Later, we show how to generalize this to more than 3 players. Recall that we have a $\phi(3)n$-bit lower bound for MPJ_3, where $\phi(3) \sim 0.6412$ is the unique real number δ such that $a_2 = \delta 2^\delta = 1$. In advance, the players fix a $[0.65n]$-covering $\mathcal{C}$ of $[n]^{[n]}$. On input (i, f_2, x), PLR_1 sends $T \in \mathcal{C}$ such that $f_2 \in T$. PLR_2 sees i, x and T, and sends x_j for all $j \in \operatorname{Range}(i, T)$. PLR_3 sees i, f_2 and recovers $x_{f_2(i)}$ from PLR_2's message.

In terms of communication cost, PLR_1 sends $\log |\mathcal{C}|$ bits. By Lemma 15, $|\mathcal{C}| \leq (n/0.65n)^n \cdot n \ln n + 1$, hence PLR_1 sends $\log |\mathcal{C}| = n \log(1/0.65)(1 + o(1)) < 0.65n$ bits. PLR_2 sends one bit for each $j \in \operatorname{Range}(i, T)$. Since $T \cong [0.65n]^n$, we must have $|\operatorname{Range}(i, T)| \leq 0.65n$. Hence, PLR_2 sends at most $0.65n$ bits, and the maximum communication cost is also 0.65 bits.

For the general case, we construct a protocol $\mathcal{P}$ for MPJ_k as follows. Fix $\delta := \phi(k)$, and for each $0 \leq j \leq k-2$, players agree in advance on a $[m_{j+1}]$-covering set $\mathcal{C}_{j+1}$ for $\mathcal{U}_{n,m_j}$. Note that by the covering lemma, $\log |\mathcal{C}_{j+1}| = m_j \log(n/m_{j+1})(1 + o(1))$. Also note that

$$\begin{aligned}
m_j \log(n/m_{j+1}) &= n 2^{-a_j} \log\left(n / n 2^{-a_{j+1}}\right) \\
&= -n 2^{-a_j} \log(2^{-a_{j+1}}) \\
&= n 2^{-a_j} a_{j+1} \\
&= n 2^{-a_j} (\delta 2^{a_j}) \\
&= \delta n.
\end{aligned}$$

On input $(i, f_2, \ldots, f_{k-1}, x)$, the players proceed as follows. $\textsc{plr}_1$ sees $f_2 \in [n]^{[n]}$ and picks $T_1 \in \mathcal{C}_1$ that contains f_2. $\textsc{plr}_1$ communicates T_1 to the rest of the players.

$\textsc{plr}_2$ sees $i \in [m]$, $f_3 \in [n]^{[n]}$, and T_1. From i and T_1, $\textsc{plr}_2$ computes $R_2 := \mathrm{Range}(i, T_1)$. Note that since T_1 is an $[m_1]$ covering, $|\mathrm{Range}(i, T_1)| = m_1$ for all i. Without loss of generality, assume $R_2 = [m_1]$. Let f_3^* be f_3 restricted to the domain R_2. Note that f_3^* is a function $[m_1] \to [n]$, so $f_3^* \in \mathcal{U}_{n,m_1}$. $\textsc{plr}_2$ picks $T_2 \in \mathcal{C}_2$ that contains f_3^* and communcates T_2 to the rest of the players.

Generalizing, $\textsc{plr}_j$ computes $R_j := \mathrm{Range}(f_{j-1} \circ \cdots \circ f_2(i), T_{j-1})$, which has size m_{j-1} because $T_{j-1} \in \mathcal{C}_{j-1}$. Noting that f_j restricted to R_j is an element in $\mathcal{U}_{n,m_{j-2}}$, $\textsc{plr}_j$ picks $T_j \in \mathcal{C}_j$ that contains f_j and commicates this to the rest of the players.

$\textsc{plr}_{k-1}$ computes $R_{k-1} := \mathrm{Range}(f_{k-2} \circ \cdots \circ f_2(i), T_{k-2})$ and sends x_r for each $r \in R_{k-1}$. $\textsc{plr}_k$ computes $r^* := f_{k-1} \circ f_{k-2} \circ \cdots \circ f_2(i)$ and recovers x_{r^*} from $\textsc{plr}_{k-1}$'s message.

For each $1 \le j \le k-2$, $\textsc{plr}_j$ sends $\log |\mathcal{C}_{j+1}| = \delta n(1 + o(1))$ bits. $\textsc{plr}_{k-1}$ sends one bit for each $j \in R_{k-1}$. By construction, $|R_{k-1}| \le m_{k-2}$. Choosing δ to be the smallest real such that $\delta 2^{a_{k-2}} = a_{k-1} \ge 1$ ensures that $m_{k-2} \le \delta n$.

