Consistent Sampling of Churn Under Periodic Non-Stationary Arrivals in Distributed Systems

XIAOMING WANG, Facebook, USA
DI XIAO, Texas A&M University
XIAOYONG LI, Nvidia
DAREN B. H. CLINE and DMITRI LOGUINOV, Texas A&M University

Characterizing user churn has become an important research area of networks and distributed systems, both in theoretical analysis and system design. A realistic churn model, often measured using periodic observation, should replicate two key properties of deployed systems – (1) the arrival process and (2) the lifetime distribution of participating agents. Because users can be sampled only by sending packets to them and eliciting responses, there is an inherent tradeoff between overhead (i.e., bandwidth needed to perform the measurement) and accuracy of obtained results. Furthermore, all observations are censored, i.e., rounded up or down to a multiple of Δ, where Δ is the minimum delay between repeat visits to the same user. Assuming a stationary arrival process, previous work shows that consistent (i.e., asymptotically accurate) estimation of the lifetime distribution is possible; however, the problem remains open for non-stationary cases. Questions include what distributions these methods sample when the assumptions on the arrival process are violated, under what conditions consistency is possible with existing techniques, and what avenues exist for improving their accuracy and overhead. To investigate these issues, we first use random-measure theory to develop a novel churn model that allows rather general non-stationary scenarios and even synchronized joins (e.g., flash crowds). We not only dispose with common assumptions, such as existence of arrival rate and ergodicity, but also show that this model can produce all metrics of interest (e.g., sampled lifetime distributions, bandwidth overhead) using simple expressions. We apply these results to study the accuracy of prior techniques and discover that they are biased unless user lifetimes are exponential or the arrival measure is stationary. To overcome these limitations, we then create a new lifetime-sampling technique that remains asymptotically robust under all periodic arrival measures and provide a methodology for undoing the bias in the sampled arrival rate created by missed users. We demonstrate that the proposed approach exhibits accuracy advantages and 1-2 orders of magnitude less bandwidth consumption compared to the alternatives. We finish by implementing the proposed framework and applying it to experimental data from massive crawls of Gnutella.

CCS Concepts: • **Networks** → **Network performance modeling**;

Additional Key Words and Phrases: Network sampling, stochastic analysis, lifetime estimation

An earlier version of the paper appeared in IEEE P2P 2009.

Authors' addresses: X. Wang, Facebook, Seattle, WA, 98109; email: xmwang@gmail.com; D. Xiao, Texas A&M University, Department of Computer Science and Engineering, College Station, TX, 77843; email: di@cse.tamu.edu; X. Li, Nvidia, Santa Clara, CA, 95050; email: xiaoyong@cse.tamu.edu; D. B. H. Cline and D. Loguinov, Texas A&M University, Department of Statistics, College Station, TX, 77843; emails: dcline@stat.tamu.edu, dmitri@cse.tamu.edu.

2376-3639/2019/12-ART22 $15.00
https://doi.org/10.1145/3368510

ACM Reference format:
Xiaoming Wang, Di Xiao, Xiaoyong Li, Daren B. H. Cline, and Dmitri Loguinov. 2019. Consistent Sampling of Churn Under Periodic Non-Stationary Arrivals in Distributed Systems. *ACM Trans. Model. Perform. Eval. Comput. Syst.* 4, 4, Article 22 (December 2019), 33 pages.
https://doi.org/10.1145/3368510

1 INTRODUCTION

The problem of sampling temporal and topological characteristics of large-scale decentralized networks (such as Gnutella [12] and KaZaA [16]) has received considerable attention [2], [3], [8], [24], [33], [35], [38]. Capturing the dynamics of these systems entails measuring *churn*, which consists of the arrival process and lifetime distribution $F_L(x)$ of its participants. These parameters provide valuable input to throughput models [11], [29], resilience analysis [20], [44], [45], and system design [13], [21], [33]. Besides P2P, other research fields (e.g., social networks, Internet measurement, security, distributed systems) face similar problems. For example, during Internet scanning, it may be beneficial to obtain the distribution of lifetime that individual IPs stay visible offering a particular service (e.g., DNS, HTTP). Each IP is an ON/OFF process, largely dependent on user actions controlling the specific device. The average IP lifetime was estimated in [19], but no systematic algorithms for sampling $F_L(x)$ have been proposed in that context. Another application is determining the lifetime of botnet nodes and C&C (command & control) hosts using active probing. This may be done by scanning the IP space to identify all live hosts running a particular botnet and issuing periodic connection attempts to track node departures, which occur due to computers being shut down, devices moved to another network, or infections cured. Yet another application are search engines, where web crawlers have to maintain up-to-date snapshots of billions of pages across the Internet. Estimating the distribution of page lifetime (i.e., delay between updates) allows scheduling of future visits, which is done by solving various optimization problems that involve bandwidth-staleness tradeoffs [5], [22], [27], [43]. In fact, any system that can be remotely observed over the Internet falls under the umbrella of our framework. We thus keep our discussion general, where terms “user,” “entity,” “participant,” and “host” are applied interchangeably.

For all studied methods, each contact with a user carries unit bandwidth cost, which consists of sending packets to the target host, performing a handshake using a particular protocol, and shutting down the connection. The measurement cannot proceed at infinitely fast rates to prevent crashing the targets and overloading the observation facility. Thus, there exists a lower-bound $\Delta > 0$ on the delay between contacts with each participant. As a result, all time-related samples are *censored*, i.e., rounded up or down to a multiple of Δ. Combining this with the fact that some users are completely missed (i.e., they join and depart between adjacent observation snapshots), estimation of $F_L(x)$ and the arrival process becomes challenging. The first direction for sampling churn is called *direct* [3], [33], [35], where the observer performs periodic crawls of the system to monitor new arrivals, detect departure of existing users, and infer their lifetimes. Unfortunately, direct sampling misses users that join/depart in between consecutive crawls, which leads to underestimation of the arrival rate and bias in the lifetime distribution towards longer-lived users [23], [38]. The second direction is called *indirect* [38], where the system is scanned only once and all discovered entities are monitored until departure. The obtained residual session lengths are then converted into a lifetime distribution using numerical methods. While this approach requires orders of magnitude less bandwidth than direct sampling, it fails to observe any parameters of the arrival process and is proven to be consistent only in stationary networks [38]. For non-stationary cases commonly found on the Internet [14], [33], [34], [37], it thus remains unknown whether unbiased sampling of churn is actually possible and if consistency under more challenging conditions inherently requires higher overhead. We focus on these issues below.

