Theory Comput Syst (2014) 54:13–23
DOI 10.1007/s00224-013-9476-x

Optimal Collapsing Protocol for Multiparty Pointer Jumping

Hongyu Liang

Published online: 9 May 2013

Abstract In this paper, we study the pointer jumping problem under the one-way number-on-the-forehead (NOF) multiparty communication model. This problem is widely considered to be a candidate for proving strong lower bounds under the NOF model, and has applications to proving lower bounds for many other problems.

We investigate the maximum communication complexity of collapsing protocols for pointer jumping, where each player sees all layers behind her and only the composition of layers ahead of her. We present a collapsing protocol in which every player communicates at most $n - \frac{1}{2}\log_2 n + 1$ bits, which tightly matches the lower bound of $n - \frac{1}{2}\log_2 n - 2$ given by Brody and Chakrabarti (in Proc. 25th Annual Symposium on Theoretical Aspects of Computer Science (STACS), pp. 145–156, 2008). Actually, in our protocol only three players need to communicate information: the first player sends $\log_2(n+1)$ bits, the second to last player sends $n - \frac{1}{2}\log_2 n + 1$ bits, and the last player just outputs the answer. A natural question is whether the $\log_2(n+1)$ bits communicated by the first player is necessary for achieving a low maximum communication complexity. We make progress towards this question by proving that in any collapsing protocol for the 3-player pointer jumping problem, if the first player only sends one bit, then the second player must communicate at least $n-2$ bits.

Keywords Multiparty communication complexity · Pointer jumping · Number-on-the-forehead model · Collapsing protocol · Total influence

This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, and the National Natural Science Foundation of China Grant 61033001, 61061130540, 61073174.

H. Liang (✉)
Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China
e-mail: lianghy08@mails.tsinghua.edu.cn

1 Introduction

Communication complexity has received a great amount of research attention since its initiation by Yao [27]. The techniques and results in communication complexity have been successfully applied to proving lower bounds for problems in a wide variety of areas, including streaming algorithms [3, 12, 18], circuit complexity [7, 19], data structures [2, 21], metric embedding [4], property testing [8], and proof complexity [6]. See [20] for a comprehensive treatment of this area.

In this paper we consider the *pointer jumping* problem under the setting of multiparty communication complexity, which has strong relations with circuit lower bounds. By the results of Beigel and Tarui [7], to show non-membership of a specific function in the complexity class ACC^0, it suffices to prove a polynomial lower bound for this function under the *number-on-the-forehead* (NOF) communication model [13] with polynomially many players. The pointer jumping problem is widely considered to be a good candidate for this task.

In the NOF multiparty communication model, there are k players $\mathrm{PLR}_1, \mathrm{PLR}_2, \ldots, \mathrm{PLR}_k$, who will collaborate to compute a function $f(x_1, x_2, \ldots, x_k)$, provided that the ith player PLR_i sees all x_j's with $j \neq i$ but cannot see x_i. (Just think of the ith input token being written on the forehead of PLR_i.) This model was introduced by Chandra, Furst and Lipton [13]. Note that this model collapses to the normal two-party communication model when $k = 2$. We notice that, in this model, each player in fact grasps a large amount of information, especially when k is large. Thus, intuitively, computation under this model should be easy, and proving lower bounds, on the contrary, might be hard. This intuition has been well reflected during the past years of research. Currently, no nontrivial lower bound has been proved for any explicit function with $\omega(\log N)$ players, where N is the size of the total input.

1.1 The Pointer Jumping Problem

The pointer jumping problem is widely considered to be a good candidate for proving communication lower bounds under the NOF model. On its own right, the problem also has many applications in proving lower bounds for other problems (see [14, 26]). The k-party pointer jumping problem MPJ_k^n is (informally) defined on a layered graph with $k + 1$ layers. The first layer consists of a single vertex t, layers 2 through k each contains n vertices, and layer $k + 1$ has two vertices labeled 0 and 1 respectively. The inputs are the set of edges between consecutive layers. For each vertex v in the first k layers, there is a directed edge from v to some vertex in the next layer. The output of the problem is the (unique) label of the vertex in layer $k + 1$ reachable from t through the directed edges. There are k players $\mathrm{PLR}_1, \mathrm{PLR}_2, \ldots, \mathrm{PLR}_k$, where each player PLR_i sees all the graph except for the edges between the ith and $(i + 1)$th layers. See Fig. 1 for an intuitive illustration. (Imagine that the edges between layers i and $i + 1$ are drawn on the forehead of PLR_i.) The players write messages on a public blackboard in the order $\mathrm{PLR}_1, \mathrm{PLR}_2, \ldots, \mathrm{PLR}_k$, where the message written by PLR_k is regarded as the output. Notice that this is a one-way communication model. Also note that the order of players sending messages is important, since a trivial $O(\log n)$ protocol exists whenever PLR_i can send her message before PLR_j for some $i > j$.