In conclusion, we have a protocol $\mathcal{P}$ where each player sends $\delta n(1 + o(1))$ bits, where δ is the smallest real such that $a_{k-1} \ge 1$. Note that this choice of δ exactly matches our lower bound. □

5 Randomizing the Lower Bound

Theorems 6 and 12 give strong lower bounds for deterministic protocols for $\textsc{mpj}_k$ and $\widehat{\textsc{mpj}}_k$ respectively. In this section, we show that our technique can also be used to show lower bounds on the randomized complexity of $\textsc{mpj}_k$.

Previously, Chakrabarti [Cha07] showed randomized lower bounds of $\Omega(n/k)$ and $\Omega(n \log^{(k-1)} n)$ for $\textsc{mpj}_k$ and $\widehat{\textsc{mpj}}_k$ rescpectively. The bound for $\widehat{\textsc{mpj}}_k$ is for the maximum communication and is tight. The bound for $\textsc{mpj}_k$ is for the total communication; this bound implies an $\Omega(n/k^2)$ lower bound on the maximum communication. In contrast, we achieve:

Theorem 17. *In any randomized myopic protocol for $\textsc{mpj}_k$, some player must communicate at least $\Omega(n/k \log n)$ bits.*

Our lower bound improves on the bound from [Cha07] for $k = \Omega(\log n)$. To prove this lower bound, we give a round elimination lemma for ϵ-error distributional protocols for $\textsc{mpj}_{m,k}$. Our "base case" is a lower bound on the ϵ-error distributional complexity of $\textsc{mpj}_{m,k}$, due to Ablayev [Abl96]:

Fact 18. *Any protocol for $\textsc{mpj}_{m,2}$ that errs on at most an ϵ-fraction of the inputs distributed uniformly must communicate at least $m(1 - H(\epsilon))$ bits.*

Lemma 19 (Round Elimination Lemma). *Let $k \ge 3$. If there is a δn-bit, ϵ-error distributional protocol $\mathcal{P}$ for $\textsc{mpj}_{m,k}$, then there is a δn-bit, $\hat{\epsilon}$-error distributional protocol $\mathcal{Q}$ for $\textsc{mpj}_{m',k-1}$ with $m' = n \cdot 2^{-2\delta \frac{n}{m}}$ and $\hat{\epsilon} = 2n\epsilon$.*

Proof. For the sake of notation, we let $z := (f_3, \ldots, f_{k-1}, x)$, so the input to $\textsc{mpj}_{m,k}$ is (i, f_2, z). Let $\mathcal{P}(i, f_2, z)$ denote the output of $\mathcal{P}$ on input (i, f_2, z). Let

$$\alpha(i, f_2, z) := \begin{cases} 1 & \text{if } \mathcal{P}(i, f_2, z) \neq \textsc{mpj}_{m,k}(i, f_2, z) \\ 0 & \text{otherwise} \end{cases}$$

Since $\mathcal{P}$ is an ϵ-error protocol, we have $\mathrm{E}_{i,f_2,z}[\alpha(i, f_2, z)] = \epsilon$. Now, let $\hat{\alpha}(i, f_2) := \mathrm{E}_z[\alpha(i, f_2, z)]$, and call (i, f_2) *bad* if $\hat{\alpha}(i, f_2) > 2n\epsilon$; otherwise, call (i, f_2) *good*. Clearly, $\mathrm{E}_{i,f_2}[\hat{\alpha}(i, f_2)] = \mathrm{E}_{i,f_2,z}[\alpha(i, f_2, z)] = \epsilon$, so by Markov's inequality, we get $\Pr[(i, f_2) \text{ is } \textit{bad}] < 1/2n$. Now, let

$$\beta(i, f_2) := \begin{cases} 1 & \text{if } (i, f_2) \text{ is bad} \\ 0 & \text{otherwise} \end{cases}$$

Also, let $\hat{\beta}(f_2) = \mathrm{E}_i[\beta(i, f_2)]$. Call f_2 *bad* if $\hat{\beta}(i, f_2) \geq 1/n$, and call f_2 *good* otherwise. Note that $\mathrm{E}_{f_2}[\hat{\beta}(f_2)] = \mathrm{E}_{i,f_2}[\beta(i, f_2)] < 1/(2n)$, so by another application of Markov's inequality, we get $\Pr[f_2 \text{ is } \textit{bad}] < 1/2$. Therefore, f_2 is good with probability at least $1/2$.