1.1 Overview of Results

All traditional models of churn assume stationarity [17], [20], [25], [28], [36], [38], [44], [45]. In fact, they either directly use a Poisson arrival process of some constant rate or rely on superposition of renewal processes that can be reduced to Poisson through scaling. For some research problems, simplicity of modeling may be more important than capturing the nuances of diurnal human activity; however, our problem requires explicitly dealing with non-stationarity.

While there are many ways to generalize stationary processes, we are interested in the smallest set of abstractions that allow churn to be measurable. To this end, our first contribution is to propose usage of random measures for modeling user lifecycles. Suppose n is the number of users known to the system, out of which only a small fraction are alive at any given time. Informally speaking, we call a sequence of distributed systems *well-behaved* if the fraction of hosts that join in each interval $(a, b]$ converges to a deterministic value as $n \to \infty$. The well-behaved property subsumes all known efforts related to churn and is necessary for the sampling problem to be solvable. Furthermore, this modeling approach not only allows us to relax such common assumptions as ergodicity, existence of arrival rate, and independence between individual users, but also leads to simple closed-form analysis of accuracy and measurement cost.

Equipped with the new arrival model, our second contribution is to examine the existing algorithms in non-stationary conditions. We first model the family of direct-sampling techniques, which are exemplified by *Create-Based Method* (CBM) [32] and its variations [3], [8], [33], [35]. Under non-stationary churn, we discover that the bias in CBM is a complex function of the arrival process, sampling rate, and lifetime distribution $F_L(x)$. Consistency is achievable, but only when the observation rate tends to infinity or $F_L(x)$ is exponential. For indirect methods, where the main representative is *ResIDual-based Estimator* (RIDE) [38], we show that bias cannot be eliminated even with infinite sampling rates; instead, asymptotic accuracy is possible only when the arrival measure is stationary or lifetimes are exponential, neither of which is realistic in practice [14], [33], [34], [37].

Based on this analysis, the issue of designing a low-overhead and robust churn estimator for non-stationary distributed systems remains open; however, unless additional assumptions are made, this is likely an unsolvable problem. Leveraging the fact that the arrival measure of real networks is often periodic, our fourth contribution is to create a novel algorithm we call *Uniform RIDE* (U-RIDE) that samples the system in random points scattered within the observation window. The naive approach would be run RIDE several times and average the result; however, this does not allow reconstruction of user lifetimes. Instead, we derive a different estimator and show that it is consistent under all well-behaved, periodic arrival measures.

Our last contribution is to derive the bandwidth overhead of the three studied methods, compare them in simulations, and apply them to a Gnutella dataset with 408M user lifetimes. We establish that RIDE is highly sensitive to non-stationarity effects and demonstrate existence of lifetime distributions where it produces completely unusable results (i.e., non-monotonic CDF curves). CBM generally exhibits less bias; however, this comes at an increased bandwidth cost compared to RIDE. In contrast, U-RIDE achieves the best of both worlds—it keeps estimation consistent under all conditions and reduces the overhead of CBM by two orders of magnitude. Since the proposed churn-sampling technique is not limited to Gnutella, it is suitable for other periodic non-stationary distributed systems in today's Internet.

This article is organized as follows: Section 2 overviews related work. Section 3 proposes our general churn model and establishes its measure-theoretic properties. Section 4 explains our objectives in measuring churn, while Section 5 studies the accuracy of previous estimation methods. Section 6 proposes our sampling algorithm and derives its model. Section 7 characterizes the

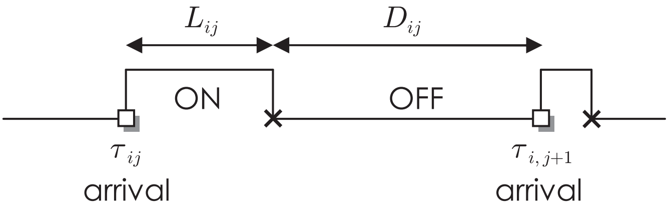


Fig. 1. Process $Z_i(t)$ under stationary ON/OFF user behavior.

overhead of the studied methods, Section 8 performs simulations and Internet experiments, and Section 9 concludes the article.

2 BACKGROUND

2.1 Lifetime Sampling

Early techniques for measuring user lifetimes have emerged from operating-systems literature, where the main issue was to determine the duration for which the various objects existed in the file system. In the *Delete-Based Method* (DBM) [1], lifetime samples were collected upon file deletion by subtracting the creation timestamp from that of each delete request. This process built a distribution of lifetimes conditioned on the corresponding files being deleted within the observation window. In distributed systems, however, this approach is difficult to apply since join timestamps may not be publicly available and departures may not be globally monitored. Thus, a more commonly used technique is the *Create-Based Method* (CBM) [32], which collects lifetime samples only from those users that join and disappear within a given window. To work around the uncertainty in exact arrival/departure times, the sampling process visit the entire system every Δ time units and verifies liveness of each user [3], [8], [33], [35], which creates a step-function approximation to the CDF of user lifetimes.

CBM was analyzed in [38], which found that it suffered from bias related to the missed samples (i.e., users with lifetimes smaller than Δ) and inconsistent round-offs (i.e., some samples rounded up, while others down), which led to potential deviation of the sampled distribution from the true lifetime CDF. To overcome this problem, [38] proposed to measure *remaining* lifetimes of the users seen in a single system-wide crawl, which were then used to recover the lifetime distribution using renewal theory. The paper showed that its proposed method RIDE provided unbiased estimation and exhibited orders of magnitude lower overhead than CBM.