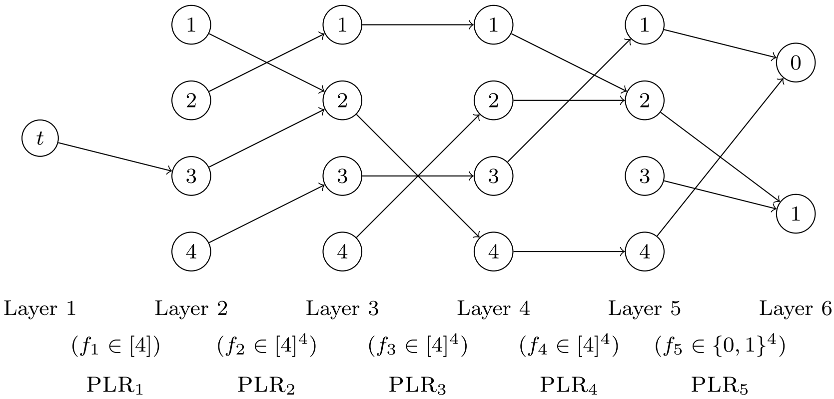

Fig. 1 An illustrative instance of MPJ$^4_5$, i.e., Pointer Jumping with $n=4$ and $k=5$. The output on this instance should be 0

MPJ$^n_2$ is actually equivalent to the one-way two-party communication problem INDEX^n, where player 1 holds a binary string s of length n and player 2 holds an index $i \in \{1, 2, \ldots, n\}$, and player 1 needs to send a message to player 2 so that the latter can determine the ith bit of s. Both deterministic and randomized communication complexity of the INDEX problem have been tightly characterized [1, 20]. Nevertheless, the complexity of MPJ^n_k for larger k is not well understood. Wigderson proved that MPJ$^n_3$ requires $\Omega(\sqrt{n})$ bits of communication based on the idea of [22]. This result is unpublished but was stated in [5]. This lower bound was generalized by Viola and Wigderson [26], who showed that any randomized protocol for MPJ^n_k requires at least $n^{1/(k-1)}/k^{O(k)}$ communication. The lower bound is nontrivial only for small values of k, and no further improvements have been made up to now.

On the upper bound side, Pudlák, Rödl and Sgall [25] found an amazing sublinear protocol (more precisely $O(n \log\log n/\log n)$) for a special case of MPJ$^n_3$ in which the middle layer of edges form a permutation. This idea was extended by Brody and Chakrabarti [11] who gave a protocol for general MPJ^n_k ($k \geq 3$) using $O(n(\frac{k \log\log n}{\log n})^{(k-2)/(k-1)})$ bits of communication. This result dashes the hopes for an $\Omega(n)$ lower bound for constant k, as in the case $k=2$.

Since proving stronger lower bounds for MPJ^n_k seems to be rather difficult, several restricted models and protocols have been considered with the hope that stronger results should be easier to prove under these models, which may in turn inspire the study of the original problem. In the *myopic* model of pointer jumping, introduced by Gronemeier [17], PLR$_i$ only has a limited view of the layers ahead of her; to be specific, she cannot see layers $i+2, \ldots, k$. Brody [9, 10] proved that in any myopic protocol for MPJ^n_k, at least n bits need to be communicated in total, and some player must send at least $n/2$ bits. Protocols with matching upper bounds are also presented in [9]. The *conservative* model was introduced by Damm, Jukna and Sgall [14], in which each player sees the layers ahead of her and the *composition* of the layers behind her. An $\Omega(n/k^2)$ communication lower bound was proved in [14] for MPJ^n_k under the conservative model. Similarly, in the *collapsing* model introduced by Brody and Chakrabarti [11], each player knows all the players behind her, but can only see the composition of the layers in front of her. They proved in [11] that in any collapsing protocol for MPJ_k^n, some player must communicate at least $n - \frac{1}{2}\log_2 n - 2$ bits. This lower bound is in particular interesting, since in their sub-linear protocol for MPJ_k^n [11], all players except for PLR_1 are collapsing, i.e., the information they send only depend on the composition of layers ahead of them. Thus, the linear lower bound for collapsing protocols actually shows that even one non-collapsing player can make the problem much easier.

