Optimal Cover Time for a Graph-Based Coupon Collector Process

Nedialko B. Dimitrov[*,a,1], C. Greg Plaxton[b,2]

[a]Naval Postgraduate School 1 University Circle, Monterey, CA, 93943
[b]University of Texas at Austin 1 University Station, Austin, TX, 78712

Abstract

In this paper we study the following covering process defined over an arbitrary directed graph. Each node is initially uncovered and is assigned a random integer rank drawn from a suitable range. The process then proceeds in rounds. In each round, a uniformly random node is selected and its lowest-ranked uncovered outgoing neighbor, if such exists, is covered. We prove that if each node has in-degree $\Theta(d)$ and out-degree $O(d)$, then with high probability, every node is covered within $O(n + \frac{n \log n}{d})$ rounds, matching a lower bound due to Alon. A special case of our result is that for any $\Theta(\log n)$-regular graph and a small rank range of $\Theta(\log n)$, every node is covered within $\Theta(n)$ rounds. Alon has also shown that, for a certain class of d-regular expander graphs, the upper bound holds no matter what method is used to choose the uncovered neighbor. In contrast, we show that for arbitrary d-regular graphs, the method used to choose the uncovered neighbor can affect the cover time by more than a constant factor.

Key words: Coupon Collector Process, Graph Covering, Delay Sequence Argument

1. Introduction

One of the most commonly discussed stochastic processes in computer science is the so-called coupon collector process [1]. In that process, there are n distinct coupons and we proceed in rounds, drawing one uniformly random coupon (with replacement) in each round. Suppose we place a mark on each coupon we draw, if we have not seen the coupon before. Are $O(n)$ rounds sufficient to place marks on all of the coupons? Put differently, is picking coupons with replacement as efficient, to within a constant factor, as picking them without replacement? No, it is a well-known fact that with high probability the number of rounds required to mark all of the coupons is $\Theta(n \log n)$.

This shortcoming has motivated Adler et al. [2] and Alon [3] to study a similar graph-based covering process. The nodes of the graph represent the coupons and covering a node represents marking a coupon. In each round, a uniformly random node w is selected. If an uncovered neighbor of w exists, choose one such uncovered neighbor and cover it. We refer to this process as process CC.

[*] Corresponding author
Email addresses: ned@nps.edu (Nedialko B. Dimitrov), plaxton@cs.utexas.edu (C. Greg Plaxton)
[1] Supported by an MCD Fellowship from the University of Texas at Austin.
[2] Supported by NSF Grants CCR–0310970 and ANI–0326001.

Process CC can use a variety of different *covering methods* to decide which uncovered neighbor to cover. If our ultimate goal is to minimize cover time, certainly the most powerful covering method available is an offline method with knowledge of the entire sequence of node selections and with infinite computing power. We refer to this powerful cover time minimizing version of process CC as process MIN. To achieve our $O(n)$ goal, it is natural to consider $\log n$-regular graphs since the work of Alon implies process MIN has an expected cover time of $\Omega(n + \frac{n \log n}{d})$ rounds on d-regular graphs [3].

1.1. Logarithmic-Degree Graphs

Another natural version of process CC — in which the covering method chooses a uniformly random uncovered neighbor, if any — was studied by Adler et al. [2] and by Alon [3]; we refer to this version of process CC as process UNI. Alon shows that for logarithmic-degree Ramanujan expander graphs, process UNI completes in $O(n)$ time, matching the lower bound for process MIN.

Adler et al. show that for the hypercube, which has a weak expansion property but is not an expander, process UNI takes $O(n)$ time, also matching the lower bound for process MIN [2]. They also show that for arbitrary logarithmic-degree graphs, process UNI completes in $O(n \log \log n)$ time. Furthermore, Adler et al. present an application of process UNI to load balancing in a hypercubic distributed hash table (DHT).

A process that is intuitively similar to process UNI is one where we initially assign a rank to each node using a uniformly random permutation of the nodes, and the covering method covers the minimum-rank uncovered neighbor, if any. We refer to this permutation-based version of process CC as process P-RANK. In this paper, we show that process P-RANK completes in $O(n)$ time on arbitrary logarithmic-degree graphs.

In fact, we analyze a more general and local version of process CC in which each node initially chooses a uniformly random rank in a suitable range, and the covering method covers the minimum-rank uncovered neighbor of the selected node. (The explicit range requirements are stated in Lemma 24 and a simplified version is presented in Theorem 25. We assume that the nodes are numbered from 1 to n, and that ties in rank are broken in favor of the lower-numbered node.) We refer to this random rank version of process CC as process R-RANK. We show that the more general and local process R-RANK completes in $O(n)$ time on arbitrary logarithmic-degree graphs (see Theorem 25).

1.2. Results for General Graphs

Alon shows that process MIN on any d-regular graph has expected cover time at least $n - \frac{n}{d} + \frac{n}{d}\ln(\frac{n}{d})$ [3]. Alon also shows that process UNI completes in time $n + (1 + o(1))\frac{n \ln n}{d}$ for random nearly d-regular graphs. Alon further shows that on any (n, d, λ)-expander graph the expected cover time of process UNI is at most $n + n(\frac{\lambda}{d})^2(\ln n + 1)$. In particular, this implies that on Ramanujan graphs process UNI completes in $(1 + o(1))n$ time, matching the lower bound for process MIN.

If our goal is to maximize cover time, certainly the most powerful covering method available is an offline adversary with knowledge of the entire sequence of node selections and with infinite computing power. We refer to this powerful cover time maximizing version of process CC as process MAX. Alon notes that the upper bounds for expanders hold even if after every round an adversary "is allowed to shift the uncovered nodes to any place he wishes, keeping their number." In particular, this shows that on Ramanujan graphs, the cover time for process MAX matches the cover time for process MIN, up to constant factors. In effect, the covering method does not matter for this class of graphs.

Another previously studied variant of process CC favors covering the selected node. In this variant, we check — immediately after selecting a uniformly random node — if the selected node is uncovered. If it is, we cover it and move to the next selection. Only otherwise do we consider the neighbors of the selected node. We refer to the selection-biased variants of processes process UNI, process P-RANK, and process R-RANK as process UNI′, process P-RANK′, and process R-RANK′, respecively.

Adler et al. show that for all d-regular graphs, processes UNI and UNI′ finish in $O(n + n(\log n)(\log d)/d)$ time[2]. They also show that for random d-regular graphs only $O(n + \frac{n \log n}{d})$ steps are needed. Furthermore, they exhibit an application of process UNI′ to load balancing in DHTs.

All of the results matching Alon's lower bound for process MIN presented prior to this work have used some expansion properties of the underlying graph. In contrast, our proof techniques do not require the underlying graph to have any particular structure. Thus, we show the following general result: for directed graphs, with self-loops but no parallel edges, where each node has in-degree at least δ_{in} and at most Δ_{in}, and out-degree at most Δ_{out}, both process R-RANK and process R-RANK′ cover all nodes in $O(n \max(\Delta_{\text{in}}\Delta_{\text{out}}/\delta_{\text{in}}^2, (\log n)/\delta_{\text{in}}))$ rounds with high probability (see Theorems 25 and 27). This result matches Alon's lower bound for $\delta_{\text{in}} = \Delta_{\text{in}} = \Delta_{\text{out}} = \Theta(d)$, and is thus optimal under these conditions.

Furthermore, Alon's results for Ramanujan graphs raise the question whether there is any separation between the cover times for process MAX and process MIN. In other words, are there any graphs for which the choice of covering method matters? We define a weakly adversarial process, process A-RANK, that is similar to process P-RANK. In process A-RANK, instead of picking a uniformly random permutation, an adversary is initally allowed to fix the permutation used to assign ranks to the nodes. We then proceed as in process P-RANK. In addition, we define the selection-biased variant of process A-RANK as process A-RANK′. We establish that there exists a logarithmic-degree graph on which process A-RANK and process A-RANK′ each take $\omega(n)$ rounds to complete (see Theorems 28 and 29). This implies that in general there is separation between the cover times of process MIN and process MAX. In other words, the covering method does matter.

1.3. High-level Approach

The proof of our theorem is inspired by the delay sequence argument, introduced by Upfal [4], and used by Ranade for the analysis of a certain packet routing problem on the butterfly [5] (see also [6, Section 3.4.4]). In a delay sequence argument, we identify certain combinatorial structures that exist whenever the random process lasts for a long time. Then, we show that the probability any of these structures exist is small. This in turn implies an upper bound on the running time of the random process. The proof techniques for cover times in coupon collecting problems, in general, bear similarities to proof techniques for cover times of random walks [7, pp.144-145].