2.2 Churn

Most existing churn models [17], [20], [25], [28], [36], [38], [45] either assume stationary Poisson arrivals from an unspecified number of users or fall under the umbrella of the alternating renewal process of [44], the latter of which we briefly review here. At time t, assume that each user $i = 1, 2, \ldots, n$ is either online (alive) or offline (dead). This behavior can be described by a set of independent alternating renewal processes $\{Z_i(t)\}$, where

$$Z_i(t) = \begin{cases} 1 & \text{user } i \text{ is alive at } t \text{ (ON)} \\ 0 & \text{otherwise (OFF)} \end{cases}. \tag{1}$$

In the illustration of Figure 1, $\{L_{ij}\}_{j=1}^{\infty}$ and $\{D_{ij}\}_{j=1}^{\infty}$ are sequences of independent and identically distributed (iid) ON/OFF durations, respectively. Variables $\{\tau_{ij}\}_{j=1}^{\infty}$ specify arrival times of user i, where $\tau_{i,j+1} = \tau_{ij} + L_{ij} + D_{ij}$. The renewal nature of this model makes each $Z_i(t)$ a stationary point process of constant rate as $t \to \infty$. As a result, superposition of n such arrival processes converges to a stationary point process with constant rate $\lambda(t) = \lambda$. Since this stationarity does not match churn characteristics observed in user-driven systems [14], [33], [34], [37], one requires a more general approach.

Our previous work [39], an earlier version of the current article, offers an initial investigation into building such a framework. That model still equips each user i with an ON/OFF process $Z_i(t)$, but the OFF state now consists of two substates. The first one randomly delays user join within a given day and the other one keeps the user offline until the midnight of the next day following a departure. Note that [39] assumes homogeneous users and requires them to follow a rigid structure of the corresponding $Z_i(t)$. This makes derivations for the general case difficult and imposes certain unnecessary restrictions (e.g., one join of user i per day, all peers follow the same arrival process, existence of arrival rate $\lambda(t)$). On the other hand, the methodology offered below is far more general, i.e., no weaker assumptions can be made. The model of [39] does get used in Section 6.4, but this is a special case needed only for replicating the observed measure in simulations.

3 GENERAL CHURN MODEL

In this section, we cover definitions, present our model of non-stationary arrivals, and derive its main properties.

3.1 Non-Stationary Arrivals

Our first goal is to develop the minimum set of conditions under which both the lifetime distribution and arrival process are measurable. Assume a system of $n \geq 1$ users. For each user i, denote by $\{\tau_{ij}\}_{j=1}^{\infty}$ a monotonically increasing sequence of its arrival times. The corresponding departure times are given by $\{\tau_{ij} + L_{ij}\}_{j=1}^{\infty}$, where $L_{ij} \sim F_L(x)$ is the lifetime of user i during its jth visit into the system, which is independent of all other lifetimes. It should also be noted that certain scenarios where i draws its lifetimes from a separate distribution $F_i(x)$ can be reduced to the homogenous case above by using $F_L(x)$ that is a weighted mixture of individual lifetime CDFs [44].

To aid with the explanation that follows, we next review several definitions from measure theory. Suppose $\mathbb{R}$ is the set of real numbers. Recall that a *measure* m on $\mathbb{R}$ is a function that (a) maps Borel subsets $S \subseteq \mathbb{R}$ to non-negative real numbers; (b) equals zero for the empty set; and (c) satisfies countable additivity [7]. Measures are called *Radon* if they are finite on bounded intervals and *trivial* if they map all sets to zero [30]. If $m(S)$ is a non-degenerate random variable for at least some S, we call the measure *random*; otherwise, *deterministic*. Given a stochastic point process, its random counting measure is the number of events (e.g., arrivals) in each S. In many cases, point processes and their counting measures are used interchangeably [30].

Let $\mathbf{1}_A$ be an indicator of event A and

$$M_i(S) := \sum_{j=1}^{\infty} \mathbf{1}_{\tau_{ij} \in S} \tag{2}$$

be the random arrival measure of user i, i.e., the number of times it joins the system in S. We use $M_i(a, b)$ to represent the measure of interval $(a, b]$. Then, a common assumption for non-stationary point processes is the existence of arrival rate, or intensity, $\lambda_i(t)$ such that for all $a < b$

$$E[M_i(a, b)] = \int_a^b \lambda_i(t) dt. \tag{3}$$

However, there are several drawbacks to using (3). First, it fails to model deterministic (synchronized) arrivals. For example, suppose a flash crowd of k users joins every day at 7 am. This makes both $M_i(0, t)$ and $E[M_i(0, t)]$ a step-function, which precludes existence of $\lambda_i(t)$ in (3) since the left side of the equation cannot be discontinuous. Second, real systems can only be measured in one sample path. Therefore, unless each M_i is ergodic, knowledge of $\lambda_i(t)$ and other expectations computed over multiple realizations provides no useful information. Ergodicity is a difficult condition to verify in practice and an unnecessary constraint in our context. Finally, even if the arrival rate exists for each individual user, consistent lifetime estimation requires $n \to \infty$ and application of the law of large numbers to observed lifetimes. Thus, for the problem to be solvable, additional constraints must be placed not only on the limiting rate

$$\lambda(t) := \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \lambda_i(t), \tag{4}$$

but also the aggregate point process $\sum_{i=1}^{n} M_i$. Since these conditions are no simpler to handle than those introduced below, reliance on rates $\{\lambda_i(t)\}$ does not simplify any of the derivations for our problem and perhaps only obscures the solution.