It is clear that n is an upper bound on the maximum number of bits that each player must communicate in a collapsing protocol for MPJ_k^n. However, to our knowledge, no protocol with better-than-trivial maximum communication complexity has been proposed for MPJ_k^n in the literature. Since there is still a $\Theta(\log n)$ gap between the lower and upper bounds, it is interesting to investigate whether this gap can be bridged, which motivates our study.

1.2 Our Contributions

In this paper, we present a collapsing protocol for MPJ_k^n in which each player communicates at most $n - \frac{1}{2}\log_2 n + 1$ bits, which tightly matches the lower bound of $n - \frac{1}{2}\log_2 n - 2$ given by [11] up to an additive constant. Actually, in our protocol only three players need to communicate information, namely PLR_1, PLR_{k-1} and PLR_k. More specifically, PLR_1 sends $\log_2(n+1)$ bits, PLR_{k-1} sends $n - \frac{1}{2}\log_2 n + 1$ bits, and PLR_k outputs the correct answer. Our result shows that the collapsing model allows nontrivial protocols.

Note that the total communication complexity of our protocol is roughly $n + \frac{1}{2}\log_2 n$, which is worse than that of the trivial protocol in which PLR_1 sends n bits indicating the destination of all the n possible starting pointers, and then any other player can give the correct answer. Thus, one may ask whether we can achieve low total and maximum communication complexity simultaneously. Towards this end, a natural question is whether the $\log_2(n+1)$ bits communicated by PLR_1 in our protocol is necessary. We make progress towards this question by proving that in any collapsing protocol for MPJ_3^n, if PLR_1 sends only one bit, then PLR_2 must communicate at least $n-2$ bits. Although this result seems marginal, the proof technique which uses tools from Boolean function analysis is interesting on its own and might have further applications.

2 Preliminaries

For an integer $m \geq 1$, let $[m]$ denote the set $\{1, 2, \dots, m\}$. We use log to denote the base 2 logarithm. Given a string s of length m, let $s(i)$, $1 \leq i \leq m$, denote the ith character of s. Throughout this paper all characters of a string are positive integers, and thus can be compared with each other. Given two strings a, b of the same length m, we say $a \leq b$ if $a(i) \leq b(i)$ for all $i \in [m]$, and $a < b$ if $a \leq b$ and $a(i) < b(i)$ for some $i \in [m]$. Notice that this forms a partial order on the set of strings of certain length. We identify a function $f : [m] \to A$ with the string $s_f \in A^m$, where $s_f(i) = f(i)$ for all $i \in [m]$. For two functions f_1, f_2, let $f_1 \circ f_2$ denote their composition, that is, $(f_1 \circ f_2)(x) = f_1(f_2(x))$.

We next give the formal definition of the pointer jumping problem considered in this paper. The pointer jumping function $\text{MPJ}_k^n : [n] \times ([n]^n)^{k-2} \times \{0,1\}^n \to \{0,1\}$ is defined as:

$$\text{MPJ}_k^n(f_1, f_2, \ldots, f_k) = (f_k \circ f_{k-1} \circ \cdots \circ f_2)(f_1),$$

where $f_1 \in [n]$, $f_i \in [n]^n$ for $2 \le i \le k-1$, and $f_k \in \{0,1\}^n$. (Intuitively, we can think of f_i as the function induced by the set of edges between layers i and $i+1$; see Fig. 1 for a comparison.)

