Reconstructing Markov processes from independent and anonymous experiments

Discrete Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Reconstructing Markov processes from independent and anonymous experiments

Silvio Micali, Zeyuan Allen Zhu *

MIT CSAIL, United States

ARTICLE INFO

Article history:
Received 21 June 2014
Received in revised form 7 June 2015
Accepted 21 June 2015
Available online xxxx

Keywords:
Graph reconstruction
Random walk
Markov process
Local algorithms

ABSTRACT

We investigate the problem of *exactly* reconstructing, with high confidence and up to isomorphism, the ball of radius r centered at the starting state of a Markov process from *independent* and *anonymous* experiments. In an anonymous experiment, the states are visited according to the underlying transition probabilities, but no global state names are known: one can only recognize whether two states, *reached within the same experiment*, are the same.

We prove quite tight bounds for such exact reconstruction in terms of both the number of experiments and their lengths.

1. Introduction

The problem of reconstructing a large "object" from partial observations is quite fundamental, and arises in many fields, such as system biology [22,29], social networks [38,27], brain networks [36,19], telecommunication networks [10], and many others.

We investigate a more complex type of reconstruction. In essence, our goal is to reconstruct a Markov process from the records produced by *limited* observers acting *independently*, without coordination, and without even sharing a common "name space". Let us explain.

1.1. Our model

Our Markov model. In a Markov process, we denote the underlying transition graph by $G = (V, E)$ and the starting vertex by v. In this paper, the graph G is undirected and has infinitely many vertices, each of finite degree. An infinite sequence of vertices is generated by the following process. The first vertex is v, and, if the ith vertex is u, then the $(i + 1)$-st vertex is chosen at random uniformly and independently among the neighbors of u.

A sequence of vertices so generated is called a *random walk*. If $(v =)v_0 \to v_1 \to \cdots$ is a random walk, then $v_0 \to \cdots \to v_\ell$ is a *random walk of length* ℓ.

Note. Assuming that G is undirected and unweighted allows us to present our results in the cleanest way. We shall discuss how to relax both assumptions in Section 1.4. Assuming that G has infinitely many vertices is a simple way to force us to consider only "local" algorithms: essentially, algorithms whose performance does not depend on the size of the whole graph, which may be larger than all the parameters we shall care about.

* Corresponding author.
E-mail addresses: silvio@csail.mit.edu (S. Micali), zeyuan@csail.mit.edu (Z.A. Zhu).

http://dx.doi.org/10.1016/j.dam.2015.06.035

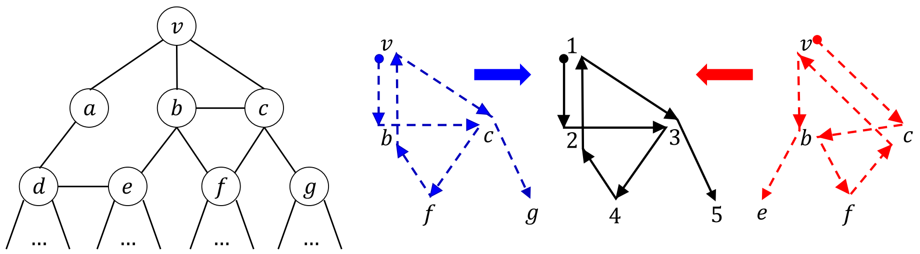


Fig. 1. An example illustrating the definition of *anonymous experiment*.

Our anonymous observation model. If $v_0 \to v_1 \to \cdots$ is a random walk, then its corresponding *(anonymous) experiment* is the sequence of integers $f(v_0) \to f(v_1) \to \cdots$, where $f(v_i) \stackrel{\text{def}}{=} |\{v_0, \ldots, v_{i^*}\}|$ and i^* is the smallest integer j such that $v_j = v_i$. Intuitively, $f(u)$ maps u to an integer indicating that u is the $f(u)$th distinct vertex in this walk.

Example. In the graph of Fig. 1, the length-7 walk $v \to b \to c \to f \to b \to v \to c \to g$ corresponds to the anonymous experiment $1 \to 2 \to 3 \to 4 \to 2 \to 1 \to 3 \to 5$.

Note the walk $v \to c \to b \to f \to c \to v \to b \to e$ *also* corresponds to the same experiment.

Rationale. Markov processes naturally model physical systems. In essence, the possible "states" of the system are the vertices of the transition graph G, and, when put in its "initial state" v, the system evolves (i.e., new states are generated) according to the transition rules.

When the system is "new" – better said, studied for the first time – no one initially has any idea about the underlying graph G. However, each individual can, on his own, experiment with the system by putting it into its initial state, and independently observe its "evolution": that is, a random walk in G.

In sum, each individual will observe and record the states encountered in a random walk. Since the system is new, no global names exist for the states. Thus each individual may very well use his own name space for the states encountered, and thus his record is an anonymous experiment as defined above.

Of course, an individual observer might consider writing down a full description of every state he sees. However, this may not be possible due to – say – memory limitations [15], or privacy reasons [26]. Also, an observer may not know how many details are sufficient to identify each encountered state. In any case, an anonymous experiment is a most compact and meaningful record.

1.2. *Our results for the basic reconstruction problem*

Whether human or not, a realistic observer has a bounded lifetime, and thus cannot visit more than ℓ nodes in his random walk.[1] Thus, even with an unlimited number of such observers, one can at most learn $\mathcal{D}_\ell$, the distribution over the anonymous experiments of length ℓ (that are induced from the random walks of the same length). Since our G has infinitely many vertices, $\mathcal{D}_\ell$ cannot suffice to reconstruct the entire graph G. However, one may be able to use $\mathcal{D}_\ell$ in order to reconstruct $B(v, r)$, the *ball of center v and radius r* (i.e., the subgraph of G induced by all vertices whose distance from v is at most r). More precisely, one may be able to compute a graph $G' = (V', E')$ and a distinguished vertex $v' \in V'$, such that G' is isomorphic to $B(v, r)$ and the isomorphism maps v' to v. Thus, our basic reconstruction problem can be formulated as follows:

For every r, is there a length ℓ such that $B(v, r)$ is reconstructible (up to isomorphism) from $\mathcal{D}_\ell$?

Notice that, given access to the distribution $\mathcal{D}_\ell$, one can also simulate access to the distributions $\mathcal{D}_1, \ldots, \mathcal{D}_{\ell-1}$. Of course, although for now we are ignoring the complexity of learning these distributions, it would be nice if, given $(\mathcal{D}_1, \ldots, \mathcal{D}_\ell)$ as *oracles*, the reconstruction algorithm is efficient. Here, we say that $(\mathcal{D}_1, \ldots, \mathcal{D}_\ell)$ are given as oracles, if the algorithm is allowed to ask for the precise probability of $\mathcal{D}_t(P)$ for any anonymous experiment P of length $t \in [\ell]$.

Notice too that, in principle, our basic reconstruction problem may be impossible. For instance, could there exist two different Markov processes, (G_1, v_1) and (G_2, v_2), having the same distribution $\mathcal{D}_\ell$ for all $\ell \geq 0$? If this were the case, the two processes would be indistinguishable by any number of anonymous experiments, of any length, which immediately implies a negative answer to the above question. Yet, we provide a constructive proof showing that our basic reconstruction problem is indeed possible, when the underlying graph G is undirected.

Theorem 1. *Let n be the number of vertices in $B(v, r)$ and m the number of edges. One can reconstruct $B(v, r)$ in time $O(n^2)$ and with $O(n^2)$ oracle accesses to $(\mathcal{D}_1, \ldots, \mathcal{D}_\ell)$, where $\ell = O(m)$. Moreover, the reconstruction algorithm only makes membership queries to* $\mathrm{supp}(\mathcal{D}_i)$ *for $i \in [\ell]$.*

In contrast, as we shall see in Section 1.4, this reconstruction becomes impossible when the underlying graph is directed but not strongly connected.

Is this algorithm tight? To answer this question we must refine our reconstruction problem.

[1] For concreteness, if he lives for at most 100 years, and each transition from node to node takes 1 s of time, then $\ell = 100 \times 366 \times 24 \times 60 \times 60$.

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035

1.3. Our results for the refined reconstruction problem

To discuss efficiency more meaningfully, we need to identify the relevant resources. First of all, notice that one may never learn $\mathcal{D}_\ell$ exactly, no matter how many anonymous experiments he may observe. Thus, we should investigate the "complexity" of reconstructing $B(v, r)$ with some *confidence* $1-\delta$. Furthermore, to appropriately count resources, we should realize that even approximating the result of a single oracle call to $\mathcal{D}_\ell$ (i.e., to approximately compute the probability of a given anonymous experiment P of length ℓ, up to a constant factor and with constant probability), one needs an exponential number of length-ℓ experiments.

This said, it is easy to see that our first result can be expressed more precisely as follows.

Theorem 1′. *Let d be the maximum degree of graph G. Then, $B(v, r)$ can be reconstructed with probability at least $1-\delta$ using $d^{O(d^r)}\log\frac{1}{\delta}$ anonymous experiments of length $O(d^r)$ each.*

The above theorem follows from Theorem 1 because m, the number of edges in $B(v, r)$, is at most $O(d^r)$ due to the bounded degree, and learning the support of $\mathcal{D}_\ell$ with $1-\delta$ probability for $\ell = O(m)$ requires $d^{O(m)}\log\frac{1}{\delta}$ samples.

Thus a natural question arises: can one improve the total number of experiments, or the length of experiments, even when the underlying graph is a tree? More concretely, suppose that $B(v, r)$ is a tree of degree at most d and depth r, and d is a constant.

- A random experiment of length $\ell = r$ can already visit (with non-zero probability) every vertex in $B(v, r)$; if one allows $\ell = \mathsf{poly}(r)$, there will be $d^{\mathsf{poly}(r)}$ distinct experiments of length ℓ that correspond to random walks within $B(v, r)$. In principle, one may hope to use the probability values of these $d^{\mathsf{poly}(r)}$ experiments from $\mathcal{D}_\ell$ (that has bit complexity at least $d^{\mathsf{poly}(r)}$), to reconstruct $B(v, r)$ (that has bit complexity only $d^{O(r)}$); could it be possible?
- Our algorithm in Theorem 1 makes oracle accesses to $\mathcal{D}_\ell$, and as we have argued, supporting even a single such query requires us to generate $d^{O(\ell)}$ random experiments. Therefore, could it be possible to design other types of algorithms that use significantly smaller number of experiments?