As a better alternative, we offer a novel way to model churn using random measures. Define the *average arrival measure of the system*

$$m_n := \frac{1}{n} \sum_{i=1}^{n} M_i \tag{5}$$

to count the number of per-user appearances in each interval. We are now interested in systems with sufficiently large n such that m_n approaches a sensible limit. To understand the next definition, note that all integrals in this article are Lebesgue, i.e., taken with respect to some measure. For counting measures, these integrals are summations of the function being integrated over all arrival points in the range of interest, i.e.,

$$\int_S f(t) dM_i(t) = \sum_{j=1}^{\infty} \mathbf{1}_{\tau_{ij} \in S} f(\tau_{ij}), \tag{6}$$

Note that (6) is a random variable for each S.

Definition 1 ([30]). Suppose there exists a Radon measure m on $\mathbb{R}$ such that for all Borel $S \subseteq \mathbb{R}$ and all continuous functions $f(t)$ with compact support on S

$$\int_S f(t) dm_n(t) \to \int_S f(t) dm(t) \tag{7}$$

almost surely (i.e., with probability 1). Then, the sequence of measures $\{m_n\}$ is called *vaguely convergent* to m.

Another way to explain the vague limit is to require that the number of per-user arrivals $m_n(a, b)$ converge to $m(a, b)$ for all a, b in some dense subset of $\mathbb{R}$ [30]. It should also be noted that (7) cannot be replaced with a simpler condition unless a specific type of process (e.g., Poisson) is assumed for each M_i. Besides existence of limiting measure, consistent estimation of $F_L(x)$ requires that m be non-zero and the same in all sample paths. This leads to the following.

Definition 2. As $n \to \infty$, a sequence of systems is called *well-behaved* if m_n converges vaguely to a non-trivial deterministic measure m.

The simplest way to construct a well-behaved network is to use iid point processes, which gives $m = E[M_1]$ from the law of large numbers. If additionally M_1 is stationary, the resulting model is equivalent to those in previous work [17], [20], [25], [28], [36], [38], [45] [44]. However, significantly more interesting cases are possible with our formulation, including all users deterministically synchronized in their arrival and various non-iid cases. Usage of Definition 2 removes the need for stationarity, independence between users, and necessity for multiple sample paths.

One obvious counter-example that fail to produce a well-behaved network requires users to change their arrival rate based on population size, e.g., $M_i(a,b) = 1$ for odd n and $M_i(a,b) = 2$ for even n. In such cases, the limit measure m would not exist and estimation of $F_L(x)$ would be impossible; however, since users are neither aware of n nor particularly sensitive to its value, such scenarios, as well as those even more esoteric, are not of practical interest.

3.2 Convergence of Arrival Rewards

As it turns out, a variety of metrics that depend on the arrival process of the whole system can be easily expressed using m. Specifically, suppose a user arrives at time τ and its lifetime is x. Then, let this arrival carry some random reward $\zeta(\tau,x) \geq 0$, where ζ does not depend on n or measures $\{M_i\}_{i=1}^n$. This allows us to define the per-user arrival reward as a random measure on Borel sets S

$$R_n(\zeta,S) := \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{\infty}\mathbf{1}_{\tau_{ij}\in S}\,\zeta(\tau_{ij},L_{ij}), \tag{8}$$

whose asymptotic behavior we study next.

Theorem 1. *Suppose $L \sim F_L(x)$ is a random lifetime and the reward function ζ is bounded. Furthermore, assume $f(t) = E[\zeta(t,L)]$ has compact support and its set of discontinuity points has zero m-measure. If a sequence of networks is well-behaved, its per-user reward in S converges almost surely to a constant as $n \to \infty$*

$$R(\zeta,S) := \lim_{n\to\infty} R_n(\zeta,S) = \int_S E[\zeta(t,L)]dm(t). \tag{9}$$

Proof. We assume without loss of generality that $m(S) > 0$ and S is a bounded set due to compact support of $f(t) = E[\zeta(t,L)]$. Since $f(t)$ is bounded, has compact support, and its set of discontinuity points has zero m-measure, vague convergence in (7) implies

$$E[R_n(\zeta,S)|\{\tau_{ij}\}] = \int_S E[\zeta(t,L)]dm_n(t) \to R(\zeta,S). \tag{10}$$

Next, observe that

$$nR_n(\zeta,S) = \sum_{i=1}^{n}\sum_{j=1}^{\infty}\mathbf{1}_{\tau_{ij}\in S}\,\zeta(\tau_{ij},L_{ij}) \tag{11}$$

is a sum of $nm_n(S) = \sum_{i=1}^n M_i(S)$ independent bounded random variables, where $nm_n(S) \to \infty$ because m is non-trivial. By the strong law of large numbers for independent random variables with bounded variance [7], it follows that

$$\frac{R_n(\zeta,S) - E[R_n(\zeta,S)|\{\tau_{ij}\}]}{m_n(S)} \to 0 \tag{12}$$

almost surely. From vague convergence, we get that $m_n(S) \to m(S)$ and the limit is finite since m is Radon. Combining this with (10) and (12), we conclude that

$$R_n(\zeta,S) = m_n(S)\frac{R_n(\zeta,S) - E[R_n(\zeta,S)|\{\tau_{ij}\}]}{m_n(S)} + E[R_n(\zeta,S)|\{\tau_{ij}\}] \to R(\zeta,S)$$

almost surely. □

Theorem 1 is the most general result that can be obtained under the circumstances. While it has several technical conditions to ensure mathematical integrity, they may be automatically satisfied in certain applications. For example, continuity of either m or $f(t)$ disposes with the need to verify whether the two share any jump points. While we may not want to relax the general shape of m, rewards ζ used later in this article ensure that $f(t)$ is continuous as long as $F_L(x)$ is. Thus, it is sufficient to assume a continuous lifetime distribution for one of the more tricky conditions in Theorem 1 to disappear. Additionally, since users cannot have unbounded lifetimes, we lose nothing by truncating $F_L(x)$ at some sufficiently large value, which guarantees compact support for $f(t)$ throughout this article. For the same reason, the reward functions used below are always bounded, which means all three conditions of Theorem 1 trivially hold in our problem.