There are significant differences between our proof and that of Ranade. For example, in our problem, the connection between the running time and the length of a delay sequence is not clear-cut, while in the butterfly routing problem analyzed by Ranade, the length of the delay sequence is equal to the running time. But let us begin by giving the notion of a delay sequence in our problem.

Consider the node that was covered last, say w_1. Why wasn't w_1 covered earlier? It was not covered earlier because at the last opportunity to cover w_1 — that is, the last selection in w_1's neighborhood — we covered some other node, w_2, instead. In such a case we consider w_1 to be delayed by w_2. Similarly, w_2 may be delayed by some node w_3, et cetera, until finally we reach a node w_k that is not delayed, i.e., w_k is covered at the first opportunity. The sequence of nodes $w_1, \ldots, w_k$ corresponds to our notion of a delay sequence.

In analyzing process R-RANK, we find it useful to first analyze a much simpler process, process SELECT, in which we repeatedly select a uniformly random node, never covering anything. After establishing several lemmas for the simpler process, we proceed to analyzing process R-RANK. This is the bulk of the proof, and includes a technical lemma to work around the difficulties in linking cover time to delay sequence length. Finally, we reduce process R-RANK′ to process R-RANK to show that the same bounds hold.

The rest of this paper is structured as follows. In Section 2, we present a number of useful but standard definitions and lemmas related to probability distributions. In Section 3, we analyze the simple process, process SELECT. In Section 4, we layer ontop of Section 3 to analyze process R-RANK. In Section 5, we layer ontop of Section 4 to analyze the biased version, process R-RANK′. In Section 6, we show the existence of a $\log(n)$-regular graph on which process A-RANK and process A-RANK′ each take $\omega(n)$ rounds to complete, establishing that the neighbor selection method does matter. We conclude the paper with Section 7, where we provide different yet equivalent views of the processes discussed here as well as discuss several remaining open problems.

2. Preliminaries

In this section we introduce a few basic definitions and some standard probabilistic facts used throughout the paper's analysis. We use the term ℓ-sequence to refer to a sequence of length ℓ. For any ℓ-sequence σ of elements of a given type, and any element x of the same type, we let $\sigma : x$ denote the $(\ell + 1)$-sequence obtained by appending element x to σ.

For any nonnegative integer n and probability p, we use the notation $X \sim \mathrm{Bin}\,(n, p)$ to denote that the random variable X has a binomial distribution with n trials and success probability p. Similarly, we write $X \sim \mathrm{Geo}\,(p)$ to indicate that the random variable X has a geometric distribution with success probability p, and we write $X \sim \mathrm{NegBin}\,(r, p)$ to indicate that the random variable X has a negative binomial distribution with r successes and success probability p.

Lemma 1. *Let p denote an arbitrary probability, let ℓ denote an arbitrary nonnegative integer, and let $X \sim \text{NegBin}(\ell, p)$. For any integer j such that $1 \leq j \leq \ell$, let p_j denote an arbitrary probability such that $p_j \geq p$, let $Y_j \sim \text{Geo}\left(p_j\right)$, and let $Y = \sum_{1 \leq j \leq \ell} Y_j$. Then for any nonnegative integer i, $\Pr(X \geq i) \geq \Pr(Y \geq i)$.*

PROOF. Note that if $p_j = p$ for all j, then the random variables X and Y have the same distribution. Furthermore, increasing any of the p_j's can only decrease Y.

Lemma 2. *For any nonnegative integers r and n, and any probability p, we have $\Pr(X < r) = \Pr(Y > n)$, where $X \sim \text{Bin}(n, p)$ and $Y \sim \text{NegBin}(r, p)$.*

PROOF. The random variables X and Y can be seen as different views of the same experiment where we successively flip coins with probability of success p. With Y, we ask "How many flips are required for r successes?" With X, we ask "How many successes are in the first n flips?" In this experiment, the event of seeing less than r successes in the first n flips ($X < r$) corresponds to the event that we have to wait more than n flips for the first r successes ($Y > n$). This gives the result.

Lemma 3. *For any integer $r \geq 2$, we have $\Pr(X \geq 2E[X]) = \Pr(X \geq 2r/p) \leq \exp(-r/8)$, where $X \sim \text{NegBin}(r, p)$.*

PROOF. Let $j = \left\lfloor \frac{2r}{p} \right\rfloor - 1$ and let $Y \sim \text{Bin}(j, p)$. By Lemma 2, we know that $\Pr(X \geq \frac{2r}{p}) \leq \Pr(X \geq \left\lfloor \frac{2r}{p} \right\rfloor) = \Pr(X > \left\lfloor \frac{2r}{p} \right\rfloor - 1) = \Pr(Y < r) = \Pr(Y \leq r - 1)$.

$$\begin{aligned} \Pr\left(Y \leq \frac{jp}{2}\right) &= \Pr\left(Y \leq r - (\eta + 1)\frac{p}{2}\right) \\ &= \Pr(Y \leq r - 1) \end{aligned}$$

where $\frac{2r}{p} = \left\lfloor \frac{2r}{p} \right\rfloor + \eta$ and the last equality holds because $0 < (\eta + 1)\frac{p}{2} < 1$.

Recall the Chernoff bounds in the form $\Pr(Y \leq (1 - \lambda)jp) \leq \exp(-\lambda^2 jp/2)$ for $0 < \lambda < 1$ (see [8, 9]).

We apply this bound with $\lambda = \frac{1}{2}$ to get

$$\begin{aligned} \Pr(Y \leq r - 1) &= \Pr(Y \leq jp/2) \\ &\leq \exp(-jp/8) \\ &\leq \exp\left(\frac{-2r + (\eta + 1)p}{8}\right) \\ &\leq \exp(-r/8) \end{aligned}$$

where η is as previously defined and the last inequality holds because $r \geq 2$.

Lemma 4. *Let p be an arbitrary probability and let X be the sum of n independent Bernoulli variables $X_1, \ldots, X_n$, where X_j has success probability $p_j \geq p$. Then $\Pr(X \leq np/2) \leq \exp(-np/12)$.*

PROOF. The result follows from Chernoff bounds (see, e.g., [8, 9]).

Lemma 5. *Suppose we repeatedly throw balls independently and uniformly at random into n bins, and let the random variable X denote the number of throws required for every bin to receive at least n balls. Then X is $O(n^2)$ with high probability, that is, with failure probability that is an arbitrary inverse polynomial in n.*

PROOF. Let Y_i be the number of balls in bin i after $12n^2$ throws. The probability that the ball of any single throw falls in bin i is $\frac{1}{n}$. The number of balls in bin i after $12n^2$ throws, Y_i, is the sum of $12n^2$ Bernoulli random variables. By Lemma 4, we have $\Pr\left(Y_i \leq 12n^2/2n\right) \leq \exp(-12n^2/12n)$. Simplifying, we have $\Pr(Y_i \leq 6n) \leq \exp(-n)$.

We can now define n bad events, one for each of the bins, stating that the bin receives less than $6n$ balls in $12n^2$ throws. The result of the previous paragraph and a union bound on the n bad events show that the probability any bin receives less than $6n$ balls is at most $n\exp(-n)$. Thus, with $O(n^2)$ throws, each bin receives at least n balls with high probability.

Lemma 6. *Let j balls be thrown independently and uniformly at random into n bins. Let X denote the number of bins with at least one ball at the end of the experiment. Then,* $\Pr(X \leq \min(n/4, j/4)) \leq \exp(-j/2)$.