We prove that the answers are both no in a very strong sense.[2]

Our first impossibility result states that one cannot "asymptotically" decrease the length of the experiments, even if the number of experiments is made arbitrary high.

Theorem 2. *If an algorithm can, for every Markov process (G, v), where G is an infinite binary tree, and every radius r, reconstruct $B(v, r)$ with probability $\frac{1}{2}$ using an arbitrary number of anonymous experiments of length no more than ℓ, then $\ell = 2^{\Omega(r)}$.*

Our second result is similarly strong, namely, one cannot "asymptotically" decrease the number of the experiments, even if the length of experiments is made arbitrary high.

Theorem 3. *If an algorithm can, for every Markov process (G, v), where G is an infinite ternary tree, and every radius r, reconstruct $B(v, r)$ with probability $\frac{1}{2}$ using N anonymous experiments of arbitrary lengths, then $N = 2^{2^{\Omega(r)}}$.*

1.4. Extensions and additional results: a quick summary

As we shall discuss in our Related Work section (Section 1.5), our approach is related but quite different from other types of reconstruction problems studied before. Here we wish to sketch various ways to generalize/improve our results.

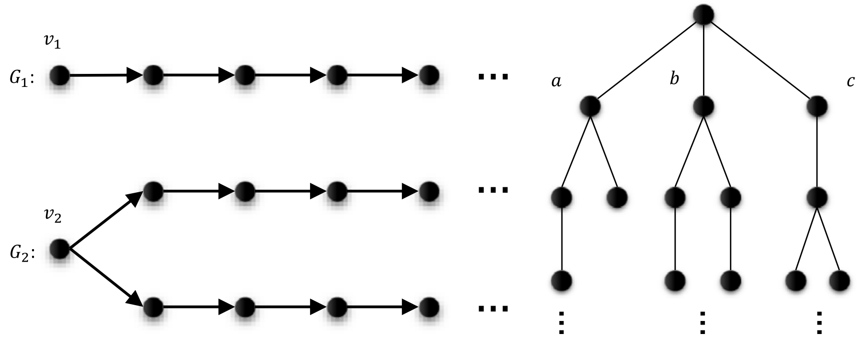
Extensions. It should be realized that in a typical Markov process, the underlying graph may be *directed* and/or *weighted.* Let us explore both possibilities separately.

An undirected graph of course is a special case of a directed one: namely a graph in which for each edge $x \to y$ there also is an edge $y \to x$. For the reconstruction problem we discuss, however, the undirected case captures all the difficulty of the problems, and certainly allows for much simplicity. For instance, the impossibility result of Theorem 2 becomes trivial. To see this, it is enough to consider the following two graphs G_1, G_2 in Fig. 2a (with starting vertices v_1, v_2 respectively). Indeed, for both graphs, there is only one anonymous experiment of length ℓ: namely, $1 \to 2 \to \cdots \to \ell + 1$.

Accordingly, our reconstruction problem becomes interesting only when the underlying graph is *strongly connected.* Better said, since we are studying infinite graphs and "local algorithms", the notion of strong connectivity needs to be strengthened so as to guarantee, for every edge $x \to y$, the existence of a path from y back to x of suitably bounded length. In the simplest case, the length of the path from y to x is upper bounded by an absolute constant c. In this case, our algorithm of Theorem 1 can be extended to reconstruct the directed ball $B(v, r)$ using experiments of length $\ell = O(m \cdot c)$. More generally, our algorithm will work with experiments of length $\ell = O(m \cdot c')$, where c' is the average length of the paths from y to x over all edges $x \to y$ in the ball $B(v, r)$, and c' need not be known by the algorithm.

[2] We also note that the answers are both yes in certain special cases, as we shall formalize in Section 1.4.

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035



(a) An example for directed graphs.

(b) An example to illustrate homogeneity.

Fig. 2. Examples for extensions and improvements.

In the weighted case, our algorithm will reconstruct the "topology" of the underlying ball, that is, all edges in the ball, without their weights. This implies, for instance, if the random walk we studied has laziness – that is, at each vertex it stays at where it is with half probability, and goes to a random neighbor with another half probability – we can still reconstruct G. In general, reconstructing the weights too will require future work.

Improvements. The performance of our algorithm of Theorem 1 can be dramatically improved given reasonable guarantees about the topology of the underlying graph. One such guarantee is "non homogeneity". Consider the simple graph (indeed a tree) in Fig. 2b.

In this graph, the three children of the root, a, b, and c, cannot be distinguished until level 3. Indeed, all of them are indistinguishable at level 1, that is, in $B(v, 1)$. Vertex c can be distinguished from the others at level 2: indeed, in $B(v, 2)$ vertex c has only one child (equivalently 2 neighbors), while each of a and b has 2 children. At this level, however, no way exists to distinguish a from b. But one additional level suffices.

Informally, we say that a graph G has *homogeneity* ω, if for each vertex u in G, every two neighbors of u can be distinguished in a ball centered at u with at most ω edges. Then, if the graph G is guaranteed to be of homogeneity ω, the algorithm for Theorem 1 (without knowing ω) can be extended to reconstruct $B(v, r)$ with experiments only of length $\ell = O(r \cdot \omega)$.

Notice that this specific improvement does not contradict the impossibility result of Theorem 2. Indeed, to prove Theorem 2 we exhibit a ball $B(v, r)$ whose homogeneity is very large, namely, $\omega = 2^{\Omega(r)}$. In fact, $B(v, r)$ is constructed so that $B(v, r - 1)$ consists of a complete binary tree, and thus the two children of the root cannot be distinguished up to level $r - 1$.

1.5. Related work

Graph reconstruction using queries. The problem of reconstructing an unknown graph from oracle queries has been studied in many different contexts, and most notably using edge detection queries [16,2,1,5,6], edge counting queries [17,7,25], or distance queries [20,21,32,24].

In an edge detection query model, the oracle, on input a subset S of the vertices, returns if there exists an edge between any two vertices in S. Angluin and Chen [6] show that using $O(\log n)$ adaptive queries per edge is sufficient for reconstructing an arbitrary graph, and this has been generalized to hypergraphs [5].

In an edge counting query model, the oracle, on input a subset S of the vertices, returns the number of edges between any two vertices in S. While Grebinski and Kucherov [17] prove tight bounds of $O(dn)$ and $O(n^2/\log n)$ non-adaptive queries for d-degree-bounded and general graphs, in a more recent work, Mazzawi [25] shows that an information-theoretically tight bound of $O(m \log(n^2/m)/\log m)$ can be achieved using non-adaptive queries for any graph with n vertices and m edges.

In a distance query model, the supported queries are of the form $dist(u, v)$, that is, the oracle returns the (possibly approximate) distance between any two given vertices. A lower bound of $\Omega(n^2)$ queries is shown by Reyzin and Srivastava [32] for general graphs. Mathieu and Zhou [24] generalize this lower bound to allow approximate distance oracles, provide an upper bound of $\widetilde{O}(n^{3/2})$ for constant-degree graphs, and $\widetilde{O}(n)$ for outerplanar graphs.

All the results above are quite different from ours: the "name space" of the vertices are shared between different queries. As a result, if one is satisfied with a polynomial running time – say, $O(n^2)$ – it is trivial to (even locally) reconstruct any graph using any of the oracles above.

Learning graphical models. Much work has been done in the machine learning community on learning the structures of graphical models. While we refer interested readers to Part III of Kollar and Friedman's book [23], we summarize a few of them below.

A first type of research in this field assumes that the topology of a graphical model (e.g., a Bayesian network) is known, and focuses on estimating the parameters in the model. Two well-known methods are the maximum likelihood estimation and the Bayesian estimation. While the earliest application of such methods is for the purpose of classification [12], Spiegelhalter and Lauritzen [34] lay the foundation for the general parameter estimation problem of the Bayesian network from data.

A second type of research is on structure learning in Bayesian networks. Two main approaches are used to learn a Bayesian network without a pre-specified topological structure. The first approach is constraint-based, where the network structure is reconstructed via various kinds of independency tests. The most famous one is the I-equivalence test due to Verma and Pearl [37,30]. The second approach is learning-score-based, where the score of a candidate network structure is based on (1) a prior distribution describing how well we believe the structure is true (where one usually prefers sparser structures), and (2) how well the structure fits the observed data. For instance, Buntine [8] first explored the use of non-uniform priors (namely, Dirichlet priors), and studied the problem of learning discrete Bayesian networks by Bayesian scores.

In general, structure learning beyond Bayesian networks is much harder. For instance, learning Markov random fields (i.e., Markov networks) has received much attention in various special cases including, most notably, the tree case tracing back to Chow and Liu [9].

A third type of research is on learning with incomplete data, where some variables are usually missing or hidden in the data, and one may still want to either estimate the parameters or learn the network structure as above. Perhaps the most famous method in this line of research is the framework of expectation maximization, introduced by Dempster et al. [11], and then applied to structural learning by Friedman [13,14].

To the best of our knowledge, there has not been any study of using independent and anonymous samples, like we do, in order to reconstruct (or in their language, do structure learning for) a Markov process. In our case, the "name space" of the observed variables are not shared across samples.

Local graph algorithms. By adaptively performing a sublinear number of queries (compared to the input graph size) around a specific vertex, one can solve certain local variants of the classical graph problems, including coloring, maximal independent set, dominating set, and many others. One can find a survey for such topics in [33]. The idea of using local random walks to obtain local properties about a graph is studied by papers such as [35,3,4,39,28], but those random walks assume a global name space.

A complementary model of limited experiment. Another type of limited experiment is that in which, although a global name space exists, each observer only writes down the *set* of the visited states, ignoring the order in which they are visited. See [31,18].

1.6. Roadmap

We introduce necessary notations in Section 2, and prove Theorems 1, 2 and 3 in Sections 3, 4 and 5 respectively.