Note that if f_2 is good, then $\Pr_i[(i, f_2) \text{ is } \textit{good}] < 1/n$. Furthermore, if (i, f_2) were good for even a single i, then we would have $\Pr_i[(i, f_2) \text{is } \textit{good}] \geq 1/n$. Therefore, (i, f_2) is good for *every* i whenever f_2 is good.

The rest of this lemma closely follows the deterministic version. There are n^m functions $f_2 : [m] \to [n]$. Since at least half the functions f_2 are *good*, there must be at least $n^m/2$ good f_2. Since PLR_1 sends at most δn bits, he must send the same message M_1 on $n/(2 \cdot 2^{\delta n})$ distinct *good* f_2. Let $\mathcal{F}$ be the set of good inputs for which PLR_1 sends M_1. It follows that $|\mathcal{F}| \geq \frac{n^m}{2 \cdot 2^{\delta n}} = 2^{m \log n - 1 - \delta n} > 2^{m \log n - 2\delta n} = (m')^m$. By Lemma 8, we must have $i \in [m]$ with $|\operatorname{Range}(i, \mathcal{F})| \geq m'$. Furthermore, every $f \in \mathcal{F}$ is *good*, so (i, f) is *good* for all $f \in \mathcal{F}$. Construct a protocol $\mathcal{Q}$ for $\mathrm{MPJ}_{m',k-1}$ as we did in Lemma 9. As in Lemma 9, the cost of $\mathcal{Q}$ remains equal to the cost of $\mathcal{P}$, $\mathrm{MPJ}_{m,k}(i, g_j, z) = \mathrm{MPJ}_{m',k-1}(j, z)$, and that $\mathcal{Q}(j, z) = \mathcal{P}(i, g_j, z)$. Finally, we get

$$\begin{aligned} \Pr_{j,z}[\mathcal{Q}(i, z) \neq \mathrm{MPJ}_{m',k-1}(j, z)] &= \Pr_{j,z}[\mathcal{P}(i, g_j, z) \neq \mathrm{MPJ}_{m,k}(i, g_j, z)] \\ &= \Pr_{j,z}[\alpha(i, g_j, z) = 1] \\ &\leq 2n\epsilon \end{aligned}$$

where the inequality holds because (i, g_j) is good for every j. □

Proof of Theorem 17. Let $\epsilon = 1/3$ and $\delta = 1/32$, and suppose an ϵ-error randomized protocol for MPJ_k exists where each player sends at most $t = \frac{n}{48\delta \ln 2(\log 3 + (k-2)\log(2n))} = \Omega(\frac{n}{k \log n})$ bits. By Chernoff bounds, there exists an $\hat{\epsilon} := \epsilon\,(2n)^{-(k-2)}$-error randomized protocol $\mathcal{P}$ for MPJ_k, where each player sends δn bits. By Yao's minimax lemma, there is a deterministic protocol where each player sends δn bits that errs on an $\hat{\epsilon}$ fraction of inputs, distributed uniformly.

Set $a_0 = 0, a_\ell = 2\delta 2^{a_{\ell-1}}$, and $m_\ell = n 2^{-a_\ell}$. Note that $a_0 < 1/8$, and if $a_{\ell-1} < 1/8$, then $a_\ell = 2\delta 2^{a_{\ell-1}} < 1/8$, so by induction, $a_\ell < 1/8$ for all ℓ. Using Lemma 19 $k-2$ times, we get a δn-bit, ϵ-error protocol for $\mathrm{MPJ}_{m_{k-2},2}$. Combining this with Fact 18, we get

$$\begin{aligned} \delta n \geq m_{k-2}\,(1 - H(1/3)) &\Leftrightarrow \delta n \geq n 2^{-a_{k-2}}\,(1 - H(1/3)) \\ &\Leftrightarrow \delta 2^{a_{k-2}} \geq 1 - H(1/3) \\ &\Leftrightarrow a_{k-1}/2 \geq 1 - H(1/3) \end{aligned}$$

However, we have already seen that $a_{k-1}/2 < 1/16 < 1 - H(1/3)$, so this is a contradiction. □

6 Concluding Remarks

In this paper, we characterize the power of deterministic myopic protocols for MPJ_k. We have shown that it is essentially necessary and sufficient for each player to send $n/2$ bits of communication. When considering the total communication of a protocol, we show that the trivial protocol is the best myopic protocol possible. Finally, we show how to randomize our result. We hope this provides another concrete step towards showing that $\mathrm{MPJ}_k \notin \mathsf{ACC}^0$.