In the number-on-the-forehead multiparty communication model, there are k players $\text{PLR}_1, \text{PLR}_2, \ldots, \text{PLR}_k$. Each player PLR_i, $1 \le i \le k$, knows $\{f_j \mid j \ne i\}$. In a *(deterministic) protocol* for MPJ_k^n, the players write messages on a public blackboard (i.e., messages are seen by all players) in the order $\text{PLR}_1, \text{PLR}_2, \ldots, \text{PLR}_k$; the last player, PLR_k, should correctly write down $\text{MPJ}_k^n(f_1, f_2, \ldots, f_k)$. We also use MPJ_k^n to denote the problem. The *maximum communication complexity* of a protocol for MPJ_k^n is the maximum number of bits written by any player in the protocol, over all possible instances. As in conventional settings of communication complexity, we assume that players have *unlimited computation power*, i.e., they can compute any computable function, and the time and space consumed by their local computation are not taken into consideration.

In a protocol for MPJ_k^n, a player PLR_i is called *collapsing* if her message depends only on $f_1, f_2, \ldots, f_{i-1}$, and $f_k \circ f_{k-1} \circ \cdots \circ f_{i+1}$. We can equivalently regard PLR_i as only knowing $f_1, f_2, \ldots, f_{i-1}$, and $f_k \circ f_{k-1} \circ \cdots \circ f_{i+1}$. A protocol is called *collapsing* if all players in it are collapsing. All protocols considered in this paper are deterministic, and we always assume that $k \ge 3$.

3 An Optimal Collapsing Protocol

In this section we design a collapsing protocol for MPJ_k^n whose maximum communication complexity tightly matches the lower bound given in [11], up to a small additive constant.

Let $S = \{s_1, s_2, \ldots, s_r\} \subseteq \{0,1\}^n$ be a set of n-bit binary strings. We call S a *chain* if we there exists a permutation of $[r]$, say $(o_1, o_2, \ldots, o_r)$, such that $s_{o_i} < s_{o_{i+1}}$ for all $1 \le i < r$. We will use the following theorem from combinatorics:

Theorem 1 (Sperner's Theorem; see e.g. [15]) *The set* $\{0,1\}^n$ *of* n*-bit binary strings can be partitioned into* $\binom{n}{\lfloor n/2 \rfloor}$ *disjoint chains.*

We assume without loss of generality that all the players know a fixed partition of $\{0,1\}^n$ into $\binom{n}{\lfloor n/2 \rfloor}$ disjoint chains. (This can be done since every player has unlimited computation power, and they can all compute the smallest partition under some orderings on the partitions.) Also, assume these chains are indexed by $1, 2, \ldots, \binom{n}{\lfloor n/2 \rfloor}$, which is known to all players. For each $i \in [k]$, let $g_i = f_k \circ f_{k-1} \circ \cdots \circ f_{i+1}$. We now give our protocol CHAIN for MPJ_k^n as follows.

Protocol CHAIN:

1. PLR_1 writes $|\{i \mid g_1(i) = 1; 1 \le i \le n\}|$ on the blackboard using $\lceil \log(n+1) \rceil$ bits. (Recall that $g_1 = f_k \circ f_{k-1} \circ \cdots \circ f_2 \in \{0,1\}^n$.)
2. PLR_{k-1} writes the index of the chain which contains $f_k \in \{0,1\}^n$ on the blackboard, using $\lceil \log \binom{n}{\lfloor n/2 \rfloor} \rceil \le n - \frac{1}{2} \log n + 1$ bits. (The upper bound estimation can be done by Stirling's formula; see e.g. [16].)
3. PLR_k utilizes the information on the blackboard, as well as $f_1, \ldots, f_{k-1}$, to output the correct answer. (Details of this process will be specified later in the proof.)

Theorem 2 *The above protocol, CHAIN, is a collapsing protocol for* MPJ_k^n *in which each player communicates at most* $n - \frac{1}{2} \log n + 1$ *bits.*

Proof Clearly CHAIN is a collapsing protocol for MPJ_k^n with maximum communication complexity $\max\{\lceil \log(n+1) \rceil, n - \frac{1}{2} \log n + 1\} = n - \frac{1}{2} \log n + 1$ for all $n \ge 1$. We next prove the correctness of this protocol. It suffices to show that PLR_k can correctly output the answer.