2. Notation

Throughout the paper we assume that the graph $G = (V, E)$ is undirected, unweighted, simple,[3] of finite degree, and of infinite size. We denote by $B^G(u, r)$ the subgraph of G induced by all vertices whose distance from u is at most r. When it is clear from the context, we abbreviate $B^G(u, r)$ as $B(u, r)$.

We distinguish a special node $v \in V$ as the *starting vertex*, and are interested in reconstructing $B(v, r)$ for some radius r.

Anonymous experiments. We say that P is a valid *(anonymous) experiment* of length ℓ if $P = p_0 \to p_1 \to \cdots p_\ell$, in which $p_i = |\{p_0, p_1, \ldots, p_{i^*}\}|$ where i^* is the smallest index j such that $p_j = p_i$. For instance, $1 \to 2 \to 3 \to 2 \to 4 \to 3$ is a valid experiment, but $1 \to 2 \to 4$ is not.

A *walk* Q of length ℓ in G corresponds to a unique anonymous experiment P of the same length, by replacing the name of each vertex u in Q by a positive integer k indicating that u is the kth distinct vertex in this walk. We denote by $Q \lhd (G, P)$ when this happens.

In the graph of Fig. 1, the length-7 walk $Q = v \to b \to c \to f \to b \to v \to c \to g$ uniquely corresponds to the experiment $P = 1 \to 2 \to 3 \to 4 \to 2 \to 1 \to 3 \to 5$; as we have seen, two different walks may correspond to the same experiment.

For notational simplicity, we sometimes remove the requirement that "the numbers in an experiment are sorted by the order they visit". For instance, we may also talk about experiments of the form $2 \to 3 \to 1 \to 3$, which is regarded equivalent to $1 \to 2 \to 3 \to 2$ due to renaming.

Random walks. A *random walk* of length ℓ on G (starting from vertex v) is generated from the following procedure. The first vertex is v, and, if the ith vertex is u, then the $(i + 1)$-st vertex is chosen at random uniformly and independently among the neighbors of u.[4] A *random (anonymous) experiment* is generated by first generating a random walk and then mapping it to its corresponding anonymous experiment.

We let $\mathcal{D}^{\text{walk}}_{v,\ell}$ denote the distribution over random walks of length ℓ, and $\mathcal{D}_{v,\ell}$ the distribution over random anonymous experiments of length ℓ. For notational simplicity we usually denote them by $\mathcal{D}_\ell$ and $\mathcal{D}^{\text{walk}}_\ell$. We also let $\text{supp}(\mathcal{D}_\ell)$ be the support of $\mathcal{D}_\ell$, so it consists of all experiments that have a positive probability to be seen in a random experiment of length ℓ.

[3] I.e., no multi-edges or self-loops. One can allow a single self-loop at each vertex for the positive result of this paper.

[4] One can also allow laziness, letting the random walk have a 1/2 probability of staying for each step, and all results in this paper remain unchanged.

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035

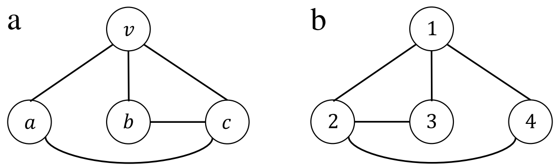


Fig. 3. An example to illustrate the proof of Theorem 1 for $r = 1$.

Supporting graph. Given an anonymous experiment P of length ℓ that contains n distinct integers, one can define its supporting graph $\texttt{Graph}(P) \stackrel{\text{def}}{=} (V, E)$, where $V = \{1, 2, \ldots, n\}$ and $(a, b) \in E$ if and only if $a \to b$ (or $b \to a$) appears in P. For instance, letting $P = 1 \to 2 \to 3 \to 1 \to 4 \to 2 \to 1$, we have $\texttt{Graph}(P)$ equal to the graph in Fig. 3b. As we shall see in detail, a usual property about supporting graphs is that given any $P \in \text{supp}(\mathcal{D}_{v,\ell})$, its supporting graph $\texttt{Graph}(P)$ is a subgraph of G (up to renaming of the vertices), where vertex 1 in $\texttt{Graph}(P)$ is mapped to v in G.

Path replacement. Given any experiment P, we denote by $\texttt{Replace}(P,\ u,\ P')$ the new experiment after replacing the last occurrence of integer u in P by the path P'. For instance

$$\texttt{Replace}(1 \to 2 \to 3 \to 4 \to 5 \to 4 \to 3,\ 4,\ 4 \to 6 \to 4) = 1 \to 2 \to 3 \to 4 \to 5 \to (4 \to 6 \to 4) \to 3$$

where the parentheses are for clarification purpose.

3. Theorem 1: a reconstructability result

In this section we show a positive result on reconstructing $B(v, r)$ from random anonymous experiments of length $\ell = O(m)$, where m is the number of edges in $B(v, r)$.

Theorem 1 (*Restated*)**.** *Let n be the number of vertices in $B(v, r)$ and m the number of edges.* $\texttt{Reconstruct}(v, r)$ *(see Fig. 4) reconstructs $B(v, r)$ with oracle accesses to $(\mathcal{D}_1, \ldots, \mathcal{D}_\ell)$, where $\ell = 2(m + 1)$. More specifically,* $\texttt{Reconstruct}$ *runs in time $O(n^2)$, and makes a total of $O(n^2)$ membership queries to* $\text{supp}(\mathcal{D}_i)$ *for $i \in [\ell]$.*

3.1. An intuitive and non-constructive proof of Theorem 1

In this subsection, we show why Theorem 1 holds in a rather "non-constructive" way, that is, without worrying about the running time of the reconstruction algorithm. In the next subsection we prove Theorem 1, with the claimed running time of its reconstruction algorithm.

The warm-up case: Reconstruction for $r = 1$. Before proving the theorem, let us build the intuition by studying the special case of $r = 1$. Consider the following simple 2-step algorithm for reconstructing $B(v, 1)$.

* (Throughout this section we slightly abuse the notation: for any experiment P of length no more than ℓ, we use $P \in \text{supp}(\mathcal{D}_\ell)$ to indicate the fact that $P \in \text{supp}(\mathcal{D}_i)$ for some $i \in [\ell]$.)

1. In the first step we learn the degree of v. Let $k \ge 1$ be the maximum integer such that the experiment

$$P = 1 \to 2 \to 1 \to 3 \to \cdots \to k \to 1$$

is in $\text{supp}(\mathcal{D}_\ell)$. It is easy to show that vertex v has precisely $k - 1$ neighbors in G according to the definition of k.

* (For the ease of describing the next step, we assume $k = 4$ and $B(v, 1)$ is given by Fig. 3a.)

2. In the second step we learn the pairwise connections among the $3 = k - 1$ neighbors of v. Letting $P = 1 \to 2 \to 1 \to 3 \to 1 \to 4 \to 1$ be the walk studied in the first step, we proceed as follows.
   - We first check if
     $$P_1 \stackrel{\text{def}}{=} \texttt{Replace}(P,\ 2,\ 2 \to 3 \to 2) = 1 \to (2 \to 3 \to 2) \to 1 \to 3 \to 1 \to 4 \to 1$$
     is in $\text{supp}(\mathcal{D}_\ell)$. If not, it indicates that there is no pairwise connection between *any* two neighbors of v, and the algorithm may terminate. Otherwise, there exists *at least* one pair of neighbors of v that are connected and the algorithm proceeds. Note that P_1 indeed exists in $\text{supp}(\mathcal{D}_\ell)$ for the graph of Fig. 3a, because $v \to a \to c \to a \to v \to c \to v \to b \to v$ is such a walk.
   - We then check if
     $$P_2 \stackrel{\text{def}}{=} \texttt{Replace}(P_1,\ 2,\ 2 \to 4 \to 2) = 1 \to \big(2 \to 3 \to (2 \to 4 \to 2)\big) \to 1 \to 3 \to 1 \to 4 \to 1$$
     is in $\text{supp}(\mathcal{D}_\ell)$. If not, it indicates that there does not exist a neighbor of v that is connected to two other neighbors, and the algorithm may terminate (in the case of $k = 4$). Otherwise, like in Fig. 3a where $v \to \big(c \to a \to (c \to b \to c)\big) \to v \to a \to v \to b \to v$ is such a walk, there exists a neighbor of v connected to two other neighbors, and the algorithm proceeds.
   - We finally check if $P_3 \stackrel{\text{def}}{=} \texttt{Replace}(P_2,\ 3,\ 3 \to 4 \to 3)$ is in $\text{supp}(\mathcal{D}_\ell)$. If not, like in Fig. 3a, we know the other two neighbors of v are not connected; otherwise they are connected. In both cases the algorithm may terminate here (in the case of $k = 4$).

Input: Membership access to $\text{supp}(\mathcal{D}_\ell)$, a starting vertex v and a radius r.
Output: A graph G' that is isomorphic to $B(v, r)$, and the isomorphism maps vertex 1 to v.

$P \leftarrow 1$.
for $r_0 \leftarrow 1$ **to** r **do**
  $P_0 \leftarrow P; G_0 \leftarrow \texttt{Graph}(P)$. ▷ G_0 is a reconstruction of $B(v, r_0 - 1)$
  $n_0 \leftarrow$ the number of vertices in G_0.
  $u_1, \dots, u_k \leftarrow$ the vertices in G_0 of distance precisely $r_0 - 1$ from vertex 1. ▷ $u_i \in [n_0]$
  for $i \leftarrow 1$ **to** k **do**
    $P' \leftarrow P_{i-1}$.
    $x \leftarrow$ the smallest integer not appearing in P'.
    while $\texttt{Replace}(P', u_i, u_i \to x \to u_i) \in \mathcal{D}_\ell$ **do**
      $P' \leftarrow \texttt{Replace}(P', u_i, u_i \to x \to u_i)$.
      for all $u' \in \{u_{i+1}, \dots, u_k\} \cup \{n_0 + 1, \dots, x - 1\}$ **do**
        if $\texttt{Replace}(P', x, x \to u' \to x) \in \mathcal{D}_\ell$ **then**
          $P' \leftarrow \texttt{Replace}(P', x, x \to u' \to x)$.
        end if
      end for
      $x \leftarrow$ the smallest integer not appearing in P'.
    end while
    $P_i \leftarrow P'$.
  end for
  $P \leftarrow P_k$.
end for
return $\texttt{Graph}(P)$.