Several questions relating to pointer jumping remain. It remains open whether $\mathrm{MPJ}_k \in \mathsf{ACC}^0$ or not. More generally, the gap between the upper and lower bounds on the communication complexity remain large. Based on the bounds in this and other work, it appears that randomization does not help this problem much; however, that remains a conjecture. It would be interesting to know if there are any randomized protocols for any pointer jumping problem (even with any input restriction) that are significantly better than the known deterministic lower bounds.

The current lower bounds seem to rely heavily on restrictions to either the input model or which parts of the input are seen by each player. This work relies heavily on the fact that each player sees only a single layer of input in front of them. The technique of Viola and Wigderson is dependent on a tree-structure to the inputs. Relaxing either of these restrictions might prove fruitful.

7 Acknowledgements

We would like to thank the Amit Chakrabarti, Piotr Indyk, and the Dartmouth Theory Reading Group for helpful discussions about various aspects of this paper. We would like to especially thank Amit Chakrabarti for providing useful advice on a preliminary draft of this paper.

References

[Abl96] Farid Ablayev. Lower bounds for one-way probabilistic communication complexity and their application to space complexity. *Theoretical Computer Science*, 175(2):139–159, 1996.

[Ajt88] Miklòs Ajtai. A lower bound for finding predecessors in Yao's cell probe model. em Combinatorica, 8(3):235-247, 1988.

[AMS99] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999. Preliminary version in *Proc. 28th Annu. ACM Symp. Theory Comput..*, pages 20–29, 1996.

[BT94] Richard Beigel and Jun Tarui. On ACC. *Comput. Complexity*, 4:350–366, 1994.

[BC08] Joshua Brody and Amit Chakrabarti. Sublinear Communication Protocols for Multi-Party Pointer Jumping and a Related Lower Bound. In *Proc. 13th International Symposium on Theoretical Aspects of Computer Science*, pages 145-157, 2008.

[CFL83] Ashok K. Chandra, Merrick L. Furst, and Richard J. Lipton. Multi-party protocols. In *Proc. 15th Annual ACM Symposium on the Theory of Computing*, pages 94–99, 1983.

[Cha07] Amit Chakrabarti. Lower bounds for multi-player pointer jumping. In *Proc. 22nd Annual IEEE Conference on Computational Complexity*, pages 33–45, 2007.

[CJP08] Amit Chakrabarti, T. S. Jayram, and Mihai Pătraşcu. Tight lower bounds for selection in randomly ordered streams. In *Proc. 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2008.

[DJS98] Carsten Damm, Stasys Jukna, and Jiří Sgall. Some bounds on multiparty communication complexity of pointer jumping. *Comput. Complexity*, 7(2):109–127, 1998. Preliminary version in *Proc. 13th International Symposium on Theoretical Aspects of Computer Science*, pages 643–654, 1996.

[Gro06] Andre Gronemeier. NOF-Multiparty Information Complexity Bounds for Pointer Jumping. In *Proc. 31st International Symposium on Mathematical Foundations of Computer Science*, 2006.

[GM07] Sudipto Guha and Andrew McGregor. Lower bounds for quantile estimation in random-order and multi-pass streaming. In *Proc. 34th International Colloquium on Automata, Languages and Programming*, pages 704–715, 2007.

[Har08] Nicholas J. A. Harvey Matroid intersection, pointer chasing, and Young's seminormal representation of S_n In *Proc. 19th annual ACM-SIAM Symposium on Discrete Algorithms*, 2008.

[HG91] Johan Håstad and Mikael Goldmann. On the power of small-depth threshold circuits. *Comput. Complexity*, 1:113–129, 1991.

[KW88] Mauricio Karchmer and Avi Wigderson. Monotone circuits for connectivity require super-logarithmic depth. In *Proc. 20th annual ACM symposium on Theory of computing*, pages 539-550, 1988.

[Mil94] Peter Bro Milterson. Lower Bounds for Union-Split-Find related problems on Random Access Machines. In *Proc. 26th annual ACM symposium on Theory of computing*, pages 625-634, 1994.

[VW07] Emanuele Viola and Avi Wigderson. One-way multi-party communication lower bound for pointer jumping with applications. In *Proc. 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 427–437, 2007.

[Yao90] Andrew C. Yao. On ACC and threshold circuits. In *Proc. 31st Annual IEEE Symposium on Foundations of Computer Science*, pages 619–627, 1990.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de/