Since PLR_k knows $f_2, \ldots, f_{k-1}$, she can compute $h := f_{k-1} \circ f_{k-2} \circ \cdots \circ f_2 \in [n]^n$. For every $1 \le i \le n$, define

$$P_i = h^{-1}(i) := \{ j \mid h(j) = i; 1 \le j \le n \} \quad \text{and} \quad \alpha_i = |P_i|.$$

Intuitively, P_i can be seen as the set of vertices in the second layer that, following the directed edges, can reach the ith vertex in the kth layer. Denote

$$T = \{i \mid \alpha_i \ge 1; 1 \le i \le n\}.$$

Thus T is the image set of h, which can also be regarded as the set of vertices in the kth layer that are reachable from some vertex in the second layer.

Remember that the message sent by PLR_1, denoted by R, is the number of 1's in $g_1 = f_k \circ h \in \{0,1\}^n$. We have

$$\begin{aligned} R &= \left| \{ j \mid (f_k \circ h)(j) = 1; 1 \le j \le n \} \right| \\ &= \sum_{i : f_k(i) = 1; 1 \le i \le n} \left| \{ j \mid h(j) = i; 1 \le j \le n \} \right| \\ &= \sum_{i : f_k(i) = 1; 1 \le i \le n} \alpha_i \\ &= \sum_{i \in T : f_k(i) = 1} \alpha_i \\ &= \sum_{i \in T} f_k(i) \alpha_i . \end{aligned}$$

Recall that PLR_k knows T and all α_i's, but does not know f_k. Regarding $\{f_k(i) \mid i \in T\}$ as the set of unknowns (with value 0/1), let PLR_k try to solve the following equation:

$$\sum_{i\in T}\alpha_i f_k(i)=R. \tag{1}$$

PLR$_k$ knows all the coefficients in Eq. (1). In addition, she knows the chain containing f_k, say C, which is written by PLR$_{k-1}$. Thus, we can put an additional requirement on Eq. (1), stating that the solution must be consistent with some string in C, that is, there exists some string $s\in C$ such that $f_k(i)=s(i)$ for all $i\in T$. We know that there exists at least one solution to the equation (which is the restriction of f_k on T). We will show that there can be at most one solution to the equation. This will complete the proof of Theorem 2, since PLR$_k$ can just exhaustively search for the unique solution $\{f_k(i)\mid i\in T\}$. As she also sees f_1, she can compute the correct answer

$$(f_k\circ f_{k-1}\circ\cdots\circ f_2)(f_1)=f_k\bigl(h(f_1)\bigr),$$

using the fact that $h(f_1)\in T$. (Note that PLR$_i$ might not uniquely determine $f_k(i)$ for $i\notin T$; nonetheless, these values are useless to her.)

We now prove the uniqueness of the solution of Eq. (1). Assume to the contrary that there exist two distinct solutions, say f' and f'', to Eq. (1). Assume without loss of generality that $f'(r)=0$ and $f''(r)=1$ for some $r\in T$. Since both f' and f'' are consistent with some string in the chain C, by the definition of a chain we have

$$f'(i)\le f''(i)\quad \text{for all } i\in T.$$

Therefore,

$$\begin{aligned}0&=R-R\\&=\sum_{i\in T}\alpha_i f''(i)-\sum_{i\in T}\alpha_i f'(i)\\&=\sum_{i\in T}\alpha_i\bigl(f''(i)-f'(i)\bigr)\\&\ge\alpha_r\\&\ge 1,\end{aligned}$$

which is a contradiction. Hence, Eq. (1) can have at most one solution. Since we know that it has at least one solution, the claim is proved and Theorem 2 follows. □

4 Lower Bound for Restricted Collapsing Protocols

Note that in our protocol CHAIN, the first player needs to send $\log n$ bits of information. It is natural to ask whether the $\log n$ bits communicated by PLR$_1$ is necessary for achieving a low maximum communication complexity. In this part we make progress towards this question by proving the following theorem.

Theorem 3 *In any collapsing protocol for* MPJ_3^n, *if* PLR_1 *sends only one bit*, *then* PLR_2 *must communicate at least* $n-2$ *bits*.