PROOF. Let $[n] = \{1, 2, \ldots, n\}$. Suppose $\min\left(\frac{n}{4}, \frac{j}{4}\right) = k$. Let $S \subseteq [n]$ be a particular subset of size k. Then,

$$\Pr(\text{all balls land in } S) \leq \left(\frac{k}{n}\right)^j$$

Thus,

$$\begin{aligned}
\Pr(X \leq k) &= \Pr\left(\bigcup_{S \text{ s.t. } |S|=k} \text{all balls land in } S\right) \\
&\leq \binom{n}{k}\left(\frac{k}{n}\right)^j \\
&\leq \left(\frac{en}{k}\right)^k \left(\frac{k}{n}\right)^j \\
&= \left(\frac{en}{k}\right)^k \left(\frac{k}{n}\right)^{\frac{j}{2}} \left(\frac{k}{n}\right)^{\frac{j}{2}}
\end{aligned}$$

Now, since $\frac{j}{2} \geq 2k$ and since $k \leq \frac{n}{4}$ implies $\frac{ek}{n} \leq \frac{e}{4} < 1$

$$\begin{aligned}
\Pr(X \leq k) &\leq \left(\frac{ek}{n}\right)^k \left(\frac{k}{n}\right)^{\frac{j}{2}} \\
&\leq \left(\frac{ek}{n}\right)^k \left(\frac{1}{4}\right)^{\frac{j}{2}} \\
&\leq \exp\left(-\frac{j}{2}\right)
\end{aligned}$$

3. Process SELECT

Throughout the remainder of the paper, we fix an arbitrary directed graph $G = (V, E)$ where $|V| = n > 0$. We say that an event holds "with high probability" if the probability that it fails to occur is upper bounded by an arbitrary inverse polynomial in n. We let δ_{in}, Δ_{in}, and Δ_{out} denote the minimum in-degree, maximum in-degree, and maximum out-degree of any node, respectively. For ease of exposition, we assume throughout the paper that $\delta_{\text{in}} > 0$. The edge set E is allowed to contain loops but not parallel edges. For any node v, we define $\Gamma_{\text{in}}(v)$ as $\{w \mid (w, v) \in E\}$, intuitively the in-neighborhood of v. For any sequence of edges $\sigma = (u_1, v_1), \ldots, (u_\ell, v_\ell)$, we define the two sequences of nodes $\text{src}(\sigma) = u_1, \ldots, u_\ell$, intuitively the source nodes, and $\text{dst}(\sigma) = v_1, \ldots, v_\ell$, intuitively the destination nodes.

In this section, we analyze a simple stochastic process, process SELECT. The main result used in the remainder of the paper from this section is Lemma 11. We define process SELECT as follows. Initially, we fix a positive integer r and independently assign each node in V a uniformly random integer rank between 1 and r. Process SELECT then proceeds in an infinite number of rounds, indexed from 1. In each round, one node is selected uniformly at random, with replacement. To analyze the defined process, process SELECT, the following definitions are central.

A node sequence is said to be *rank-sorted* if the associated sequence of node ranks is nondecreasing.

For any node sequence σ, we inductively define duration(σ), a nonnegative integer, and a node sequence select(σ) as follows. If σ is empty, then duration(σ) is 0 and select(σ) is empty. Otherwise, σ is of the form $\tau : v$ for some shorter node sequence τ and node v. Let i denote the least i such that $i >$ duration(τ) and the node selected in round i belongs to $\Gamma_{\text{in}}(v)$. Let u denote the node selected in round i. Then we define duration(σ) as i, and select(σ) as select(τ) : u. Intuitively in the definition of duration(σ), we are waiting for every node in the sequence — in order — to receive a single selection within its in-neighborhood. Intuitively, select(σ) is simply the sequence of nodes selected within the in-neighborhoods of σ.

Lemma 7. *For any ℓ-sequence of distinct nodes σ,* $\Pr(\sigma$ *is rank-sorted*$) = \binom{\ell+r-1}{\ell} r^{-\ell}$.

PROOF. There are $\binom{\ell+r-1}{\ell}$ ways that ranks can be assigned to the ℓ distinct nodes so that the resulting ℓ-sequence is rank-sorted. The result follows since each such assignment occurs with probability $r^{-\ell}$.

Lemma 8. *For any ℓ-sequence of nodes $\sigma = v_1, \ldots, v_\ell$ and any nonnegative integer i, we have* $\Pr(\text{duration}(\sigma) = i) \leq \Pr(X \geq i)$*, where* $X \sim \text{NegBin}\left(\ell, \frac{\delta_{\text{in}}}{n}\right)$.

PROOF. We proceed by proving that

$$\Pr(\text{duration}(\sigma) = i) \quad = \quad \Pr\left(\sum_{k=1}^{\ell} Y_k = i\right)$$

where $Y_k \sim \text{Geo}\left(\frac{d_k}{n}\right)$ and d_k denotes the in-degree of v_k. The desired bound then follows by Lemma 1.

We prove the foregoing claim by induction on ℓ. If $\ell = 0$, the claim holds since $\text{duration}(\sigma) = \sum_{k=1}^{\ell} Y_k = 0$.

For $\ell > 0$, we let τ denote the node sequence $v_1, \ldots, v_{\ell-1}$ and assume inductively that

$$\Pr(\text{duration}(\tau) = i) \quad = \quad \Pr\left(\sum_{k=1}^{\ell-1} Y_k = i\right).$$

Thus,

$$\begin{aligned}
&\Pr(\text{duration}(\sigma) = i)\\
&\quad= \sum_{j=0}^{i-1} \Pr(\text{duration}(\tau) = j) \cdot\\
&\qquad\qquad \Pr(\text{duration}(\sigma) - \text{duration}(\tau) = i - j \mid \text{duration}(\tau) = j)\\
&\quad= \sum_{j=0}^{i-1} \Pr(\text{duration}(\tau) = j) \cdot \Pr(\text{duration}(\sigma) - \text{duration}(\tau) = i - j)\\
&\quad= \sum_{j=0}^{i-1} \Pr(\text{duration}(\tau) = j) \cdot \Pr(Y_\ell = i - j)\\
&\quad= \sum_{j=0}^{i-1} \Pr\left(\sum_{k=1}^{\ell-1} Y_k = j\right) \cdot \Pr(Y_\ell = i - j)\\
&\quad= \Pr\left(\sum_{k=1}^{\ell} Y_k = i\right).
\end{aligned}$$

The second equality holds because each selection is independent of previous selections. The third equality holds because the waiting time to obtain a selection in $\Gamma_{\text{in}}(v_\ell)$ is distributed as Y_ℓ.

Lemma 9. *For any ℓ-sequence of edges σ,* $\Pr(\text{select}(\text{dst}(\sigma)) = \text{src}(\sigma)) \le \delta_{\text{in}}^{-\ell}$.

Proof. We proceed by induction on ℓ. For $\ell = 0$, $\Pr(\text{select}(\text{dst}(\sigma)) = \text{src}(\sigma)) = 1 = \delta_{\text{in}}^{0}$ since we have assumed that $\delta_{\text{in}} > 0$.

For $\ell > 0$, σ can be written in the form $\tau : (u, v)$, where we inductively assume that the claim of the lemma holds for τ. Let A denote the event that the first node selected in $\Gamma_{\text{in}}(v)$ after round duration(dst(τ)) is u. We have

$$\begin{aligned}
&\Pr(\text{select}(\text{dst}(\sigma)) = \text{src}(\sigma))\\
&\quad= \Pr(\text{select}(\text{dst}(\tau)) = \text{src}(\tau)) \cdot \Pr(A \mid \text{select}(\text{dst}(\tau)) = \text{src}(\tau))\\
&\quad= \Pr(\text{select}(\text{dst}(\tau)) = \text{src}(\tau)) \cdot \Pr(A)\\
&\quad\le \delta_{\text{in}}^{-\ell}.
\end{aligned}$$

The second step follows from the independence of the events A and $\text{select}(\text{dst}(\tau)) = \text{src}(\tau)$. (These two events are independent since each selection is independent of previous selections.) The third step follows from the induction hypothesis and the observation that $\Pr(A)$ is equal $1/\Gamma_{\text{in}}(v)$, which is at most $1/\delta_{\text{in}}$.

Lemma 10. *For any ℓ-sequence of edges σ and nonnegative integer i, the three events specified as A = "dst(σ) is rank-sorted", B = "duration(dst(σ)) = i", and C = "select(dst(σ)) = src(σ)" are mutually independent.*

Proof. Note that event A depends only on the rank assignments, while events B and C depend only on the selections. Thus event A is independent of events B and C. Below we argue that events B and C are independent.

Let $\sigma = (u_1, v_1), \ldots, (u_\ell, v_\ell)$ and let σ_j denote the length-j prefix of σ, $0 \leq j \leq \ell$. Define a selection to be *j-special*, $1 \leq j \leq \ell$, if it is the first selection after round duration(σ_{j-1}) in $\Gamma_{\text{in}}(v_j)$. A selection is *special* if it is j-special for some j. Note that event B depends only on the timing of the special events; in particular, B occurs if and only if the ℓ-special selection occurs in round i. Suppose we run process SELECT, but at each step, instead of revealing the selected node, we reveal only whether the selection is special. This information is sufficient to determine the unique i for which B occurs, but does not bias the distribution of select(dst(σ)). Since event C only depends on select(dst(σ)), it is independent of B.