To appreciate the usefulness of the framework introduced in this section and understand the types of rewards we will be using, consider several examples. Suppose t is some observation time and define $S = (-\infty, t]$. One important parameter is the fraction of users alive at t, as a function of $F_L(x)$ and m. Setting $\zeta_1(\tau, x) = \mathbf{1}_{x>t-\tau}$, Theorem 1 yields

$$R(\zeta_1, S) = \int_{-\infty}^{t} \bar{F}_L(t-y)dm(y), \tag{13}$$

where $\bar{F}_L(x) = 1 - F_L(x)$ is the complementary lifetime CDF. This example illustrates a case where an infinite sets S requires a truncated $F_L(x)$ to guarantee compact support in Theorem 1. For finite S, there is no such restriction. Next, suppose we are interested in the combined age of live users at t, normalized by n. This metric can be computed using $\zeta_2(\tau, x) = (t-\tau)\mathbf{1}_{x>t-\tau}$ as

$$R(\zeta_2, S) = \int_{-\infty}^{t} (t-y)\bar{F}_L(t-y)dm(y). \tag{14}$$

Finally, the average age of a live host at time t can be obtained using $R(\zeta_2, S)/R(\zeta_1, S)$. Of course, real networks cannot have an infinite number of participants, which means that our models should be viewed as approximations to actual systems with a sufficient user base. In such cases, scaling the limiting reward by n yields an estimated count of participants that satisfy a given condition. For example, the average population size of a finite network at time t is approximately $nR(\zeta_1, S)$.

Note that for systems that fail to be well-behaved (i.e., Theorem 1 does not hold), it is impossible to not only perform unbiased estimation of $F_L(x)$, but also determine such basic metrics as the expected number of arrivals in S or the overhead needed to crawl these users. Therefore, the well-behaved property is both sufficient and necessary for the main objective of this article to be feasible, i.e., no weaker assumptions can be made.

3.3 Periodic Churn

Many distributed systems in the Internet are driven by diurnal human activity and thus exhibit periodicity in the arrival process. The next two definitions formalize this notion.

Definition 3. An arrival process is *stationary* if for all $\delta > 0$ its measure satisfies

$$\forall t \in \mathbb{R} : m(t, t+\delta) = m(0, \delta), \tag{15}$$

and *non-stationary* otherwise.

Note that stationary processes have linear $m(a, b) = \mu(b-a)$, where $\mu > 0$ is the per-user arrival rate. When $\mu = 1$, this becomes the well-known Lebesgue measure [30].

Definition 4. A non-stationary arrival process is *periodic* if (15) holds for some $\delta > 0$, where the smallest such value is the *period* of the system. Otherwise, the process is *aperiodic*.

For human-driven systems, $\delta = 24$ hours is the most commonly considered period. When it is important to take weekends into account, $\delta = 7$ days is also appropriate.

4 OBJECTIVES

We now explain the goals of this article, the measured parameters, and the conditions under which estimation can be considered successful.

4.1 Lifetimes

Suppose that the target networked system is fully decentralized and that the sampling process has only recurrent access to information about which hosts are currently alive (i.e., continuous observation is impossible). Two sampling activities are possible – discovery of the entire population (e.g., using crawls or scanning) and verification whether one of the previously-seen entities is still online. Due to bandwidth and politeness restrictions on how frequently users can be probed, the sampling process cannot query the system more frequently than one full snapshot per Δ interval, where Δ usually varies from minutes to hours depending on the speed of the measurement facility and network size [2], [3], [8], [10], [14], [15], [19], [24], [31], [33], [34], [35], [37], [38]. Additionally, the sampling process must terminate within some finite window W (e.g., a few days) and operate on a single sample path (i.e., the network cannot be restarted). As a result, all measured lifetimes are discrete (i.e., rounded to a multiple of Δ) and no larger than W. For the problem to be interesting, additionally assume $F_L(x)$ has some mass in $[0, W]$. With this in mind, consider the next definition.

Definition 5. An estimation algorithm Q is *asymptotically* Δ*-consistent* with respect to a target random variable $L \sim F_L(x)$ if it produces a CDF function $F_Q(x)$ that matches the distribution of L in all discrete points $x_j = j\Delta$, for $j = 0, 1, \ldots, W/\Delta$, as sample size scales to infinity.

Note that empirical distributions based on finite averaging will likely deviate from the target distribution $F_L(x)$, which is not a source of bias but rather a limitation of the finite measurement process. Definition 5 instead refers to errors that cannot be eliminated even after obtaining a sufficient number of observations. Additionally, since Δ-sampling cannot produce any observations between points x_j and x_{j+1}, we require that the two CDFs match only in points for which the lifetime samples are available, i.e., $F_Q(x_j) = F_L(x_j)$.

4.2 Arrival Process

Besides estimation of $F_L(x)$, our second goal is to replicate the arrival measure of actual networks using a standalone model with n' point processes $\{\tau'_{ij}\}$. This entails creating a simulated system that is statistically indistinguishable from the real one when sampled by a scanner. In other words, an observer with a given inter-snapshot delay Δ would sample in both cases the same lifetime distribution $F_L(x)$ and see an identical rate of arrival in all intervals $(x_j, x_{j+1}]$ as $n, n' \to \infty$.

It should be noted that repeat visits of the same user into the network may be accompanied by different identities (e.g., IPs, ports) due to NAT, DHCP, and general mobility. As a result, the crawler may not always be able to differentiate new users from those previously seen. This implies that any network with twice as many users, each joining at half the frequency, appears equivalent to the original. Consequently, any n' that is appropriately matched with the corresponding $\{\tau'_{ij}\}$ yields consistent estimation.

5 UNDERSTANDING EXISTING METHODS

Our next step is to characterize the accuracy of current lifetime-estimation techniques under non-stationary arrivals. Note that this section works with generic m and is not limited to periodic behavior.