We will utilize tools from Boolean function analysis to prove the theorem. For a function $f:\{0,1\}^n \to \{0,1\}$, the *total influence* of f, denoted by $I(f)$, is defined as:

$$I(f)=\frac{1}{2^n}\sum_{x\in\{0,1\}^n}\big|\{y\in\{0,1\}^n \mid y \text{ is a neighbor of } x \text{ and } f(y)\neq f(x)\}\big|,$$

where y is a neighbor of x if and only if x and y differ in exactly one bit. Clearly $I(f)\leq n$. It is well known that $I(f)=n$ if and only if f is the PARITY function or the complement of the PARITY function. See, e.g., [23, 24] for some basic results in Boolean function analysis.

We now prove Theorem 3.

Proof of Theorem 3 The proof is by a reduction from the two-party communication problem INDEX. In the problem INDEX^n, player 1 sees a string $s\in\{0,1\}^n$ and player 2 sees an integer $p\in[n]$. In a protocol for INDEX^n, player 1 first sends a message to player 2, and then player 2 needs to correctly output $s(p)$. It is well-known that in any deterministic protocol for INDEX^n, player 1 needs to send at least n bits (see e.g. [20]).

Consider any collapsing protocol for MPJ_3^n in which PLR_1 sends one bit. What PLR_1 sees is $g_1=f_3\circ f_2\in\{0,1\}^n$. Thus, the behavior of PLR_1 can be viewed as a function $h:\{0,1\}^n\to\{0,1\}$, which means that, upon seeing the string (composed function) $g_1\in\{0,1\}^n$, she writes one bit $h(g_1)$ on the blackboard.

Now consider $I(h)$, the total influence of the Boolean function h. We distinguish two cases.

- *Case 1:* $I(h)<n$. In this case we know that there exists an index $i\in[n]$ and a set of binary values $\{a_j\in\{0,1\}\mid j\in[n]\setminus\{i\}\}$, such that $h(a_1a_2\dots a_{i-1}0a_{i+1}\dots a_n)=h(a_1a_2\dots a_{i-1}1a_{i+1}\dots a_n)$. In other words, if the jth input bit of h is fixed to a_j for every $j\in[n]\setminus\{i\}$, the function h will degenerate to a constant function, i.e., it does not depend on the ith input bit (even if this input bit is not fixed).

  We now present a reduction from INDEX^{n-2} to MPJ_3^n. Let $\mathcal{I}$ be an instance of INDEX^{n-2} in which player 1 sees $s\in\{0,1\}^{n-2}$ and player 2 sees $p\in[n-2]$. Create an instance (f_1,f_2,f_3) of MPJ_3^n as follows:

  Let $f_1=i$ (recall that i is the "irrelevant" index of h mentioned before); let

$$f_3(j)=\begin{cases} s(j) & \text{if } 1\leq j\leq n-2;\\ 0 & \text{if } j=n-1;\\ 1 & \text{if } j=n,\end{cases}$$

  and

$$f_2(j)=\begin{cases} p & \text{if } j=i;\\ n-1 & \text{if } j\in[n]\setminus\{i\} \text{ and } a_j=0;\\ n & \text{if } j\in[n]\setminus\{i\} \text{ and } a_j=1.\end{cases}$$

It is easy to check that $f_3(f_2(f_1)) = s(p)$, and that for every $j \in [n] \setminus \{i\}$, $g_1(j) = f_3(f_2(j)) = a_j$. Hence, as argued before, h is fixed as a constant function regardless of the value of $g_1(i)$, and thus the message of PLR_1 is a fixed bit regardless of p and s. Therefore, both players in INDEX^{n-2} knows this message. (The knowledge about h can be regarded as known to both players, since we are considering a *fixed* protocol for MPJ_3^n.) Notice that player 1 in INDEX^{n-2} also knows f_1 and f_3, and player 2 knows f_1 and f_2. Thus they can simulate the collapsing protocol for MPJ_3^n, with player 1 emulating PLR_2, and player 2 emulating PLR_3. Since $f_3(f_2(f_1)) = s(p)$, player 2 will correctly output the answer. The number of bits that player 1 communicates is exactly that sent by PLR_2 in the protocol for MPJ_3^n. However, we know that in any protocol for INDEX^{n-2}, player 1 must send at least $n-2$ bits. Therefore, in the protocol for MPJ_3^n, PLR_2 must also send at least $n-2$ bits.