Lemma 11. *Let σ be an ℓ-sequence of edges so that the nodes of* dst(σ) *are distinct, let $X \sim \text{NegBin}\left(\ell, \frac{\delta_{\text{in}}}{n}\right)$, let i be a nonnegative integer, and let events A, B, and C be defined as in the statement of Lemma 10. Then* $\Pr(A \cap B \cap C) \leq \binom{\ell+r-1}{\ell} \cdot \Pr(X \geq i) \cdot (r\delta_{\text{in}})^{-\ell}$.

Proof. By Lemma 7, $\Pr(A) \leq \binom{\ell+r-1}{\ell} r^{-\ell}$. By Lemma 8, $\Pr(B) \leq \Pr(X \geq i)$. By Lemma 9, $\Pr(C) \leq \delta_{\text{in}}^{-\ell}$. The claim then follows by Lemma 10.

4. Process R-RANK

In the section we analyze an augmented version of process SELECT, referred to as Process R-RANK. The main result of this section is Theorem 25, and the main techincal lemma is Lemma 22. We define Process R-RANK by augmenting process SELECT with the notion of a "covered subset" of the nodes. Initially, all of the nodes are uncovered. Process R-RANK then proceeds in rounds in exactly the same manner as process SELECT, except that in any given round, if one or more outgoing neighbors of the selected node are uncovered, we cover the uncovered outgoing neighbor with minimum rank. (As indicated in Section 1, ties are broken according to some arbitrary numbering of the nodes.)

Note that process R-RANK simply augments process SELECT by also covering nodes; rank assignment and selections are performed in exactly the same manner in the two processes. Thus all of the definitions and lemmas presented in Section 3 are applicable to process R-RANK. The following additional definitions are useful for our analysis of process R-RANK.

The *cover time* of process R-RANK is defined as the number of rounds required to cover all of the nodes.

We inductively define the notion of a *linked* sequence of edges. For ℓ equal to 0 or 1, any ℓ-sequence of edges is linked. For $\ell > 1$, an ℓ-sequence of edges of the form $\sigma : (u, v) : (u', v')$ is linked if the $(\ell - 1)$-sequence $\sigma : (u, v)$ is linked and (u, v') belongs to E. Intuitively, if the selection of u covers v, and the selection of u' covers v', for a linked sequence, the selection of u also provided an opportunity to cover v'.

For any node v, we define parent(v) as follows. Let i denote the round in which node v is covered. If i is the first round in which some node in $\Gamma_{\text{in}}(v)$ is selected, then parent(v) is defined to be nil. Otherwise, parent(v) is the node covered in the first round prior to round i in which the selected node belongs to $\Gamma_{\text{in}}(v)$. Intuitively, the parent of v is the node covered at the previous opportunity to cover v.

We inductively define the notion of a *chronological* sequence of nodes as follows. Any ℓ-sequence of nodes with $\ell \leq 1$ is chronological. An ℓ-sequence of nodes of the form $\sigma : v : v'$ is chronological if $\sigma : v$ is chronological and node v is covered before node v'. Intuitively, a chronological sequence of nodes is covered in the same order as the nodes appear in the sequence.

We inductively define the notion of an *active* node sequence as follows. The empty node sequence is active. A singleton node sequence consisting of the node v is active if $\text{parent}(v) = \text{nil}$. An ℓ-sequence of nodes of the form $\sigma : v : v'$ is active if $\sigma : v$ is active and $\text{parent}(v') = v$. Intuitively, active sequences act as delay sequences in our argument. More specifically, an active sequence presents a sequence of cover delays, where the covering of v' is delayed by v.

We call an ℓ-sequence of edges σ *active* if $\text{dst}(\sigma)$ is active and $\text{select}(\text{dst}(\sigma)) = \text{src}(\sigma)$.

We call an ℓ-sequence of edges σ *i-active* if it is active and either $\ell = i = 0$ or $\ell > 0$, σ is of the form $\sigma : (u, v)$, and v is the node covered in round i. Intuitively, an i-active sequence delays the completion of covering all nodes until round i.

Lemma 12. *For any nonnegative integer ℓ, there are at most $n\Delta_{\text{out}}^{\ell}\Delta_{\text{in}}^{\ell-1}$ linked ℓ-sequences of edges.*

Proof. We proceed by induction on ℓ, treating $\ell = 0$ and $\ell = 1$ as the base cases. For $\ell = 0$, the empty sequence is the only linked 0-sequence, and the claim holds since $n/\Delta_{\text{in}} \geq 1$. (Note that Δ_{in} is at most n since we do not allow parallel edges.) For $\ell = 1$, the number of linked 1-sequences is at most $|E| \leq n\Delta_{\text{out}}$.

Now let ℓ be greater than 1 and inductively assume that the number of linked $(\ell - 1)$-sequences of edges is at most $n\Delta_{\text{out}}^{\ell-1}\Delta_{\text{in}}^{\ell-2}$. Recall that any linked ℓ-sequence of edges is of the form $\sigma : (u, v) : (u', v')$ where the $(\ell - 1)$-sequence of edges $\sigma : (u, v)$ is linked and (u, v') belongs to E. Observe that for any linked $(\ell - 1)$-sequence of edges $\sigma : (u, v)$, there are at most Δ_{out} nodes v' such that (u, v') belongs to E, and for each such choice of v', there are at most Δ_{in} nodes u' such that (u', v') belongs to E. Thus the number of linked ℓ-sequences is at most $\Delta_{\text{out}}\Delta_{\text{in}}$ times the number of linked $(\ell - 1)$-sequences, and the desired bound follows from the induction hypothesis.

Lemma 13. *Suppose we run two instances of process R-RANK in parallel using the same random ranks and the same sequence of random selections, but in the second instance, we allow an arbitrary subset of the covered nodes to be uncovered after each round. Then the cover time of the first instance is at most the cover time of the second instance.*

Proof. By a straightforward induction on the number of rounds, at all times, the set of covered nodes in the first instance contains the set of covered nodes in the second instance. The claim of the lemma follows.

Lemma 14. *For any rank assignment, the expected cover time of process R-RANK is $O(n^2)$.*

Proof. It follows from Lemma 5 that the cover time is $O(n^2)$ with high probability since in that time each vertex is selected at least n times, implying that all of its neighbors are covered.

We can then consider a modified version of process R-RANK in which the infinite sequence of rounds is partitioned into epochs of $O(n^2)$ rounds, and where at the end of each epoch, if the nodes are not all covered, all nodes are uncovered before proceeding to the next epoch. Since each epoch covers all the nodes with high probability, the expected cover time of this modified version of process R-RANK is $O(n^2)$. By Lemma 13, for any rank assignment, the expected cover time of process R-RANK is $O(n^2)$.

Lemma 15. *Assume that v is the node covered in round i and let u be the node selected in round i. Then there is an i-active edge sequence σ terminating in edge (u, v) and such that* $\mathrm{duration}(\mathrm{dst}(\sigma)) = i$.

PROOF. Observe that u belongs to $\Gamma_{\mathrm{in}}(v)$. Furthermore, if $\mathrm{parent}(v) = \mathrm{nil}$, then the singleton node sequence v is active with $\mathrm{duration}(v) = i$. Thus the singleton edge sequence $\sigma = (u, v)$ is i-active with $\mathrm{duration}(\mathrm{dst}(\sigma)) = i$.

We prove the claim by induction on i. For $i = 1$, we have $\mathrm{parent}(v) = \mathrm{nil}$ and so the claim follows by the observations of the previous paragraph.

For $i > 1$, if $\mathrm{parent}(v) = \mathrm{nil}$, the claim once again follows from the foregoing observations. Otherwise, $\mathrm{parent}(v) = v'$ where v' is the node covered in round j with $j < i$. Let u' denote the node selected in round j. Since $j < i$, we can inductively assume that there is a j-active edge sequence, call it τ, terminating in edge (u', v') and such that $\mathrm{duration}(\mathrm{dst}(\tau)) = j$. Since τ is active, the node sequence $\mathrm{dst}(\tau)$ is active and $\mathrm{select}(\mathrm{dst}(\tau)) = \mathrm{src}(\tau)$. Let $\sigma = \tau : (u, v)$. Thus $\mathrm{src}(\sigma) = \mathrm{src}(\tau) : u$ and $\mathrm{dst}(\sigma) = \mathrm{dst}(\tau) : v$. Since $\mathrm{parent}(v) = v'$ and $\mathrm{dst}(\tau)$ is an active node sequence terminating in node v', $\mathrm{dst}(\sigma)$ is active. Since $\mathrm{duration}(\mathrm{dst}(\tau)) = j$, $\mathrm{select}(\mathrm{dst}(\tau)) = \mathrm{src}(\tau)$, u was selected in round i, and i is the least integer greater that j such that the node selected in round i belongs to $\Gamma_{\mathrm{in}}(v)$, we have $\mathrm{duration}(\mathrm{dst}(\sigma)) = i$ and $\mathrm{select}(\mathrm{dst}(\sigma)) = \mathrm{src}(\sigma)$. Since $\mathrm{dst}(\sigma)$ is active and $\mathrm{select}(\mathrm{dst}(\sigma)) = \mathrm{src}(\sigma)$, σ is active. Since σ is active and v is the node covered in round i, σ is i-active. Thus the edge sequence σ satisfies all of the requirements of the lemma.