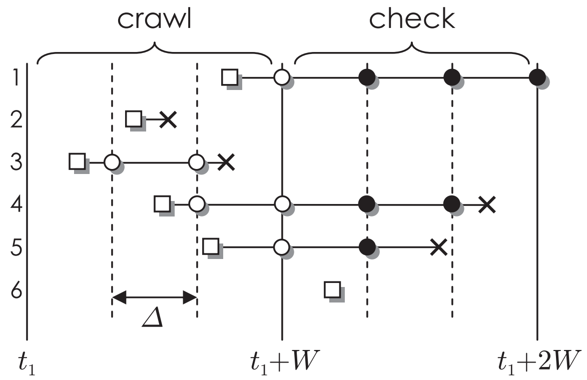


Fig. 2. Illustration of CBM (squares are arrivals, hollow circles are detections during crawls, crosses are departures, and solid circles are liveness checks).

5.1 CBM Estimator

Denote by t_1 the start time of the algorithm and recall from [32] that CBM uses an observation window of size $2W$, which is split into small intervals of size Δ. Within the first half $[t_1, t_1 + W]$, as illustrated in Figure 2, the algorithm takes snapshots of the system at points $t_1 + j\Delta$, i.e., at the beginning of each interval. If a user is observed k times, its lifetime is recorded as $k\Delta$. To avoid sampling bias, Roselli et al. [32] suggests considering only users that appear in the first half of the window $[t_1, t_1 + W]$, disappear before $t_1 + 2W$, and whose recorded lifetimes are no larger than W. Observations that comply with all three rules are called *valid*.

There are two causes of bias in CBM sampling [38]—missed users (i.e., those joining/departing between consecutive crawls) and random direction of round-offs (i.e., some samples rounded up and others down). Define $x_j = j\Delta$ for integer $j \geq 0$ and let a user's lifetime $L \in [x_j, x_{j+1})$ be *inconsistently* sampled if it is rounded down to x_j and *consistently* sampled otherwise (i.e., recorded as x_{j+1}). These concepts are covered by Figure 2 using six different scenarios. The first case produces an invalid sample because user departure does not belong to the window. The second lifetime is missed entirely. The third sample is consistent because its value 1.3Δ is rounded up to 2Δ. The fourth observation is invalid due to its being larger than W. The fifth one is inconsistently sampled because its lifetime 2.7Δ is rounded down to 2Δ. Finally, the sixth user is ignored because it arrives in the second half of the window. As a result, only two valid samples (both 2Δ) are produced.

Let $N_C(x_j, n)$ be number of valid CBM lifetimes no larger than x_j from a measurement that started at time t_1 and $J(n)$ be the total number of user joins observed in $[t_1, t_1 + W]$. Then, the CBM estimator of lifetime distribution $F_L(x)$ is given by

$$F_C(x_j) := \lim_{n\to\infty} \frac{N_C(x_j, n)}{J(n)}, \tag{16}$$

where $j = 0, 1, \ldots, W/\Delta$. Note that the normalization factor $J(n)$ includes *all* users that appear in the first half, not just those with lifetimes smaller than W. Now suppose ρ_j is the fraction of users whose lifetimes are inconsistently rounded-off to x_j, where ρ_0 refers to the probability of missing a user. To make the integrals below more readable, define $a = t_1$, $b = t_1 + W$, and $t^* = (t - t_1) \bmod \Delta$ to be the offset of t from the initial crawl point. The next theorem indicates that the bias in CBM is determined not only by Δ and lifetime distribution $F_L(x)$, but also by the crawl start time t_1, arrival measure m, and window size W.

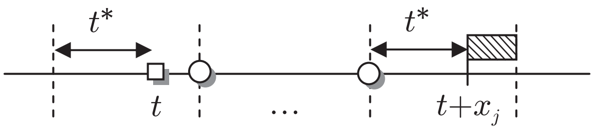


Fig. 3. Inconsistent round-off to x_j in CBM (shaded region).

Theorem 2. *For well-behaved systems, the CBM estimator (16) produces the following distribution as $n \to \infty$*

$$F_C(x_j) = \frac{F_L(x_j) - \rho_0 + \rho_j}{1 - \rho_0}, \tag{17}$$

where for all $j \geq 0$

$$\rho_j = \frac{\int_a^b F_L(x_{j+1} - t^*) dm(t)}{m(a,b)} - F_L(x_j). \tag{18}$$

Proof. Assume a user v joins the system at time $t \in (a,b] = (t_1, t_1 + W]$, i.e., in the first half of the window. To obtain an expression for $N_C(x_j, n)$, notice from Figure 3 that v's lifetime has to satisfy four conditions to be valid and measured no larger than x_j: (1) $L \geq \Delta - t^*$, i.e., seen at least once; (2) $L < x_{j+1} - t^*$, i.e., observed no more than j times; (3) $L \leq t_1 + 2W - t$, i.e., the user departs before the second half is over; and (4) $L \leq W$. The last two condition are automatically satisfied as long as $j \leq W/\Delta$, i.e., the CDF is estimated only up to W. Combining the other two, define the reward variable for each arrival as

$$\zeta_j(t,L) = \begin{cases} 1 & \Delta - t^* \leq L < x_{j+1} - t^* \\ 0 & \text{otherwise} \end{cases}, \tag{19}$$

which using (9) produces

$$\lim_{n\to\infty} \frac{N_C(x_j, n)}{n} = \int_a^b E[\zeta_j(t,L)]dm(t) = \int_a^b \left(\bar{F}_L(\Delta - t^*) - \bar{F}_L(x_{j+1} - t^*)\right) dm(t). \tag{20}$$

As a special case of $x_{j+1} = \infty$, we also get the fraction of users who are seen joining in $(a,b]$ as

$$\lim_{n\to\infty} \frac{J(n)}{n} = \int_a^b \bar{F}_L(\Delta - t^*) dm(t). \tag{21}$$