- *Case 2*: $I(h) = n$. In this case h is either PARITY (i.e., the exclusive OR of all its input bits) or its complement. We assume that h is PARITY, and the other case is very similar to prove. We perform a similar reduction as in Case 1. Further consider two subcases:
  - *n is even*. The reduction is from INDEX^n to MPJ_3^n, where $f_1 = 1$, $f_3 = s$, and $f_2(j) = p$ for all $j \in [n]$. It is easy to check that $f_3(f_2(f_1)) = s(p)$, and that h becomes a constant function (since it is the XOR of an even number of identical bits). By a similar analysis as in Case 1, we know that in any collapsing protocol for MPJ_3^n, PLR_2 must communicate at least n bits.
  - *n is odd*. The reduction is from INDEX^{n-1} to MPJ_3^n, where $f_1 = 1$,

$$f_3(j) = \begin{cases} s(j) & \text{if } 1 \le j \le n-1; \\ 0 & \text{if } j = n, \end{cases}$$

and

$$f_2(j) = \begin{cases} p & \text{if } j \in \{1, 2\}; \\ n & \text{if } j \in [n] \setminus \{1, 2\}. \end{cases}$$

It is easy to check that $f_3(f_2(f_1)) = s(p)$, and that h becomes a constant function (since it is the XOR of two $s(p)$'s and $n-2$ zeros). By a similar analysis as in Case 1, we know that in any collapsing protocol for MPJ_3^n, PLR_2 must communicate at least $n-1$ bits.

The proof of Theorem 3 is thus complete. □

5 Conclusions

In this paper, we have investigated collapsing protocols for the pointer jumping problem under the NOF multiparty computation model. We presented a collapsing protocol for the problem with maximum communication complexity $n - \frac{1}{2}\log n + 1$, which matches the lower bound given in [11] up to an additive constant. We also prove that in any collapsing protocol for MPJ_3^n, if the first player only sends one bit, then the second player must send at least $n-2$ bits.

There are many problems in this area that deserve further explorations. A big open question is whether the gap between upper and lower bounds for the total communication complexity of MPJ_k^n can be bridged, even for small values of k. Another question is to strengthen the result of Theorem 3 by allowing the first player to send more bits.

It is also interesting to consider randomized collapsing protocols for MPJ_k^n. The following public-coin randomized collapsing protocol for MPJ_k^n was suggested by an anonymous reviewer of an early draft of this paper. First, the players use the public coins to randomly select $k-1$ subsets of $[n]$ of size $2n/k$, say $S_1, S_2, \ldots, S_{k-1}$. Then, for each $i \in [k-1]$, PLR_i sends $g_i(j)$ for all $j \in S_i$. Note that each of the first $k-1$ players sends $2n/k$ bits. For each $i \in [k-1]$, it holds with probability $2/k$ that $(f_i \circ \cdots \circ f_2)(f_1) \in S_i$. Thus, with probability at least $1-(1-2/k)^{k-1} > 0.7$, there exists $i' \in [k-1]$ such that $(f_{i'} \circ \cdots \circ f_2)(f_1) \in S_{i'}$. Then PLR_k, who knows $f_{i'}, \ldots, f_1$, can obtain the correct answer via $\text{PLR}_{i'}$'s message. By increasing the size of the chosen subsets, we can get a collapsing protocol with maximum communication complexity $\Theta(n/k)$ that succeeds with probability at least $1-\epsilon$ for any fixed $\epsilon > 0$. This shows that randomness does help in terms of maximum communication complexity. However, the total communication complexity of the protocol is still $\Theta(n)$. This motivates the following question: is there a randomized collapsing protocol for MPJ_k^n with total communication complexity $o(n)$?

Acknowledgement We would like to thank Joshua Brody for valuable discussions and suggestions on this problem. We are also grateful to the anonymous reviewers for their helpful comments on improving the quality and presentation of this paper, and in particular to an anonymous reviewer who suggested the randomized collapsing protocol for MPJ_k^n.