Lemma 16. *Any active node sequence is rank-sorted, chronological, and consists of distinct nodes.*

PROOF. Note that any chronological node sequence consists of distinct nodes. Thus, in what follows, it is sufficient to prove that any active node sequence is rank-sorted and chronological.

We proceed by induction on the length of the sequence. For the base case, note that any node sequence of length 0 or 1 is rank-sorted and chronological. For the induction step, consider an active node sequence σ of the form $\tau : v : v'$. Since σ is active, $\tau : v$ is active and $\mathrm{parent}(v') = v$. Since $\tau : v$ is active, the induction hypothesis implies that it is also rank-sorted and chronological. Since $\mathrm{parent}(v') = v$, $\mathrm{rank}(v) \leq \mathrm{rank}(v')$ and v is covered before v'. Hence σ is rank-sorted and chronological.

Lemma 17. *For any nonempty active edge sequence σ, if the last edge in σ is (u, v), then v is the node covered in round* $\mathrm{duration}(\mathrm{dst}(\sigma))$ *and node u is selected in the same round.*

PROOF. We prove the claim by induction on the length of the active edge sequence σ.

If σ consists of a single edge (u, v), then by the definition of an active edge sequence, the singleton node sequence $\mathrm{dst}(\sigma)$ is active and $\mathrm{select}(\mathrm{dst}(\sigma)) = \mathrm{src}(\sigma)$. Since $\mathrm{dst}(\sigma)$ is active, $\mathrm{parent}(v) = \mathrm{nil}$, that is, v is the node covered in the first round in which a node in $\Gamma_{\mathrm{in}}(v)$ is selected, which is round $\mathrm{duration}(\mathrm{dst}(\sigma))$. Since $\mathrm{select}(\mathrm{dst}(\sigma)) = \mathrm{src}(\sigma)$, node u is selected in the same round.

Now assume that σ is an active edge sequence of the form $\tau : (u, v)$, where τ is of the form $\tau' : (u', v')$. Since σ is active, the node sequence $\mathrm{dst}(\sigma)$ is active and $\mathrm{select}(\mathrm{dst}(\sigma)) = \mathrm{src}(\sigma)$. It follows that $\mathrm{dst}(\tau)$ is active and $\mathrm{select}(\mathrm{dst}(\tau)) = \mathrm{src}(\tau)$, that is, τ is also active. Since τ is active and shorter than σ, we can inductively assume that v' is the node covered in round $\mathrm{duration}(\mathrm{dst}(\tau))$ and node u' is selected in the same round. Since $\mathrm{dst}(\sigma)$ is active, $\mathrm{parent}(v) = v'$, that is, v is the node covered in the first round after round $\mathrm{duration}(\mathrm{dst}(\tau))$ in which a node in $\Gamma_{\mathrm{in}}(v)$ is selected. Applying the definition of $\mathrm{duration}(\mathrm{dst}(\sigma))$, we conclude that v is the node covered in round $\mathrm{duration}(\mathrm{dst}(\sigma))$. Since $\mathrm{select}(\mathrm{dst}(\sigma)) = \mathrm{src}(\sigma)$, node u is selected in the same round.

Lemma 18. *If σ is an active sequence of edges, then σ is linked.*

PROOF. We proceed by induction on the length of σ. If the length of σ is 0 or 1, then σ is linked by definition.

Now assume that σ is an edge sequence of the form $\tau : (u, v)$, where τ is of the form $\tau' : (u', v')$ and σ is active. Since σ is active, $\mathrm{dst}(\sigma)$ is active. Since $\mathrm{dst}(\sigma)$ is active, $\mathrm{dst}(\tau)$ is also active. Since $\mathrm{dst}(\tau)$ is active and τ is shorter than σ, we can inductively assume that τ is linked. Therefore, in order to establish that σ is linked, it is sufficient to prove that (u', v) is an edge. Since $\mathrm{dst}(\sigma)$ is active, $\mathrm{parent}(v) = v'$. Hence, letting i denote the round in which node v is covered, we find that v' is the node covered in the first round prior to round i in which the selected node belongs to $\Gamma_{\mathrm{in}}(v)$. By Lemma 17, v' is covered in a round in which node u' is selected. Thus u' belongs to $\Gamma_{\mathrm{in}}(v)$, that is, (u', v) is an edge, as required.

Lemma 19. *If an edge sequence σ is i-active, then* $\mathrm{duration}(\mathrm{dst}(\sigma)) = i$.

PROOF. If σ is empty, then the claim holds since $i = 0$ and $\mathrm{duration}(\mathrm{dst}(\sigma)) = 0$. Otherwise, σ is of the form $\tau : (u, v)$, and by the definition of an i-active edge sequence, v is the node covered in round i. By Lemma 17, v is the node covered in round $\mathrm{duration}(\mathrm{dst}(\sigma))$, so $\mathrm{duration}(\mathrm{dst}(\sigma)) = i$.

Lemma 20. *For any ℓ-sequence of edges σ, and any nonnegative integer i, the probability that σ is i-active is at most $\binom{\ell+r-1}{\ell} \cdot \Pr(X \geq i) \cdot (r\delta_{\mathrm{in}})^{-\ell}$, where $X \sim \mathrm{NegBin}\left(\ell, \frac{\delta_{\mathrm{in}}}{n}\right)$.*

PROOF. If the nodes in $\mathrm{dst}(\sigma)$ are not all distinct, then $\Pr(\sigma \text{ is } i\text{-active}) = 0$ by Lemma 16 and the claimed inequality holds since the right-hand side is nonnegative.

Now assume that $\mathrm{dst}(\sigma)$ consists of distinct nodes, and let events A, B, and C be as defined in the statement of Lemma 10. Below we prove that if σ is i-active, then events A, B, and C all occur. The claimed inequality then follows by Lemma 11.

Assume that σ is i-active. Thus event B occurs by Lemma 19. Furthermore, σ is active, so $\mathrm{dst}(\sigma)$ is active and event C occurs by the definition of an active edge sequence. Since $\mathrm{dst}(\sigma)$ is active, event A occurs by Lemma 16.

Lemma 21. *For any nonnegative integers i and ℓ, the probability that some ℓ-sequence of edges is i-active is at most*

$$n\Delta_{\mathrm{out}}^{\ell}\Delta_{\mathrm{in}}^{\ell-1}\binom{\ell+r-1}{\ell}\frac{\Pr(X \geq i)}{(r\delta_{\mathrm{in}})^{\ell}}$$

where $X \sim \mathrm{NegBin}\left(\ell, \frac{\delta_{\mathrm{in}}}{n}\right)$.

Proof. By Lemma 18, if an edge sequence σ is not linked, then $\Pr(\sigma \text{ is } i\text{-active}) = 0$. A union bound then implies that the probability some ℓ-sequence of edges is i-active is at most the number of linked ℓ-sequences of edges multiplied by the maximum probability that any particular ℓ-sequence is i-active. The desired inequality then follows by Lemmas 12 and 20.

Lemma 22. *For nonnegative integers i, ℓ, and r satisfying the properties $i \geq 64n\max(\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}^2, (\ln n)/\delta_{\mathrm{in}})$ and $r \geq \min(\lceil 2e^2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}\rceil, \ell)$, we have*

$$\Delta_{\mathrm{out}}^{\ell}\Delta_{\mathrm{in}}^{\ell-1}\binom{\ell+r-1}{\ell}\frac{\Pr(X \geq i)}{(r\delta_{\mathrm{in}})^{\ell}} \quad \leq \quad \exp(-i\delta_{\mathrm{in}}/(32n))$$

where $X \sim \mathrm{NegBin}\left(\ell, \frac{\delta_{\mathrm{in}}}{n}\right)$.

Proof. First, we show that the LHS of the claimed inequality is a nonincreasing function of r.