Fig. 4. $\texttt{Reconstruct}^{\mathcal{D}_\ell}(v, r)$.

In the end of the algorithm, we output the supporting graph of the last experiment seen in $\text{supp}(\mathcal{D}_\ell)$ by the above steps. In our example, this is $\texttt{Graph}(P_2)$, shown in Fig. 3b. Note that Fig. 3b is isomorphic to Fig. 3a and the isomorphism maps vertex 1 to vertex v, so is indeed a reconstruction of $B(v, 1)$. In this example, the longest experiment ever queried is P_3, of length $12 = 2(m + 1) = \ell$.

The general case: Reconstruction for $r > 1$. One can learn from the above warm-up case that, for any experiment P of length no more than ℓ,

- if $P \in \text{supp}(\mathcal{D}_\ell)$, then $\texttt{Graph}(P)$ is a subgraph of G (up to renaming with 1 being mapped to v in G), and conversely
- if $\texttt{Graph}(P)$ is a subgraph of G (up to renaming with 1 being mapped to v in G), then $P \in \text{supp}(\mathcal{D}_\ell)$.

We summarize this as

$$P \in \text{supp}(\mathcal{D}_\ell) \iff \texttt{Graph}(P) \text{ is a subgraph of } G \text{ (up to renaming with 1 mapped to } v). \tag{3.1}$$

Therefore, one would hope to enumerate over all possible experiments P and use the information of whether P is in $\text{supp}(\mathcal{D}_\ell)$ to reconstruct $B(v, r)$. Let us formalize this.

We call an experiment P *economical* if for any two integers a, b in the path, the segment $a \to b$ appears at most once in P. All paths studied in the warm-up case are economical.

One can now study the following algorithm `NaiveReconstruct`. It enumerates over all valid experiments by the increasing order of their lengths, in order to find the longest experiment $P^* \in \text{supp}(\mathcal{D}_\ell)$ such that

both P^* is economical and $\texttt{Graph}(P^*)$ is of radius r from vertex 1.

Owing to (3.1), this P^* satisfies that $\texttt{Graph}(P^*)$ is isomorphic to $B(v, r)$ and the isomorphism maps vertex 1 to vertex v. Since any economical experiment P of length $2(m + 1)$ has at least $m + 1$ edges in its supporting graph, $\texttt{Graph}(P)$ cannot be a subgraph of $B(v, r)$ and thus $P \notin \text{supp}(\mathcal{D}_\ell)$. This implies that `NaiveReconstruct` only needs oracle access to $\text{supp}(D_\ell)$ for $\ell \le 2(m + 1)$ in order to determine that P^* is the longest such experiment.

3.2. A constructive proof of Theorem 1

Although being sufficient for reconstructing $B(v, r)$ given oracle access to $\text{supp}(\mathcal{D}_\ell)$, `NaiveReconstruct` is still unsatisfactory because (1) the enumeration procedure is too slow and (2) the algorithm is not generalizable to the improvement

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035

case studied in Section 1.4. We thus propose a more constructive algorithm `Reconstruct` that only makes $O(n^2)$ membership queries to $\mathrm{supp}(\mathcal{D}_\ell)$.

At a high level, `Reconstruct` builds $B(v, r)$ by learning $B(v, 1), \dots, B(v, r)$ layer by layer, and for each layer, by learning the vertices one by one. At any time of the algorithm, we maintain an economical experiment P whose supporting graph $\mathtt{Graph}(P)$ is a subgraph of $B(v, r)$. We incrementally "add" new vertices or edges to $\mathtt{Graph}(P)$, verify if the new graph is still a subgraph of $B(v, r)$ using (3.1), and if so, we update the current experiment P and continue. The details are as follows.

We describe `Reconstruct` in Fig. 4 and show its correctness by an induction on r. Suppose that we have reconstructed $B(v, r_0 - 1)$ for some value $r_0 - 1 \geq 0$, and we now want to reconstruct $B(v, r_0)$ using $\mathcal{D}_\ell$ where $\ell = 2(m + 1)$.

Let n_0 be the number of vertices in $B(v, r_0 - 1)$, and P_0 an arbitrary experiment such that $G_0 \stackrel{\text{def}}{=} \mathtt{Graph}(P_0)$ is a reconstruction of $B(v, r_0 - 1)$.[5] We also denote by $u_1, \dots, u_k \in [n_0]$ the vertices in G_0 that have distance precisely $r_0 - 1$ from vertex 1. We iterate over all $i = 1, 2, \dots, k$, and for each i we first let $P' = P_{i-1}$ and repeatedly do the following (see Line 7 through 8 in Fig. 4).

Whenever $\mathtt{Replace}(P',\ u_i,\ u_i \to \star \to u_i)$ exists in $\mathrm{supp}(\mathcal{D}_\ell)$, where $\star$ is the smallest integer not appearing in P', we know that there is at least one more vertex neighboring to u_i that is not explored so far, and we add it to P' by updating $P' \leftarrow \mathtt{Replace}(P',\ u_i,\ u_i \to \star \to u_i)$. Equivalently, this update on P' can be understood as we are introducing a new vertex x along with a new edge (x, u_i) to $\mathtt{Graph}(P')$.

As soon as a new vertex $\star$ is added to P', we add the edges connecting $\star$ to other vertices in $\mathtt{Graph}(P')$ as follows. In principle, $\star$ may be connected to any vertex in $u' \in \{u_{i+1}, \dots, u_k\} \cup \{n_0 + 1, \dots, \star - 1\}$, and we check them one by one. For each such a candidate neighbor u', we check if $\mathtt{Replace}(P',\ \star,\ \star \to u' \to \star)$ exists in $\mathrm{supp}(\mathcal{D}_\ell)$, and if so, we update $P' \leftarrow \mathtt{Replace}(P',\ \star,\ \star \to u' \to \star)$ and continue to the next u'. Equivalently, this update can be understood as we are adding an extra edge between x and u' into $\mathtt{Graph}(P')$.

Let P_i be the final experiment P' after exploring all the vertices neighboring to u_i, and $G_i = \mathtt{Graph}(P_i)$. We have, according to (3.1), that G_i is a subgraph of G. In fact, the last such subgraph G_k reconstructs $B(v, r_0)$:

Claim 3.1. *G_k is isomorphic to $B(v, r_0)$ and the isomorphism maps vertex 1 to vertex v.*

Proof. First of all, G_k must be a subgraph of $B(v, r_0)$ because $P_k \in \mathrm{supp}(\mathcal{D}_\ell)$ and, by construction, all vertices of G_k are within distance r_0 from vertex 1. Therefore, we only need to verify if there is any vertex or edge in $B(v, r_0)$ missing from G_k.

Let σ be an arbitrary *embedding* of G_k into $B(v, r_0)$, i.e., a mapping from the vertex set of G_k to that of $B(v, r_0)$, preserving edges, and mapping vertex 1 to vertex v.

For the missing vertex case, we prove by way of contradiction and suppose there is a vertex w in $B(v, r_0) \setminus B(v, r_0 - 1)$ missing from G_k under this embedding σ. Because w is at distance r_0 from v, it must be connected to some vertex at distance $r_0 - 1$ from v. Let this vertex be $\sigma(u_i)$ for some $i \in [k]$. (There must exist such a u_i because G_0 reconstructs $B(v, r_0 - 1)$ from the inductive step.)

Next, since w is missing from G_k, vertex u_i must have fewer neighbors in G_k than vertex $\sigma(u_i)$ does in $B(v, r_0)$. At the time we finish constructing P_i (so the while loop in Line 9 from Fig. 4 terminates), $G_i = \mathtt{Graph}(P_i)$ can be embedded into G under the same σ. Letting $\widehat{P} = \mathtt{Replace}(P_i,\ u_i,\ u_i \to x \to u_i)$, the same embedding σ, while appended with $\sigma(x) \mapsto w$, should provide a valid embedding of $\mathtt{Graph}(\widehat{P})$ into G, and according to (3.1) this implies $\widehat{P} \in \mathrm{supp}(\mathcal{D}_\ell)$. This contradicts the termination condition of the while loop in Line 9 that says $\widehat{P} \notin \mathrm{supp}(\mathcal{D}_\ell)$. Therefore there is no missing vertex.

One can perform a similar argument for the missing edge case. □

In sum, we have shown that $B(v, r_0)$ can be constructed by the algorithm above, and by induction, `Reconstruct` outputs a reconstruction of $B(v, r)$. Notice that the experiment P, at the end of the algorithm, has a total length of $2m$ because each edge in $B(v, r)$ is traversed precisely once in each direction. Therefore the longest experiment `Reconstruct` has ever queried is of length $2(m + 1)$, and choosing $\ell = 2(m + 1)$ is sufficient for our purpose. In addition, `Reconstruct` makes no more than $O(n^2)$ membership queries to $\mathrm{supp}(\mathcal{D}_\ell)$. ■

4. Theorem 2: a lower bound on experiment length

In this section, for any integer $h \geq 1$, we construct two (infinite) binary trees $T_1 = T_1^{(h)}$ and $T_2 = T_2^{(h)}$ with the starting vertex being the root for both cases. We show, quite surprisingly, although T_1 and T_2 are different at depth $r = 2h + 3$, any anonymous experiment of length no longer than $\ell = O(2^h)$ has the same probability to be generated from T_1 and T_2. Formally,

Lemma 4.1. *There exists a constant c such that, given two binary trees $T_1 = T_1^{(h)}$ and $T_2 = T_2^{(h)}$ (as constructed in Fig. 5), and letting the starting vertex v_1 and v_2 be their roots, we have:*

- *$B^{T_1}(v_1, 2h + 3)$ and $B^{T_2}(v_2, 2h + 3)$ are different (i.e., non-isomorphic), but*
- *the distributions over random experiments of length $\ell \leq c \cdot 2^h$ in T_1 and T_2 are the same.*

[5] I.e., P_0 satisfies that $\mathtt{Graph}(P_0)$ is isomorphic to $B(v, r_0 - 1)$ and the isomorphism maps vertex 1 to v. In fact, P_0 is inherited from the inductive step of the algorithm, and corresponds to an arbitrary walk that starts from v and traverses each edge in $B(v, r_0 - 1)$ exactly once in each direction.