References

1. Ablayev, F.: Lower bounds for one-way probabilistic communication complexity and their application to space complexity. Theor. Comput. Sci. **175**(2), 139–159 (1996)
2. Ajtai, M.: A lower bound for finding predecessors in Yao's cell probe model. Combinatorica **8**(3), 235–247 (1988)
3. Alon, N., Matias, Y., Szegedy, M.: The space complexity of approximating the frequency moments. J. Comput. Syst. Sci. **58**(1), 137–147 (1999)
4. Andoni, A., Indyk, P., Krauthgamer, R.: Earth mover distance over high-dimensional spaces. In: Proc. 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 343–352 (2008)
5. Babai, L., Hayes, T.P., Kimme, P.G.: The cost of the missing bit: communication complexity with help. Combinatorica **21**(4), 455–488 (2001)
6. Beame, P., Pitassi, T., Segerlind, N.: Lower bounds for Lovász-Schrijver systems and beyond follow from multiparty communication complexity. In: Proc. 32nd International Colloquium on Automata, Languages and Programming (ICALP), pp. 1176–1188 (2005)
7. Beigel, R., Tarui, J.: On ACC. Comput. Complex. **4**, 350–366 (1994)
8. Blais, E., Brody, J., Matulef, K.: Property testing lower bounds via communication complexity. In: Proc. 26th IEEE Conference on Computational Complexity (CCC), pp. 210–220 (2011)
9. Brody, J.: The maximum communication complexity of multi-party pointer jumping. In: Proc. 24th IEEE Conference on Computational Complexity (CCC), pp. 379–386 (2009)
10. Brody, J.: Some communication complexity results and their applications. PhD thesis, Dartmouth College (2010)
11. Brody, J., Chakrabarti, A.: Sublinear communication protocols for multi-party pointer jumping and a related lower bound. In: Proc. 25th Annual Symposium on Theoretical Aspects of Computer Science (STACS), pp. 145–156 (2008)

12. Chakrabarti, A., Jayram, T.S., Pătraşcu, M.: Tight lower bounds for selection in randomly ordered streams. In: Proc. 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA) (2008)
13. Chandra, A.K., Furst, M.L., Lipton, R.J.: Multi-party protocols. In: Proc. 15th ACM Symposium on Theory of Computing (STOC), pp. 94–99 (1983)
14. Damm, C., Jukna, S., Sgall, J.: Some bounds on multiparty communication complexity of pointer jumping. Comput. Complex. **7**(2), 109–127 (1998)
15. Engel, K.: Sperner Theory. LNCS. Springer, Berlin (2001)
16. Graham, R.L., Knuth, D.E., Patashnik, O.: Concrete Mathematics. Addison–Wesley, Reading (1994)
17. Gronemeier, A.: NOF-multiparty information complexity bounds for pointer jumping. In: Proc. 31st International Symposium on Mathematical Foundations of Computer Science (MFCS) (2006)
18. Guha, S., McGregor, A.: Lower bounds for quantile estimation in random-order and multi-pass streaming. In: Proc. 34th International Colloquium on Automata, Languages and Programming (ICALP), pp. 704–715 (2007)
19. Karchmer, M., Wigderson, A.: Monotone circuits for connectivity require super-logarithmic depth. In: Proc. 20th ACM Symposium on Theory of Computing (STOC), pp. 539–550 (1988)
20. Kushilevitz, E., Nisan, N.: Communication Complexity. Cambridge University Press, Cambridge (1996)
21. Bro Miltersen, P.: Lower bounds for union-split-find related problems on random access machines. In: Proc. 26th ACM Symposium on Theory of Computing (STOC), pp. 625–634 (1994)
22. Nisan, N., Wigderson, A.: Rounds in communication complexity revisited. SIAM J. Comput. **22**(1), 211–219 (1993)
23. O'Donnell, R.: Analysis of Boolean Functions. Lecture Notes. http://www.cs.cmu.edu/~odonnell/boolean-analysis/
24. O'Donnell, R.: Some topics in analysis of boolean functions. In: Proc. 40th Annual ACM Symposium on Theory of Computing (STOC) (2008)
25. Pudlák, P., Rödl, V., Sgall, J.: Boolean circuits, tensor ranks and communication complexity. SIAM J. Comput. **26**(3), 605–633 (1997)
26. Viola, E., Wigderson, A.: One-way multi-party communication lower bound for pointer jumping with applications. In: Proc. 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 427–437 (2007)
27. Yao, A.C.: Some complexity questions related to distributed computing. In: Proc. 11th ACM Symposium on Theory of Computing (STOC), pp. 209–213 (1979)