It is sufficient to prove that the expression $\binom{\ell+r-1}{\ell}r^{-\ell}$ is a nonincreasing function of r. Fix ℓ and let $f(r)$ denote the preceding expression. Note that

$$\begin{aligned}\frac{f(r+1)}{f(r)} &= \frac{r+\ell}{r}\left(\frac{r}{r+1}\right)^{\ell} \\ &= \left(1+\frac{\ell}{r}\right)\left(1+\frac{1}{r}\right)^{-\ell} \\ &\leq 1,\end{aligned}$$

where the last inequality holds since the binomial theorem implies $(1+\frac{1}{r})^{\ell} \geq 1+\frac{\ell}{r}$.

Since we have established that the LHS of the claimed inequality is a nonincreasing function of r, we can assume that $r = \min(\lceil 2e^2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}\rceil, \ell)$.

Let us rewrite the LHS of the claimed inequality as $\lambda \cdot \Pr(X \geq i)$, where

$$\begin{aligned}\lambda &= \Delta_{\mathrm{out}}^{\ell}\Delta_{\mathrm{in}}^{\ell-1}\binom{\ell+r-1}{\ell}(r\delta_{\mathrm{in}})^{-\ell} \\ &\leq \Delta_{\mathrm{out}}^{\ell}\Delta_{\mathrm{in}}^{\ell}\left(\frac{e(\ell+r-1)}{\ell r\delta_{\mathrm{in}}}\right)^{\ell} \\ &\leq \left(\frac{e\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}(\ell+r)}{\ell r\delta_{\mathrm{in}}}\right)^{\ell}. \qquad (1)\end{aligned}$$

We begin by establishing two useful upper bounds on λ, namely, Equations (2) and (4) below.

If $r = \lceil 2e^2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}\rceil$, then since since $r = \min(\lceil 2e^2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}\rceil, \ell)$, we have $r \leq \ell$. Substituting the value of r into Equation (1), we find that

$$\begin{aligned} \lambda &\leq \left(\frac{e(\ell + r)}{2e^2\ell}\right)^{\ell} \\ &\leq \left(\frac{2e\ell}{2e^2\ell}\right)^{\ell} \\ &\leq e^{-\ell}. \end{aligned} \tag{2}$$

If $r = \ell$, then Equation (1) implies

$$\lambda \leq \left(\frac{2e\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}}{\ell\delta_{\mathrm{in}}}\right)^{\ell}. \tag{3}$$

Let $h(\ell)$ denote the natural logarithm of the RHS of Equation (3), that is, $h(\ell) = \ell\ln(2e\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/(\ell\delta_{\mathrm{in}}))$. Using elementary calculus, it is straightforward to prove that the derivative of $h(\ell)$ with respect to ℓ is positive for $\ell < 2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}$, is 0 when $\ell = 2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}$, and is negative for $\ell > 2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}$. It follows that $h(\ell) \leq h(2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}) = 2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}$. Since ln is monotonic, the RHS of Equation (3) is also maximized when $\ell = 2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}$. Combining this result with Equation (2), we find that for any r

$$\lambda \leq \exp(2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}). \tag{4}$$

(Note that $\exp(2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}) \geq 1$ and Equation (2) implies $\lambda \leq 1$ when $r = \lceil 2e^2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}\rceil$.)

We are now ready to proceed with the proof of the lemma. We consider the two cases $\ell > \lceil i\delta_{\mathrm{in}}/(2n)\rceil$ and $\ell \leq \lceil i\delta_{\mathrm{in}}/(2n)\rceil$ separately.

If $\ell > \lceil i\delta_{\mathrm{in}}/(2n)\rceil$, then $\ell > 2ec\max(\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}, \ln n)$ where $c = 16/e > e$. Thus $\ell > \lceil 2e^2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}\rceil$ and so $r = \lceil 2e^2\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}\rceil$. It follows from Equation (2) that $\lambda \leq e^{-\ell} \leq \exp(-i\delta_{\mathrm{in}}/(2n)) \leq \exp(-i\delta_{\mathrm{in}}/(64n))$, and hence the claim holds since $\Pr(X \geq i) \leq 1$.

Now assume that $\ell \leq \lceil i\delta_{\mathrm{in}}/(2n)\rceil$. Let $Y \sim \mathrm{NegBin}\left(\left\lfloor \frac{i\delta_{\mathrm{in}}}{2n}\right\rfloor, \frac{\delta_{\mathrm{in}}}{n}\right)$ and $Z \sim \mathrm{NegBin}\left(\left\lfloor \frac{i\delta_{\mathrm{in}}}{2n}\right\rfloor - \ell, \frac{\delta_{\mathrm{in}}}{n}\right)$. By the definition of the negative binomial distribution, $\Pr(Y \geq i) = \Pr(X + Z \geq i)$. And, since Z is nonnegative, $\Pr(X + Z \geq i) \geq \Pr(X \geq i)$. Thus

$$\Pr(X \geq i) \leq \Pr(Y \geq i). \tag{5}$$

Since $E[Y] \leq \frac{i}{2}$ and $\lfloor i\delta_{\mathrm{in}}/(2n)\rfloor \geq \lfloor 32\max(\Delta_{\mathrm{out}}\Delta_{\mathrm{in}}/\delta_{\mathrm{in}}, \ln n)\rfloor > 2$, Lemma 3 implies $\Pr(Y \geq i) \leq \Pr(Y \geq 2E[Y]) \leq$

$\exp\left(\frac{-i\delta_{\text{in}}}{16n}+\frac{1}{8}\right)$. The claim follows since

$$\begin{aligned}\lambda\cdot\Pr(X\geq i) &\leq \exp\left(\frac{2\Delta_{\text{out}}\Delta_{\text{in}}}{\delta_{\text{in}}}\right)\cdot\Pr(Y\geq i)\\ &\leq \exp\left(\frac{-i\delta_{\text{in}}}{16n}+\frac{1}{8}+\frac{2\Delta_{\text{out}}\Delta_{\text{in}}}{\delta_{\text{in}}}\right)\\ &\leq \exp\left(\frac{-i\delta_{\text{in}}}{32n}+\frac{1}{8}\right)\\ &\leq \exp\left(\frac{-i\delta_{\text{in}}}{64n}\right).\end{aligned}$$

(The first step follows from Equations (4) and (5). For the third step and fourth steps, note that the assumption $i\geq 64n\max(\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}^2,(\ln n)/\delta_{\text{in}})$ implies $i\delta_{\text{in}}/(32n)\geq 2\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}$ and $i\delta_{\text{in}}/(64n)\geq 1/8$, respectively.)

Lemma 23. *If $r\geq\min(\lceil 2e^2\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}\rceil,n)$, then every active edge sequence is $O(n\max(\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}^2,(\log n)/\delta_{\text{in}}))$-active, with high probability.*

PROOF. Let c denote an arbitrary positive real greater than or equal to 1, and let i denote the positive integer $\lceil 64cn\max(\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}^2,(\ln n)/\delta_{\text{in}}))\rceil$.

For any nonnegative integer j, let p_j denotes the probability that there is a j-active edge sequence. Any j-active edge sequence σ is active, so the associated node sequence $\text{dst}(\sigma)$ is active. It follows from Lemma 16 that any j-active sequence has length at most n. In other words, $\ell\leq n$ for any j-active ℓ-sequence of edges. Furthermore, if $j>0$ then the length of a j-active sequence is nonzero. Since any j-active ℓ-sequence of edges satisfies $\ell\leq n$, the condition $r=\min(\lceil 2e^2\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}\rceil,n)$ allows us to apply Lemmas 21 and 22. Applying these two lemmas, together with a union bound, we obtain $p_j\leq n^2\exp(-j\delta_{\text{in}}/(64n))$ for $j>i$.

Let p denote the probability that there is a j-active edge sequence for some $j\geq i$. By a union bound, $p\leq\sum_{j\geq i}p_j$. Using the upper bound on p_j derived in the preceding paragraph, we find that p is upper bounded by an infinite geometric sum with initial term $n^2\exp(-i\delta_{\text{in}}/(64n))$ and ratio $\exp(-\delta_{\text{in}}/(64n))$. Thus

$$\begin{aligned}p &= O((n^3/\delta_{\text{in}})\exp(-i\delta_{\text{in}}/(64n)))\\ &= O(n^3\exp(-c\max(\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}},\log n)))\\ &= O(n^{3-c}).\end{aligned}$$

By setting c to a sufficiently large positive constant, we can drive p below any desired inverse polynomial threshold. The claim of the lemma follows.