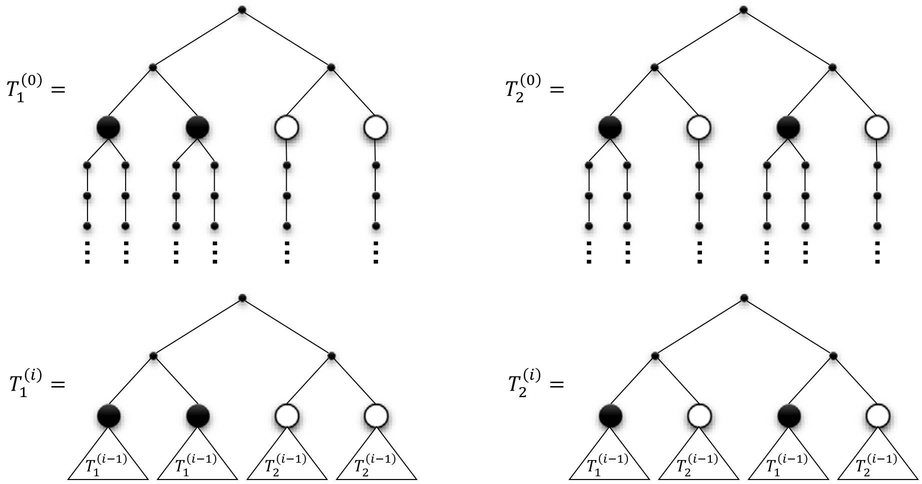


Fig. 5. The recursive definition of the hard instance for Theorem 2.

Theorem 2 is immediately implied by the above lemma, because it rules out the possibility of reconstructing $B(v, r)$, even for binary trees, with oracle access to $(\mathcal{D}_1, \dots, \mathcal{D}_\ell)$ for any $\ell = 2^{o(r)}$.

4.1. Our hard instance

We define $T_1 = T_1^{(h)}$ and $T_2 = T_2^{(h)}$ recursively.

Let $T_1^{(0)}$ and $T_2^{(0)}$ be defined as follows (see Fig. 5): the roots of both trees have two children and each child in turns has two children; among the four grandchildren of the root, two of them are "black", having two infinite chains of descendants, and two of them are "white", having one infinite chain of descendants.

$T_1^{(i)}$ and $T_2^{(i)}$ are defined similarly (see Fig. 5): the roots of both trees have two children and each child in turns has two children; among the four grandchildren of the root, two of them are "black", having $T_1^{(i-1)}$ as subtrees, and two of them are "white", having $T_2^{(i-1)}$ as subtrees.

4.2. A warm-up property

For $j \in \{1, 2\}$, let $\mathcal{D}_{j,\ell}$ be the distribution over random experiments of length ℓ generated from the Markov process starting from the root of T_j. Given an experiment P of length ℓ, we denote by $\Pr[P \mid T_j]$ the probability that P is generated from $\mathcal{D}_{j,\ell}$.

Recall that one can associate P with its supporting graph $G_P = \mathtt{Graph}(P)$. Since T_1 and T_2 are binary trees, if the supporting graph G_P has cycles or is non-binary, P cannot exist in $\mathcal{D}_{j,\ell}$. We thus focus only on the experiments P for which G_P is a binary tree. We make the following claim:

Claim 4.2. *If the root (i.e., vertex* 1*) of G_P has at most one grandchild, then* $\Pr[P \mid T_1] = \Pr[P \mid T_2]$.

Before proving Claim 4.2, we summarize the high level intuition as follows.

Any experiment P is consistent with a set of walks $\mathcal{Q}_1$ on T_1, and a set of walks $\mathcal{Q}_2$ on T_2. The probability $\Pr[P \mid T_j]$ is equal to $\sum_{Q \in \mathcal{Q}_j} \Pr[Q \mid T_j]$, the sum of probabilities over the walks in $\mathcal{Q}_j$, i.e., those walks consistent with P. We show that, under the condition P visits only one grandchild of the root, there is a one-to-one mapping τ between $\mathcal{Q}_1$ and $\mathcal{Q}_2$ that preserves probabilities. This immediately implies that $\Pr[P \mid T_1] = \Pr[P \mid T_2]$. The one-to-one mapping τ is illustrated in Fig. 6, and note that if P visits two grandchildren such a mapping may not exist.

Proof of Claim 4.2. We prove the claim when the root has only one grandchild in G_P. The other case – when the root has no grandchild – is only simpler. We denote by $u \in \mathbb{Z}_+$ this unique grandchild, and focus on the case of $h = 0$; the case of $h > 0$ is similar.

Let the four grandchildren of the root in $T_1^{(0)}$ be denoted by a_1, a_2, a_3, a_4 respectively, and the four grandchildren of the root in $T_2^{(0)}$ be denoted by b_1, b_2, b_3, b_4. We order them according to Fig. 6 so a_1, a_2, b_1, b_3 are black, and a_3, a_4, b_2, b_4 are white.

We now construct a one-to-one mapping τ from the walks on $T_1^{(0)}$ that are consistent with P to the walks on $T_2^{(0)}$ that are consistent with P. Our τ is defined "by picture", with four representative examples given in Fig. 6.

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035

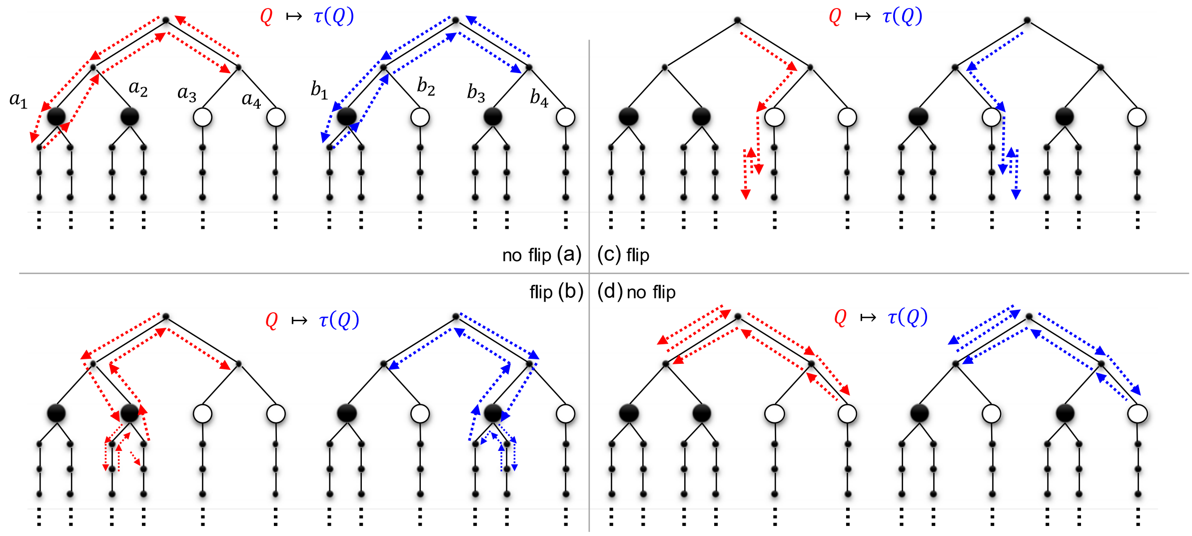


Fig. 6. The illustration of our mapping τ used in Claim 4.2.

More precisely, to define τ, we first draw $T_1^{(0)}$ and $T_2^{(0)}$ on the plane with four grandchildren of the root sorted as a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4 from left to right. Then, given a walk Q on $T_1^{(0)}$ (starting from the root) that is consistent with P, denoted as $Q \lhd (T_1^{(0)}, P)$, vertex u in P must be mapped to one of $\{a_1, a_2, a_3, a_4\}$.

- If u is mapped to a_2 or a_3 in Q (see Fig. 6(b) and (c)), we let $Q' = \tau(Q)$ be the walk in $T_2^{(0)}$ that is *flipped* left and right (with respect to the plane), and thus u is mapped to b_3 or b_2 respectively in Q'.
- If u is mapped to a_1 or a_4 in Q (see Fig. 6(a) and (d)), we let $Q' = \tau(Q)$ be the "same" walk Q under translation on the plane, and thus u is mapped to b_1 or b_4 respectively in Q'.

It is not hard to verify that τ is a one-to-one mapping. In addition, the ith vertex in Q has the same degree as the ith vertex in $Q' = \tau(Q)$ for any i and any Q satisfying $Q \lhd (T_1^{(0)}, P)$. Therefore we have $\Pr\left[Q \mid T_1^{(0)}\right] = \Pr\left[Q' \mid T_2^{(0)}\right]$, i.e., Q and Q' have the same probability to be generated in the random walk from $T_1^{(0)}$ and $T_2^{(0)}$ respectively. This implies

$$\Pr\left[P \mid T_1^{(0)}\right] = \sum_{Q \lhd (T_1^{(0)}, P)} \Pr\left[Q \mid T_1^{(0)}\right] = \sum_{Q \lhd (T_1^{(0)}, P)} \Pr\left[\tau(Q) \mid T_2^{(0)}\right]$$
$$= \sum_{Q' \lhd (T_2^{(0)}, P)} \Pr\left[Q' \mid T_2^{(0)}\right] = \Pr\left[P \mid T_2^{(0)}\right],$$

that is, P has the same chance to be generated as an experiment in $T_1^{(0)}$ and $T_2^{(0)}$. □

4.3. A general property

For any $i \in \{0, 1, \ldots, 2h\}$, we denote by L_i the set of vertices (in the form of integer numbers) in $G_P = \mathtt{Graph}(P)$ at depth i from the root (where the root itself is in L_0). We prove the following property about a shortest experiment in which $\Pr[P \mid T_1] \neq \Pr[P \mid T_2]$.

Lemma 4.3. *Given a shortest experiment P in which $\Pr[P \mid T_1] \neq \Pr[P \mid T_2]$, any $i \in \{0, 1, \ldots, h\}$, and any $u \in L_{2i}$, vertex u has at least two grandchildren in G_P.*

Notice that the case of $i = 0$ is a direct consequence of Claim 4.2, but the proof for the $i \geq 1$ case is more involved. Before proving it formally, we summarize the basic idea as follows.