Dividing (20) by (21) yields the CDF measured by CBM. However, to make it more digestible and easily comparable to previous work [38], we next express it using ρ_j in (18). From Figure 3, observe that v's lifetime is inconsistently rounded off to x_j if and only if its $L \in [x_j, x_{j+1} - t^*)$. Applying (9) again, the fraction of joining users whose lifetimes are inconsistently rounded-off to x_j is

$$\rho_j = \frac{\int_a^b \left(F_L(x_{j+1} - t^*) - F_L(x_j)\right) dm(t)}{m(a,b)}, \tag{22}$$

which is the same as (18). Since (20) simplifies to

$$(F_L(x_j) + \rho_j - \rho_0) m(a,b) \tag{23}$$

and (21) to $(1 - \rho_0) m(a,b)$, their ratio produces (17). □

5.2 CBM Discussion

Note that Theorem 2 generalizes the result developed in [38] to a wider class of networks. Assume W is a multiple of Δ. Since stationary arrivals imply $dm(t) = \mu dt$, (18) becomes

$$\rho_j = \frac{\mu \int_a^b F_L(x_{j+1} - t^*)dt}{\mu W} - F_L(x_j) = \frac{1}{\Delta} \int_{x_j}^{x_{j+1}} F_L(t)dt - F_L(x_j), \tag{24}$$

which together with (17) gives the same expression for the CBM estimator as [38, Theorem 1]. We next investigate whether there exist cases that make CBM unbiased under the new churn model.

Theorem 3. *For well-behaved systems, CBM is asymptotically Δ-consistent for all m iff lifetimes are exponential or $\Delta \to 0$.*

Proof. For asymptotic Δ-consistency, $F_C(x_j)$ in (17) must be equal to $F_L(x_j)$, which is equivalent to requiring that

$$\rho_j = \bar{F}_L(x_j)\rho_0 \tag{25}$$

hold simultaneously for all m and j. The first approach to achieving this is to expand (25) using (18) and obtain

$$\int_a^b [F_L(x_{j+1} - t^*) - F_L(x_j)]dm(t) = \bar{F}_L(x_j) \int_a^b F_L(x_1 - t^*)dm(t). \tag{26}$$

For this to hold for all m and j, one needs to satisfy

$$F_L(x_{j+1} - t^*) - F_L(x_j) = \bar{F}_L(x_j)F_L(x_1 - t^*) \tag{27}$$

for all t. Writing $u = x_j$ and $v = x_1 - t^*$, we have

$$F_L(u + v) - F_L(u) = \bar{F}_L(u)F_L(v), \tag{28}$$

which must be true for all $u, v > 0$. Note that (28) simplifies to the well-known functional equation $\bar{F}_L(u + v) = \bar{F}_L(u)\bar{F}_L(v)$, to which the only non-trivial solution is the exponential family of distributions $\bar{F}_L(x) = e^{-\lambda x}$.

The second method for ensuring (25) is to force $\rho_j \to 0$ for all j. From Taylor expansion, we have

$$F_L(x_{j+1} - t^*) - F_L(x_j) = f_L(x_j)(\Delta - t^*) + \Theta(\Delta^2) = \Theta(\Delta), \tag{29}$$

where $f_L(x)$ is the density of lifetimes. Recalling (22), it follows that $\rho_j = \Theta(\Delta)$, i.e., $\rho_j \to 0$ iff $\Delta \to 0$. □

Interestingly, CBM's conditions for removing bias did not change from those under stationary churn, although its probability of inconsistent round-offs (18) became a more complex function compared to (24). The positive news is that with very small Δ (i.e., seconds), CBM's bias can be negligible in certain situations. However, this requires a substantial bandwidth investment to massively scan the system, which may not only place a non-trivial burden on the measurement facility, but also potentially interfere with normal operation of the network.

In terms of the observed arrival measure in $(a, b]$, CBM samples only users that survive to the nearest crawl boundary, i.e., those with lifetimes larger than $\Delta - t^*$ for arrivals at t. Therefore, its observed arrival measure (i.e., number of seen users) is given by

$$m'(a, a + x_j) \approx n \int_a^{a+x_j} \bar{F}_L(\Delta - t^*)dm(t). \tag{30}$$

Direct usage of m' in place of the unknown nm leads to under-estimation of the true arrival rate. Since CBM does not propose any mechanisms for undoing this bias, we delay a more in-depth

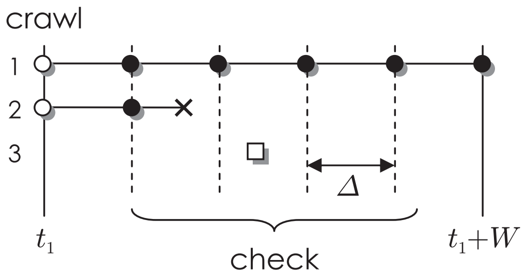


Fig. 4. Illustration of RIDE.

discussion until later. For now, a qualitative assessment is that larger Δ leads to more missed joins and thus a smaller estimated arrival volume than truly present in the network.

5.3 RIDE Estimator

To overcome the bias of CBM in measuring $F_L(x)$, Wang et al. [38] proposed to sampling *remaining* lifetimes of the users seen in a single system-wide crawl, which were then used to recover the lifetime distribution using renewal theory. This article showed that its proposed method RIDE provided unbiased results and exhibited orders of magnitude lower overhead than CBM. While this holds under stationary arrivals, RIDE's performance in more general cases remains unknown. This is our next topic.