Lemma 24. *If r is at least $\min(\lceil 2e^2\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}\rceil,n)$, then the cover time of process R-RANK is, with high probability, $O(n\max(\Delta_{\text{out}}\Delta_{\text{in}}/\delta_{\text{in}}^2,(\log n)/\delta_{\text{in}}))$. The same asymptotic bound holds for the expected cover time.*

PROOF. The high probability claim is immediate from Lemmas 15 and 23. The bound on the expected cover time then follows by Lemma 14.

Theorem 25. *If both Δ_{in} and Δ_{out} are $O(\delta_{\text{in}})$, then there is an r in $O(\delta_{\text{in}})$ such that the cover time of process R-RANK is $O(n + \frac{n \log n}{\delta_{\text{in}}})$ with high probability. The same asymptotic bound holds for the expected cover time.*

Proof. Immediate from Lemma 24.

The result of Theorem 25 matches the lower bound proved by Alon for process MIN and is thus optimal [3].

Note that as r tends to infinity, the behavior of process R-RANK converges to that of process P-RANK. Thus, the bounds of Theorem 25 also hold for process P-RANK.

5. Process R-RANK′

In this section we analyze a biased version of process R-RANK, which we call process R-RANK′. The main result of this section is Theorem 27. Process R-RANK′ is similar to process R-RANK, except that immediately after a selection, if the selected node is uncovered we cover it and move to the next selection. Otherwise, we proceed as in process R-RANK.

In our analysis, we find it helpful to consider another process, which we call process H. Process H runs in two phases. For the first phase, consisting of the first $cn \max(1, (\log n)/\delta_{\text{in}})$ rounds, we run process SELECT. At the end of phase 1, we remove from the graph all edges which did not have at least one end-point selected during phase 1. After the edge removal, we proceed to phase 2 where we begin to cover vertices as in process R-RANK.

Lemma 26. *If process H and process R-RANK′ use the same random rank assignment, infinite series of selections, and tie-breaking node order, the cover time of process R-RANK′ is at most the cover time of process H.*

Proof. We prove the stronger claim that if process H and process R-RANK′ use the same random rank assignment, infinite series of selections, and tie-breaking node order, then at the end of any round i, all nodes covered in process H are also covered in process R-RANK′.

Call a round i low if $i \le cn \max(1, (\log n)/\delta_{\text{in}})$, and high otherwise. We call a node *marked* if it was selected in some low round.

We proceed by induction on i. For the base case, we consider any low round i. In these rounds, process H covers no nodes, so there is nothing to prove.

Now, assume i is high. Let u be the node selected in round i in both process R-RANK′ and process H. If no node is covered in process H, the claim follows from the induction hypothesis. Now assume node v is covered in process H in round i. If v is covered in process R-RANK′ in some round prior to round i, there is nothing to prove. Thus, assume that v is not covered in process R-RANK′ prior to round i. We now complete the induction step by arguing that v must also be covered in process R-RANK′ in round i.

If v is marked, then v is covered in process R-RANK′ in a low round since it was selected in a low round. But, v is not covered in process R-RANK′ prior to round i, so v is unmarked. Since process H selects u and covers v in round i, (u, v) is not removed by process H at the end of phase 1. Thus, u and v cannot both be unmarked, so u is marked.

It follows that u is not equal to v and u is already covered in process R-RANK′ as it was selected in a low round. Since u is marked, it has the same set of outgoing neighbors in both processes, i.e., no edge (u, w) is thrown away in process H at the end of the first phase.

Let S (resp., T) be the uncovered outgoing neighbors of u in process R-RANK′ (resp., process H) at the beginning of round i. By the induction hypothesis, S is contained in T. Since both processes use the same random ranks and tie-breaking node order, the neighbor selection procedure gives well defined order of the nighbors of u. Since $S \subseteq T$ and v is the minimum order node in T and belongs to S, v is the minimum order node in S. Thus v also is covered in round i in process R-RANK′.

Theorem 27. *If $r \geq \min(\lceil 2e^2 \Delta_{\text{out}} \Delta_{\text{in}} / \delta_{\text{in}} \rceil, n)$, then the cover time of process R-RANK′ is, with high probability, $O(n \max(\Delta_{\text{out}} \Delta_{\text{in}} / \delta_{\text{in}}^2, (\log n)/\delta_{\text{in}}))$. The same asymptotic bound holds for the expected cover time.*

Proof. We run a copy of process R-RANK′ in parallel with a copy of process H, using the same random ranks, selections, and tie-breaking node order.

We call phase 1 of process H successful if at least $\delta_{\text{in}}/4$ of every node's in-neighbors are selected. If phase 1 is unsuccessful, we over estimate the cover time of process R-RANK′ by the $O(n \log n)$ cover time of coupon collector. If phase 1 is successful, by Lemma 26 we may overestimate the cover time of process R-RANK′ with the cover time bound of process H. To find the cover time bound of process H, we add the number of rounds during phase 1, to the cover time bound of process R-RANK during phase 2. We apply Lemma 24 to phase 2 of process H where the graph has in-degree at least $\delta_{\text{in}}/4$, to get a cover time bound of $O(\max(\Delta_{\text{out}} \Delta_{\text{in}} / \delta_{\text{in}}^2, (\log n)/\delta_{\text{in}}))$ for process H. Since the bound on the cover time of process H is both with high probability and in expectation, if phase 1 is successful with high probability, the same bound holds for process R-RANK′.

All that remains to be shown is the required result is that phase 1 is successful with high probability.

Consider a specific node w. The probability of selecting a node in $\Gamma_{\text{in}}(w)$ on any selection is a Bernoulli random variable with success probability at least δ_{in}/n. The number of selections in $\Gamma_{\text{in}}(w)$ during phase 1 is the sum of $cn \max(1, (\log n)\delta_{\text{in}})$ such independent Bernoulli random variables. Thus, by Lemma 4, the probability of getting less than $(c/2) \max(\delta_{\text{in}}, \log n)$ selections in $\Gamma_{\text{in}}(w)$ during phase 1 is at most $\exp((c/12) \max(\delta_{\text{in}}, \log n))$, which is an arbitrary inverse polynomial by choosing a large enough constant c.

Given that $(c/2) \max(\delta_{\text{in}}, (\log n))$ selections during phase 1 select a vertex in $\Gamma_{\text{in}}(w)$, we apply Lemma 6. To do so, let the variables in the lemma be $n = |\Gamma_{\text{in}}(w)| \geq \delta_{\text{in}}$, and $j = (c/2) \max(\delta_{\text{in}}, (\log n))$ which is also at least δ_{in} if we set $c \geq 2$. Thus, Lemma 6 tell us that the probability less than $\frac{\delta_{\text{in}}}{4}$ distinct nodes of $\Gamma_{\text{in}}(w)$ are selected during phase 1 of process H is at most $\exp(\frac{c}{2} \max(\delta_{\text{in}}, \log n))$, which is an arbitrary inverse polynomial by selecting a large enough constant c. Taking the union bound over all nodes in the graph shows that phase 1 is successful with high probability.

6. Lower Bounds

In this section, we show lower bound results on the cover times of the processes process A-RANK and process A-RANK′ defined in Section 1. The main results of this section are Theorems 28 and 29. These results establish that the method of picking which uncovered neighbor to cover does make a difference to the resulting expected cover time. While the full proofs of the two theorems are rather lengthy, the main ideas are straightforward. We summarize these main ideas in the two proof outlines that follow. The main technical tools employed in the full proofs are Chernoff bounds and Azuma's inequality (see, e.g., [1, 8]). Note that our lower bounds hold even if we restrict attention to the special class of directed graphs where edge (u, v) is present if and only if edge (v, u) is present; below we refer to such graphs as undirected.

Theorem 28. *For all n, there is an n-node undirected graph G in which each node has degree* $\Theta(\log n)$*, and an assignment of ranks* 1 *through n to the nodes of G, such that process A-RANK has cover time* $\Omega(n\sqrt{(\log n)/\log\log n}) = \omega(n)$*.*

Proof sketch: Fix n and construct G as follows. First, partition the n nodes into ℓ *levels*, numbered from 0 to $\ell - 1$, so that the following conditions hold: level 0 contains $n/2$ nodes; successive levels have a geometrically decreasing number of nodes with ratio $\alpha = \sqrt{(\log n)/\log\log n}$; level $\ell - 1$ is the only level with fewer than $\sqrt{n}$ nodes. Thus $\ell = \Theta((\log n)/\log\log n)$. Assign ranks 1 through n to the nodes in such a way that nodes on lower-numbered levels have lower ranks. For each node u at level i, select $\Theta(\log n)$ nodes at random from each of levels i and $i - 1$ (with replacement), and add an edge from u to each selected node. (If node u is at level 0, then only select nodes from level 0.) We let s_i denote the number of nodes on level i and define τ_i to be $\frac{ni}{c\alpha}$, where c is a sufficiently large constant. We call a level *crowded* if more than half of the nodes on that level are covered.