If P is a shortest such experiment, and if there exists some u in P with only one grandchild, we shorten P to a new experiment P' by essentially removing all occurrences of u and the descendants of u. In a rough sense, $\Pr[P \mid T_j]$ equals to $\Pr[P' \mid T_j] \times \Pr[P \setminus P' \mid T_j]$ where $P \setminus P'$ is an experiment corresponding to the removed segment of vertices. Because u has only one grandchild in G_P, this removed subsegment $P \setminus P'$ has the same probability to be generated in T_1 and T_2 (owing to Claim 4.2). We therefore conclude that $\Pr[P' \mid T_1] \neq \Pr[P' \mid T_2]$, contradicting to the fact that P is the shortest such experiment.

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035

Proof of Lemma 4.3. The case of $i = 0$ is inherited from Claim 4.2, so the rest of this section is devoted to proving Lemma 4.3 for $i \geq 1$.

For $j \in \{1, 2\}$, let $\mathcal{D}_{j,\ell}$ be the distribution over random *experiments* of length ℓ in tree T_j, and $\mathcal{D}^{\text{walk}}_{j,\ell}$ the distribution over random *walks* in tree T_j. We make a quick observation first.

Given an experiment P, the probability $\Pr[P \mid T_j]$ is the sum of the probabilities $\Pr[\sigma(P) \mid T_j]$ over all choices of *embeddings* $\sigma : G_P \to T_j$:

$$\Pr[P \mid T_j] = \sum_{\text{embedding } \sigma : G_P \to T_j} \Pr[\sigma(P) \mid T_j]. \tag{4.1}$$

Here an embedding σ is a mapping from the vertices in G_P to the vertices in T_j, while preserving edges and mapping vertex 1 to vertex v. Accordingly, σ maps an experiment P to an actual walk $\sigma(P)$ on T_j, and $\Pr[\sigma(P) \mid T_j]$ is the probability for $\sigma(P)$ to be generated from $\mathcal{D}^{\text{walk}}_{j,\ell}$. We also recall a useful fact by the definition of random walk:

$$\Pr[\sigma(P) \mid T_j] = \prod_{i=1}^{\ell} \frac{1}{\deg(\sigma(P^{(i)}))}, \tag{4.2}$$

where $P^{(i)}$ is the ith integer in the experiment P, and thus $\deg(\sigma(P^{(i)}))$ is the degree of the ith vertex in the length-ℓ walk $\sigma(P)$.

We are now ready to prove Lemma 4.3. Suppose that Lemma 4.3 does not hold for some $i \in \{1, \dots, h\}$, and vertex $u \in L_{2i}$ has only one grandchild in G_P, we will show that one can shorten P to construct a new experiment P' where it also satisfies $\Pr[P' \mid T_1] \neq \Pr[P' \mid T_2]$, contradicting the fact that P is the shortest such experiment. In order to shorten P, we first discover that P must be of some special structure, described as follows.

We note that P can be viewed as a "walk" on its supporting graph $G_P = \texttt{Graph}(P)$, and let the w be parent of u in G_P. Clearly, P must visit w before it visits u in this walk, but we claim that P can only be one of the two forms:

- either it enters the subtree rooted at u, then comes back to w and *never* visits u again;
- or it enters the subtree rooted at u and *never* comes back to w.

Formally,

Claim 4.4. *P must be of the form:*

$$P = P_1 \to w \to u \to P_2 \to u \to w \to P_3 \quad \textit{or} \quad P = P_1 \to w \to u \to P_2$$

where P_2 consists of only vertices that are u or descendants of u (in G_P), while P_1 and P_3 consist of only vertices that are neither u nor descendants of u (in G_P).

Proof. Suppose that P is of neither of the two forms above, then P must visit some descendants of u first, then non-descendants, and then descendants again. For instance, such a walk could be

$$P = P_1 \to w \to u \to P_2 \to u \to w \to P_3 \to w \to u \to P_2'$$

where P_2 and P_2' consist of only u or descendants or u, while P_1 and P_3 consist of only vertices that are neither u nor descendants of u. We only prove the claim for this case above, and other cases are similar.

We first swap the order of the vertices in P and construct the following experiment P':

$$P' = P_1 \to w \to P_3 \to w \to u \to w \to u \to P_2 \to u \to P_2'.$$

Since for any two integers a and b the directed edge $a \to b$ appears exactly the same number of times in P and P', we have that $\Pr[P \mid T_j] = \Pr[P' \mid T_j]$ according to (4.1) and (4.2), for both $j = 1$ and 2.

We next observe that the subsequence $w \to u \to w \to u$ is redundant: since u and w are of depth $2i$ and $2i-1$ respectively, they will always be mapped to vertices with degree 3 in T_1 or T_2. As a result, if we define

$$P'' = P_1 \to w \to P_3 \to w \to u \to P_2 \to u \to P_2'$$

we must have $\Pr[P' \mid T_j] = (\frac{1}{3})^2 \Pr[P'' \mid T_j]$ for both $j = 1$ and $j = 2$, according to (4.1) and (4.2) again. This indicates $\Pr[P' \mid T_1] \neq \Pr[P' \mid T_2]$, contradicting the choice of P which is the shortest experiment that makes $\Pr[P \mid T_1] \neq \Pr[P \mid T_2]$. □

Now we focus only on the case of $P = P_1 \to w \to u \to P_2 \to u \to w \to P_3$ because the other one is only simpler. We want to shorten it to $P_1 \to w \to P_3$.

Claim 4.5. *If an experiment $P = P_1 \to w \to u \to P_2 \to u \to w \to P_3$ satisfies $\Pr[P \mid T_1] \neq \Pr[P \mid T_2]$ (where the definitions of P_1, P_2 and P_3 are the same as Claim 4.4), and u has at most one grandchild in $\texttt{Graph}(P)$, then we have*

$$\Pr[P_1 \to w \to P_3 \mid T_1] \neq \Pr[P_1 \to w \to P_3 \mid T_2].$$

Proof. To see this, we consult (4.1) again. For each $j \in \{1, 2\}$, and given an embedding $\sigma : G_P \to T_j$, we write it as a pair $\sigma = (\sigma_1, \sigma_2)$ where

- σ_1 maps all the vertices excluding the descendants of u (so including u) to T_j,
- σ_2 maps all the descendants of u (including u) to T_j, and
- σ_1 and σ_2 map u to the same vertex in T_j. (When this is the case, we denote by $\sigma_1 \sim \sigma_2$.)

We therefore rewrite:

$$\Pr[P \mid T_j] = \sum_{\sigma} \Pr[\sigma(P) \mid T_j] = \sum_{\sigma_1} \sum_{\sigma_2 : \sigma_1 \sim \sigma_2} \Pr\big[(\sigma_1, \sigma_2)(P) \,\big|\, T_j\big]. \tag{4.3}$$

Next, recall that $Q = \sigma(P) = (\sigma_1, \sigma_2)(P)$ is a walk on the tree T_j, and $\Pr[Q \mid T_j]$ can be written as a product of the reciprocal of degrees, i.e., $\Pr[Q \mid T_j] = \prod_{i=1}^{\ell} \frac{1}{\deg(Q^{(i)})}$ in which $\deg(Q^{(i)})$ is the degree of the ith vertex in the walk Q. This allows us to break Q into five segments: $\sigma_1(P_1 \to w)$, $\sigma_1(w \to u)$, $\sigma_2(u \to P_2 \to u)$, $\sigma_1(u \to w)$, and $\sigma_1(w \to P_3)$, and compute

$$\begin{aligned}\Pr[(\sigma_1, \sigma_2)(P) \mid T_j] = {} & \Pr\big[\sigma_1(P_1 \to w) \,\big|\, T_j\big] \cdot \Pr\big[\sigma_1(w \to u) \,\big|\, T_j\big] \\ & \cdot \Pr\big[\sigma_2(u \to P_2 \to u) \,\big|\, T_j\big] \cdot \Pr\big[\sigma_1(u \to w) \,\big|\, T_j\big] \cdot \Pr\big[\sigma_1(w \to P_3) \,\big|\, T_j\big].\end{aligned}$$

We reorder them into four segments $\sigma_1(P_1 \to w \to P_3)$, $\sigma_1(w \to u)$, $\sigma_1(u \to w)$, $\sigma_2(u \to P_2 \to u)$, and conclude that

$$\Pr[(\sigma_1, \sigma_2)(P) \mid T_j] = \Pr\big[\sigma_1(P_1 \to w \to P_3) \,\big|\, T_j\big] \cdot \Pr\big[\sigma_1(w \to u) \,\big|\, T_j\big] \cdot \Pr\big[\sigma_1(u \to w) \,\big|\, T_j\big] \cdot \Pr\big[\sigma_2(u \to P_2 \to u) \,\big|\, T_j\big].$$

However, we must have $\Pr[\sigma_1(w \to u) \mid T_j] = \Pr[\sigma_1(u \to w) \mid T_j] = 1/3$ because any embedding $\sigma = (\sigma_1, \sigma_2)$ maps u and w to vertices with degree 3. This, combined with (4.3) gives us

$$\begin{aligned}\Pr[P \mid T_j] &= \sum_{\sigma_1} \sum_{\sigma_2 : \sigma_1 \sim \sigma_2} \Pr\big[\sigma_1(P_1 \to w \to P_3) \,\big|\, T_j\big] \cdot \frac{1}{3} \cdot \frac{1}{3} \cdot \Pr\big[\sigma_2(u \to P_2 \to u) \,\big|\, T_j\big] \\ &= \frac{1}{9} \sum_{\sigma_1} \Pr\big[\sigma_1(P_1 \to w \to P_3) \,\big|\, T_j\big] \cdot \sum_{\sigma_2 : \sigma_1 \sim \sigma_2} \Pr\big[\sigma_2(u \to P_2 \to u) \,\big|\, T_j\big].\end{aligned}$$

Now, fixing any σ_1, we know that u is mapped to vertex $\sigma_1(u)$ in T_j, and $\sigma_1(u)$ must be the root of some $T_k^{(h-i)}$ tree for $k \in \{1, 2\}$. Here the value of k depends on the choice of σ_1. We observe that the summation

$$\sum_{\sigma_2 : \sigma_1 \sim \sigma_2} \Pr\big[\sigma_2(u \to P_2 \to u) \,\big|\, T_j\big]$$

is precisely the probability for the experiment $u \to P_2 \to u$ (after renaming so that the integers are 1-based) to be generated in $T_k^{(h-i)}$, and this value does not depend on the choice of k owing to Claim 4.2 and the fact that u has at most one grandchild in P_2. Let this value be $p \in [0, 1]$, and we conclude that

$$\Pr[P \mid T_j] = \frac{1}{9} \sum_{\sigma_1} \Pr\big[\sigma_1(P_1 \to w \to P_3) \,\big|\, T_j\big] \cdot p = \frac{p}{9} \cdot \Pr[P_1 \to w \to P_3 \mid T_j],$$

that is, the value of $\Pr[P \mid T_j]$ is a fixed constant $\frac{p}{9}$ multiplied by that of a shorter experiment $P_1 \to w \to P_3$ on the same tree T_j. Since this is true for both $j \in \{1, 2\}$, we conclude that $\Pr[P_1 \to w \to P_3 \mid T_1] \neq \Pr[P_1 \to w \to P_3 \mid T_2]$. □

Since the above claim contradicts the choice of P which is the shortest such sequence that makes $\Pr[P \mid T_1] \neq \Pr[P \mid T_2]$, we finish the proof of Lemma 4.3. ■

4.4. *Proof of Lemma 4.1*

Proof. It is immediate that Lemma 4.3 implies Lemma 4.1: the shortest experiment P that distinguishes $\Pr[P \mid T_1]$ and $\Pr[P \mid T_2]$ must branch out *at least once for every two levels*, and therefore $|L_{2i}| \geq 2^i$ and in particular $L_{2(h+1)} \geq 2^{h+1}$. This shows that the length of P must be at least $\Omega(2^h)$ (in order to visit 2^{h+1} distinct vertices at depth $2(h+1)$). In other words, there exists some constant c where $\Pr[P \mid T_1] = \Pr[P \mid T_2]$ for any experiment of length $\ell \leq c \cdot 2^h$. ■

5. Theorem 3: a lower bound on the number of experiments

5.1. *Our new hard instance*

We slightly modify our hard instance in Fig. 5, by replacing the definitions of $T_1^{(0)}$ and $T_2^{(0)}$ with Fig. 7: instead of having a black vertex to be the root of two infinite chains and the white vertex to be the root of one (recall Fig. 5), we let a black vertex be the parent of three infinite complete binary trees, and the white one be the parent of two. The new trees $T_1 = T_1^{(h)}$

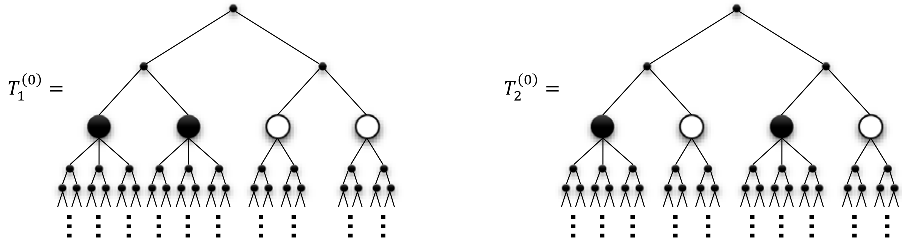


Fig. 7. The new choices of $T_1^{(0)}$ and $T_2^{(0)}$ for Theorem 3.

and $T_2 = T_2^{(h)}$ so constructed are ternary. (In fact, one can verify that the binary variants of T_1 and T_2 also suffice for the purpose of Theorem 3, but they will make the analysis more involved.)

It is a simple exercise to verify that all the proofs in Section 4 remain true for this new hard instance pair (T_1, T_2), and therefore Lemma 4.1 still applies: that is, there exists a constant c such that, letting the starting vertex v_1 and v_2 be the corresponding roots, we have:

- $B^{T_1}(v_1, 2h+3)$ and $B^{T_2}(v_2, 2h+3)$ are different (i.e., non-isomorphic), but
- the distributions over random experiments of length $\ell \le c \cdot 2^h$ in T_1 and T_2 are the same.

This new construction satisfies an additional property:

- in a random walk on either T_1 or T_2, if the current vertex is at depth d for some $d \in \mathbb{Z}_{\ge 0}$, then with probability *at least* $2/3$, the next vertex is going to be at depth $d+1$.

5.2. A structural lemma

We say that an experiment P is *bad*, if it has less than 2^{h+1} vertices at depth $2(h+1)$ in its supporting graph $G_P = \mathtt{Graph}(P)$. We denote by $\mathcal{BAD}$ the set of bad experiments. According to the proof of Lemma 4.1, any bad experiment $P \in \mathcal{BAD}$ has the same chance to be seen in T_1 and T_2, that is, $\Pr[P \mid T_1] = \Pr[P \mid T_2]$.

We now compute a lower bound on the chance of a random experiment to be bad.

Lemma 5.1. *For any $j \in \{1, 2\}$, and any value of ℓ, with probability at least $1 - e^{-\Omega(2^h)}$, the random experiment of length ℓ generated from T_j is bad.*

The proof of Lemma 5.1 mostly consists of careful applications of Chernoff and union bounds, and we summarize its intuition as follows.

By our construction of the trees, any random walk (on either T_1 or T_2) of length t is likely to arrive at a vertex at depth $\Omega(t)$. This is because, in each step of the random walk, the depth increases by 1 with probability at least $2/3$, and decreases by 1 with probability at most $1/3$. More precisely, by Chernoff bound, the random walk will land at a vertex of depth $\Omega(2^h)$ after $t = 2^h$ steps, with probability at least $1 - e^{-\Omega(2^h)}$. Since $2^h < 2^{h+1}$, in order for this random walk to correspond to a good experiment, it has to come back from depth $\Omega(2^h)$ to depth $2(h+1)$ in order to visit 2^{h+1} vertices at that depth. This, again using Chernoff bound, is a very unlikely event, because going back from depth $\Omega(2^h)$ to depth $2(h+1)$ has a probability at most $e^{-\Omega(2^h)}$, no matter how long the random walk is. It is crucial here that the probability does not depend on ℓ.

Proof of Lemma 5.1. If suffices to prove the lemma for $\ell \ge 2^{h+1}$, because otherwise any experiment P of length ℓ cannot visit 2^{h+1} vertices at depth $2(h+1)$ and is by definition bad.

Let $\mathsf{dep}_i \in \mathbb{Z}_{\ge 0}$ be the random variable indicating the depth of the ith vertex in the random walk on T_j, where $i \in \{0, 1, \dots, \ell\}$. We have $\mathsf{dep}_0 = 0$. Let the random variable x_i be defined as $\frac{\mathsf{dep}_i - \mathsf{dep}_{i-1} + 1}{2} \in \{0, 1\}$, so that $\mathsf{dep}_i = \mathsf{dep}_{i-1} + (-1 + 2x_i) = \mathsf{dep}_{i-1} \pm 1$. By the construction of our graph (either T_1 or T_2), we always have $\mathbb{E}[x_i] \ge \frac{2}{3}$, that is, with probability at least $\frac{2}{3}$ the depth increases by 1 in a step.

Let us consider a special timestamp in the random walk: time $t = 2^h$. Using Chernoff bound, we deduce below that with very high probability (i.e., $1 - e^{-\Omega(t)}$), we have that $\mathsf{dep}_t \ge \frac{1}{6}t = \frac{2^h}{6}$:

$$\Pr\left[\mathsf{dep}_t < \frac{1}{6}t\right] = \Pr\left[-t + 2(x_1 + \cdots + x_t) < \frac{1}{6}t\right] = \Pr\left[x_1 + \cdots + x_t < \frac{7}{12}t\right]$$
$$\le \Pr\left[x_1 + \cdots + x_t < \frac{7}{8}\mathbb{E}[x_1 + \cdots + x_t]\right] \le e^{-\Omega(t)}.$$

Recall that within $t = 2^h$ steps the random walk cannot visit enough (i.e., at least 2^{h+1}) vertices at depth $2(h+1)$, so for a random walk of length ℓ to correspond to a good experiment, it must come back from depth dep_t to depth $2(h+1)$ in the remaining $\ell - t$ steps.

Conditioning on that $\mathsf{dep}_t \ge \frac{1}{6}t = \frac{2^h}{6}$, we compute the chance of the random experiment to reach back to a vertex at depth $\le 2(h+1)$ at time $t+t'$ where $t' \in \{1, \dots, \ell - t\}$.

$$\begin{aligned}\Pr\left[\mathsf{dep}_{t+t'} \le 2(h+1)\right] &\le \Pr\left[\mathsf{dep}_{t+t'} - \mathsf{dep}_t \le 2(h+1) - \frac{2^h}{6}\right] \\ &= \Pr\left[-t' + 2(x_{t+1} + \dots + x_{t+t'}) \le 2(h+1) - \frac{2^h}{6}\right] \\ &= \Pr\left[x_{t+1} + \dots + x_{t+t'} \le \frac{t'}{2} + (h+1) - \frac{2^h}{12}\right]. \end{aligned} \tag{5.1}$$

We assume that h is sufficiently large (e.g. $h \ge 8$) so that $\frac{t'}{2} + (h+1) - \frac{2^h}{12} \le \frac{t'}{2} - \frac{2^h}{24}$. Then obviously t' has to be at least $\frac{2^h}{12}$ before this probability in (5.1) becomes non-zero. Therefore, we only focus on the choices of $t' \ge \frac{2^h}{12}$ and continue our calculation using Chernoff bound:

$$\begin{aligned}\Pr\left[\mathsf{dep}_{t+t'} \le 2(h+1)\right] &\le \Pr\left[x_{t+1} + \dots + x_{t+t'} \le \frac{t'}{2} - \frac{2^h}{24}\right] \\ &\le \Pr\left[x_{t+1} + \dots + x_{t+t'} \le \frac{t'}{2}\right] \\ &\le \Pr\left[x_{t+1} + \dots + x_{t+t'} \le \frac{3}{4}\mathbb{E}[x_{t+1} + \dots + x_{t+t'}]\right] \le e^{-\Omega(t')}. \end{aligned}$$

Finally, since we only need to focus on $t' \ge \frac{2^h}{12}$ due to the discussed reason, by a union bound over all integers $t' \in \left[\frac{2^h}{12}, \ell - t\right]$, we have that the chance for a random experiment to visit back to depth $2(h+1)$ is at most $\sum_{t'=2^h/12}^{\ell-t} e^{-\Omega(t')} = e^{-\Omega(t)}$.