We inductively show that, with high probability, level i is not crowded until τ_i. For the base case, $i = 0$, the claim is trivially true since τ_0 is 0. For induction, we have two subclaims. First, using the inductive hypothesis we can show that, with high probability, only a small constant fraction of level i is covered between rounds 0 and τ_{i-1}. Second, we can show that, with high probability, only a small constant fraction of level i is covered between rounds τ_{i-1} and τ_i. This completes the proof of the inductive claim. The theorem results from setting $i = \ell$ in the inductive claim. Specifically, with high probability, level ℓ is not crowded on round $\tau_\ell = \Omega(n\sqrt{(\log n)/\log\log n})$, which gives the theorem statement. All that remains to be shown are the two subclaims.

First, we show that, with high probability, only a small constant fraction of level i is covered between rounds 0 and τ_{i-1}. We define a *bad set* as a set of nodes from levels i and $i - 1$ with no uncovered neighbor on level $i - 1$ on round τ_{i-1}. Inductively assuming that level $i - 1$ is not crowded until τ_{i-1}, we can use the probabilistic method to show that with high probability no bad set of size greater than s_{i+1} exists. We over estimate the covers on level i from round 0 to τ_{i-1}, by assuming all selections throughout the specified rounds on level $i + 1$ or on a bad set of size s_{i+1} result in covers on level i. Thus, with high probability, the rate of coverage on level i between rounds 0 to τ_{i-1} is $2s_{i+1}/n$. The expected number of nodes covered is then $2\tau_{i-1}s_{i+1}/n = o(s_i)$. Using Chernoff bounds, we can show that, with high probability, only a small constant fraction of level i is covered by round τ_{i-1}.

Second, we show that, with high probability, only a small constant fraction of level i is covered between rounds τ_{i-1} and τ_i. We over estimate the covers on level i from round τ_{i-1} to τ_i by assuming that all selections on levels $i-1$, i, and $i+1$ result in a cover on level i. The rate of coverage is upper bounded by $2s_{i-1}/n$. From the definitions of s_{i-1}, τ_{i-1} and τ_i, we find the expected number of nodes covered on level i from round τ_{i-1} to τ_i is a small constant fraction of s_i. Using Chernoff bounds, and combining the results from this and the previous paragraph, we can show that with high probability, level i does not become crowded until τ_i, completing the inductive claim.

Theorem 29. *For all n, there is an n-node undirected graph G in which each node has degree $\Theta(\log n)$, and an assignment of ranks 1 through n to the nodes of G, such that process A-RANK′ has cover time $\Omega(n \log\log n) = \omega(n)$.*

Proof sketch: The proof of this theorem proceeds in much the same was as the proof for Theorem 28. Again, partition the n nodes into ℓ levels, numbered from 0 to $\ell - 1$ with level 0 containing about $n/2$ nodes. However, this time, let the ratio α of the number of nodes between successive levels be $(\log n)^{1/4}$. We restrict the number of levels ℓ to $\Theta((\log n)^{3/8} \log\log n)$. Again, assign ranks 1 through n to the nodes in such a way that nodes on lower-numbered levels have lower ranks. For each node u at level i, select $\Theta(\log n)$ nodes at random from each of levels i and $i-1$ (with replacement), and add an edge from u to each selected node. (If node u is at level 0, then only select nodes from level 0.) Again, we let s_i denote the number of nodes on level i.

This time, however, we define τ_i to be a more conservative $\frac{ni}{c\alpha^{3/2}}$, where c is a sufficiently large constant. Furthermore, we change the meaning of *crowded* to denote that more than a $1 - 1/\sqrt{\alpha}$ fraction of the nodes on that level are covered. The motivation for these changes is that node covers resulting from the bias towards the selected node quickly cover a significant fraction of the nodes in each level.

Again, we inductively show that, with high probability, level i is not crowded until τ_i. For the base case, $i = 0$, the claim is trivially true since τ_0 is 0. For induction, we have two subclaims. First, using the inductive hypothesis we can show that, with high probability, only a $1 - 2/\sqrt{\alpha}$ fraction of level i is covered between rounds 0 and τ_{i-1}. Second, we can show that, with high probability, only a further $1/\sqrt{\alpha}$ fraction of level i is covered between rounds τ_{i-1} and τ_i. This completes the proof of the inductive claim. The theorem results from setting $i = \ell$ in the inductive claim. Specifically, with high probability, level ℓ is not crowded on round $\tau_\ell = \Omega(n \log\log n)$, which gives the theorem statement. All that remains to be shown are the two subclaims.

First, we show that, with high probability, only a $1 - 2/\sqrt{\alpha}$ fraction of level i is covered between rounds 0 and τ_{i-1}. We define a *bad set* as a set of nodes from levels i and $i-1$ with no uncovered neighbor on level $i-1$ on round τ_{i-1}. Inductively assuming that level $i-1$ is not crowded until τ_{i-1}, we can use the probabilistic method to show that with high probability no bad set of size greater than s_{i+1} exists. To show the desired result, we analyse an over-estimate of the covers on level i from round 0 to τ_{i-1} in to two parts. First, we assume that all selections throughout the specified rounds on level $i+1$ or a bad set of size s_{i+1} result in covers on level i. We show that with high probability, covers of this type cover no more than a $1/\sqrt{\alpha}$ fraction of level i. Second, using Azuma's inequality, we show that the covers due to bias towards the selected node on level i cover no more than a $1 - 3/\sqrt{\alpha}$ fraction of the nodes on level i with high probability.

In the final remaining claim, we show that, with high probability, only an $1/\sqrt{\alpha}$ fraction of level i is covered between rounds τ_{i-1} and τ_i. We over estimate the covers on level i from round τ_{i-1} to τ_i by assuming that all selections on levels $i-1$, i, and $i+1$ result in a cover on level i. The rate of coverage is upper bounded by $2s_{i-1}/n$. From the definitions of s_{i-1}, τ_{i-1} and τ_i, we find the expected number of nodes covered on level i from round τ_{i-1} to τ_i is a $1/\sqrt{\alpha}$ fraction of s_i. Using Chernoff bounds, and combining the results from this and the previous paragraph, we can show that with high probability, level i does not become crowded until τ_i, completing the inductive claim.

7. Concluding Remarks

For completeness, the reader should notice that assigning r a greater value in the proof of Lemma 22 does not alter the result. It is this fact that makes the locally assigned random ranks more general than the random permutation discussed in the introduction. If we let $r = 2^n$, the random ranks will fix a random permutation with high probability.

We also note that process UNI is equivalent to a process where each node selects a uniformly random permutation of the vertices. Then, when a node is selected, we pick the min-rank neighbor based on the selected node's ranks – as opposed to the global ranks in process P-RANK. This once again highlights the similarities between process P-RANK and process UNI.

Furthermore, we note that process CC on a directed graph can be viewed as a process on a family of sets. Let there be s, not necessarily distinct, sets from a universe of n elements. In each round, we select a set uniformly at random and cover an uncovered element from that set. When $s = n$ and each of the n sets is the set of out-neighbors of a particular node from the directed graph, the two processes are equivalent. Our proof techniques can be used to derive results in this set based process.

References

[1] R. Motwani, P. Raghavan, Randomized Algorithms, Cambridge University Press, Cambridge, UK, 1995.

[2] M. Adler, E. Halperin, R. M. Karp, V. V. Vazirani, A stochastic process on the hypercube with applications to peer-to-peer networks, in: Proceedings of the 35th Annual ACM Symposium on Theory of Computing, 2003, pp. 575–584.

[3] N. Alon, Problems and results in extremal combinatorics—II, Discrete Mathematics 308 (2008) 4460–4472.

[4] E. Upfal, Efficient schemes for parallel communication, Journal of of the ACM 31 (1984) 507–517.

[5] A. G. Ranade, How to emulate shared memory, Journal of Computer and System Sciences 42 (1991) 307–326.

[6] F. T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, and Hypercubes, Morgan-Kaufmann, San Mateo, CA, 1991.

[7] D. A. Levin, Y. Peres, E. L. Wilmer, Markov Chains and Mixing Times, American Mathematical Society, 2008.

[8] N. Alon, J. H. Spencer, The Probabilistic Method, Wiley, New York, NY, 1991.

[9] S. Jukna, Extremal Combinatorics, Springer-Verlag, Berlin, 2001.