In sum, we know that with probability at least $1 - e^{-\Omega(t)} = 1 - e^{-\Omega(2^h)}$, the random walk generated (from either T_1 or T_2) will: (1) have $\mathsf{dep}_t \ge \frac{2^h}{12}$ and (2) never visit back to depth $2(h+1)$. The experiment corresponding to this walk has to be bad, and therefore we finish the proof. □

5.3. Proof of Theorem 3

We argue that in order to distinguish $T_1 = T_1^{(h)}$ from $T_2 = T_2^{(h)}$ with probability at least $\frac{1}{2}$, one needs at least $e^{\Omega(2^h)}$ samples of random experiments of arbitrary lengths.

Indeed, let $\mathcal{D}_{1,\ell}$ be the distribution over random experiments of length ℓ for tree T_1, and $\mathcal{D}_{2,\ell}$ that for T_2. By definition, $\mathcal{D}_{1,\ell}$ and $\mathcal{D}_{2,\ell}$ are identical on the support of $\mathcal{BAD}$, the set of bad experiments. Therefore, owing to Lemma 5.1, the total variation distance (i.e., half of the 1-norm distance) between them $\|\mathcal{D}_{1,\ell} - \mathcal{D}_{2,\ell}\|_{TV}$ is at most $e^{-\Omega(2^h)}$ for any ℓ; that is, any algorithm that samples an experiments from $\mathcal{D}_{1,\ell}$ or $\mathcal{D}_{2,\ell}$, can only tell the difference with probability at most $e^{-\Omega(2^h)}$.

Using union bound, given any algorithm that takes samples from distributions $(\mathcal{D}_{j,1}, \mathcal{D}_{j,2}, \dots,)$, unless it takes $e^{\Omega(2^h)}$ samples, it cannot distinguish T_1 from T_2 with any constant probability.

Let A be an algorithm that reconstructs $B(v, r)$ – even only for the case when the underlying graph is a ternary tree – with probability 1/2 using N random experiments. If $N = 2^{2^{o(r)}}$, then using A one can reconstruct $B(v, 2h+3)$ for T_1 and T_2 respectively, and thus distinguish T_1 from T_2. This leads to a contradiction because no algorithm can distinguish T_1 from T_2 using only $e^{o(2^h)}$ samples; therefore we must have $N = 2^{2^{\Omega(r)}}$. ■

References

[1] Noga Alon, Vera Asodi, Learning a hidden subgraph, SIAM J. Discret. Math. 18 (4) (2005) 697–712.
[2] Noga Alon, Richard Beigel, Simon Kasif, Steven Rudich, Benny Sudakov, Learning a hidden matching, in: Proceedings of the 43rd Symposium on Foundations of Computer Science, FOCS'02, Washington, DC, USA, 2002, pp. 197–206.
[3] Reid Andersen, Fan Chung, Kevin Lang, Using pagerank to locally partition a graph, Internet Math. 4 (1) (2007) 35–64. An extended abstract appeared in FOCS'2006.
[4] Reid Andersen, Yuval Peres, Finding sparse cuts locally using evolving sets, in: Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing, STOC'09, 2009, pp. 235–244.
[5] Dana Angluin, Jiang Chen, Learning a hidden hypergraph, J. Mach. Learn. Res. 7 (2006) 2215–2236.
[6] Dana Angluin, Jiang Chen, Learning a hidden graph using $O(\log n)$ queries per edge, J. Comput. System Sci. 74 (4) (2008) 546–556. Appeared in COLT 2004.
[7] Mathilde Bouvel, Vladimir Grebinski, Gregory Kucherov, Combinatorial search on graphs motivated by bioinformatics applications: a brief survey, in: Proceedings of the 31st International Conference on Graph-Theoretic Concepts in Computer Science, WG'05, 2005, pp. 16–27.
[8] Wray Buntine, Theory refinement on bayesian networks, in: Proceedings of the Seventh International Conference on Uncertainty in Artificial Intelligence, 1991, pp. 52–60.

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035

[9] C.N. Chow, C.K. Liu, Approximating discrete probability distributions with dependence trees, IEEE Trans. Inf. Theory 14 (3) (1968) 462–467.
[10] Mark Coates, Alfred O. Hero III, Robert Nowak, Bin Yu, Internet tomography, IEEE Signal Process. Mag. 19 (3) (2002) 47–65.
[11] Arthur P. Dempster, Nan M. Laird, Donald B. Rubin, Maximum likelihood from incomplete data via the em algorithm, J. R. Stat. Soc. Ser. B Stat. Methodol. 39 (1) (1977) 1–38.
[12] Richard O. Duda, Peter E. Hart, Pattern Classification and Scene Analysis, Vol. 3, Wiley, New York, 1973.
[13] Nir Friedman, Learning belief networks in the presence of missing values and hidden variables, in: ICML, vol. 97, 1997, pp. 125–133.
[14] Nir Friedman, The Bayesian structural EM algorithm, in: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, 1998, pp. 129–138.
[15] Leszek Gasieniec, Tomasz Radzik, Graph-theoretic concepts in computer science, in: Chapter Memory Efficient Anonymous Graph Exploration, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 14–29.
[16] Vladimir Grebinski, Gregory Kucherov, Reconstructing a hamiltonian cycle by querying the graph: application to dna physical mapping, Discrete Appl. Math. 88 (1–3) (1998) 147–165.
[17] Vladimir Grebinski, Gregory Kucherov, Optimal reconstruction of graphs under the additive model, Algorithmica 28 (1) (2000) 104–124.
[18] Vincent Gripon, Michael Rabbat, Reconstructing a graph from path traces, in: Proceedings of International Symposium on Information Theory, July 2013.
[19] Patric Hagmann, Leila Cammoun, Xavier Gigandet, Reto Meuli, Christopher J. Honey, Van J. Wedeen, Olaf Sporns, Mapping the structural core of human cerebral cortex, PLoS biology 6 (7) (2008) e159.
[20] Jotun J Hein, An optimal algorithm to reconstruct trees from additive distance data, Bull. Math. Biol. 51 (5) (1989) 597–603.
[21] Valerie King, Li Zhang, Yunhong Zhou, On the complexity of distance-based evolutionary tree reconstruction, in: Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'03, Philadelphia, PA, USA, 2003, pp. 444–453.
[22] Edda Klipp, Ralf Herwig, Axel Kowald, Christoph Wierling, Hans Lehrach, Systems Biology in Practice: Concepts, Implementation and Application, John Wiley & Sons, 2008.
[23] Daphne Kollar, Nir Friedman, Probabilistic Graphical Models: Principles and Techniques, The MIT Press, 2009.
[24] Claire Mathieu, Hang Zhou, Graph reconstruction via distance oracles, in: ICALP (1), in: Lecture Notes in Computer Science, vol. 7965, Springer, 2013, pp. 733–744.
[25] Hanna Mazzawi, Optimally reconstructing weighted graphs using queries, in: Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'10, Philadelphia, PA, USA, 2010, pp. 608–615.
[26] Arvind Narayanan, Vitaly Shmatikov, De-anonymizing social networks, in: Security and Privacy, 2009 30th IEEE Symposium on, IEEE, 2009, pp. 173–187.
[27] Mark Newman, Albert-László Barabási, Duncan J Watts, The Structure and Dynamics of Networks, Princeton University Press, 2006.
[28] Lorenzo Orecchia, Zeyuan Allen Zhu, Flow-based algorithms for local graph clustering, in: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'14, 2014.
[29] Bernhard O. Palsson, Properties of reconstructed networks, in: Cambridge: Systems Biology, 2006.
[30] Judea Pearl, Thomas S. Verma, A theory of inferred causation, Stud. Logic Found. Math. 134 (1995) 789–811. Appeared in Proc. Conference on Knowledge Representation and Reasoning (KR) 1991.
[31] Michael G. Rabbat, Mário A.T. Figueiredo, Robert D. Nowak, Network inference from co-occurrences, IEEE Trans. Inf. Theory 54 (9) (2008) 4053–4068.
[32] Lev Reyzin, Nikhil Srivastava, Learning and verifying graphs using queries with a focus on edge counting, in: Proceedings of the 18th international conference on Algorithmic Learning Theory, ALT'07, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 285–297.
[33] Ronitt Rubinfeld, Gil Tamir, Shai Vardi, Ning Xie, Fast local computation algorithms. in ICS, 2011, pp. 223–238.
[34] David J. Spiegelhalter, Steffen L. Lauritzen, Sequential updating of conditional probabilities on directed graphical structures, Networks 20 (5) (1990) 579–605.
[35] Daniel A. Spielman, Shang-Hua Teng, Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems, in: Proceedings of the Thirty-sixth Annual ACM Symposium on Theory of Computing, STOC'04, 2004, pp. 81–90.
[36] Olaf Sporns, Dante R. Chialvo, Marcus Kaiser, Claus C. Hilgetag, Organization, development and function of complex brain networks, Trends Cogn. Sci. 8 (9) (2004) 418–425.
[37] Thomas Verma, Judea Pearl, An algorithm for deciding if a set of observed independencies has a causal explanation, in: Proceedings of the Eighth International Conference on Uncertainty in Artificial Intelligence, 1992, pp. 323–330.
[38] Stanley Wasserman, Social Network Analysis: Methods and Applications, Vol. 8, Cambridge university press, 1994.
[39] Zeyuan Allen Zhu, Silvio Lattanzi, Vahab Mirrokni, A local algorithm for finding well-connected clusters, in: Proceedings of the 30th International Conference on Machine Learning, ICML'13, 2013.

Please cite this article in press as: S. Micali, Z.A. Zhu, Reconstructing Markov processes from independent and anonymous experiments, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.06.035