At time t_1, this method crawls the network and retains an ϵ-fraction of the discovered users. In each of the following Δ-intervals, the algorithm probes these hosts until they either disappear or W expires. A user seen k times yields a residual lifetime equal to $k\Delta$. In the scenario of Figure 4, the algorithm obtains two samples – 6Δ and 2Δ – while the third user is ignored. Suppose $N_R(x, n)$ is the number of acquired residuals no larger than x. After the observation window is over, the algorithm computes the empirical CDF of residual lifetimes using

$$G_R(x_j) := \lim_{n\to\infty} \frac{N_R(x_j, n)}{N_R(\infty, n)}. \tag{31}$$

Ideally, the method expects $G_R(x_j)$ to equal the residual CDF of the lifetime distribution

$$G(x_j) := \frac{1}{E[L]} \int_0^{x_j} (1 - F_L(t))dt. \tag{32}$$

If this holds, $G(x)$ can yields function $F_L(x)$ using $1 - g(x)/g(0)$, where $g(x) = G'(x)$ is the density of residuals. Therefore, the second step of RIDE is to numerically differentiate (31) to produce an empirical derivative $g_R(x) = G'_R(x)$ and to estimate $F_L(x)$ using

$$F_R(x_j) := 1 - \frac{g_R(x_j)}{g_R(0)}. \tag{33}$$

To quantify the accuracy of (33), we must first determine how its companion function $G_R(x)$ relates to $F_L(x)$.

Theorem 4. *For well-behaved systems, estimator (31) can be written as*

$$G_R(x_j) = 1 - \frac{\int_0^\infty \bar{F}_L(y + x_j)dv(y)}{\int_0^\infty \bar{F}_L(y)dv(y)}. \tag{34}$$

where $\bar{F}_L(y) = 1 - F_L(y)$ and $v(a, b) = m(t_1 - b, t_1 - a)$ is a reverse arrival measure that starts at t_1 and moves to $-\infty$.

Proof. Assume a user joins the system at time $t \le t_1$ and is alive at time t_1. Then, its residual lifetime is sampled as no larger than x_j iff its L falls into $[t_1 - t, t_1 - t + x_j)$. Let the corresponding reward at time t be

$$\zeta_j(t, L) = \begin{cases} 1 & t_1 - t \le L < t_1 - t + x_j \\ 0 & \text{otherwise} \end{cases}. \tag{35}$$

Applying (9), we have

$$\lim_{n\to\infty} \frac{N_R(x_j, n)}{n} = \int_{-\infty}^{t_1} E[\zeta_j(t, L)]dm(t) = \int_{-\infty}^{t_1} (F_L(t_1 - t + x_j) - F_L(t_1 - t))dm(t)$$
$$= \int_0^{\infty} (F_L(y + x_j) - F_L(y))d\nu(y). \tag{36}$$

Using $x_j = \infty$, this also yields

$$\lim_{n\to\infty} \frac{N_R(\infty, n)}{n} = \int_0^{\infty} \bar{F}_L(y)d\nu(y). \tag{37}$$

Dividing (36) by (37), we get (34). □

Differentiating (34) and substituting the result into (33), we get that RIDE estimates $F_L(x)$ via

$$F_R(x_j) = 1 - \frac{\int_0^{\infty} f_L(y + x_j)d\nu(y)}{\int_0^{\infty} f_L(y)d\nu(y)}. \tag{38}$$

5.4 RIDE Discussion

Compared to (32), the result in (38) is generally biased. But things can get worse. As examples later in this article demonstrate, depending on the shape of lifetime density $f_L(x)$ and measure m, (38) may be non-monotonic. As a result, RIDE may produce output that is not only inaccurate, but also completely nonsensical (i.e., not a CDF). Usable information can still be extracted from (38), but it requires pretty strong assumptions, as shown next.

Theorem 5. *For well-behaved systems, RIDE is asymptotically Δ-consistent for all m iff $F_L(x)$ exponential. Furthermore, consistency holds for all $F_L(x)$ iff m is stationary.*

Proof. We start by examining what lifetime distributions can be sampled without bias under all m. To make (38) equal to $F_L(x)$, the following must hold

$$\forall x, t \ge 0 : f_L(t + x) = \bar{F}_L(x) f_L(t). \tag{39}$$

Integrating this over t from y to ∞ produces the same functional equation $\bar{F}_L(y + x) = \bar{F}_L(x)\bar{F}_L(y)$ as in Theorem 3, to which the only non-trivial solution is the exponential tail.

The alternative is to investigate what measures m allow Δ-consistency for all $F_L(x)$. It is not difficult to see that each non-stationary m has a counter-example $f_L(x)$ that makes (38) biased. On the other hand, stationary $m(0, t) = \mu t$, where $\mu > 0$, yields $d\nu(y) = -\mu dy$ and

$$F_R(x_j) = 1 - \frac{\mu \int_0^{\infty} f_L(t + x_j)dt}{\mu} = \int_0^{x_j} f_L(t)dt, \tag{40}$$

which is the same as $F_L(x_j)$. □

Interestingly, interval Δ has no impact on the amount of bias in (38). This means that no matter how fast RIDE samples the system, the error cannot be eliminated. This is in contrast to CBM, which gets more accurate as Δ decreases. Additionally, since RIDE takes only one snapshot, it is unable to estimate the arrival measure m.

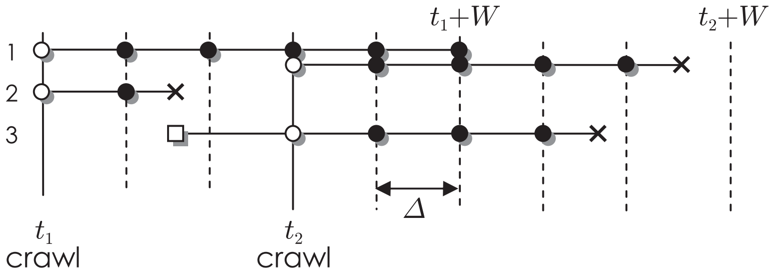


Fig. 5. Illustration of U-RIDE.

5.5 Summary

Our analysis has shown that both existing families of methods (i.e., CBM and RIDE) suffer from bias under general arrival measures m and, to be accurate, require either high overhead (i.e., $\Delta \approx 0$ in CBM) or unrealistic assumptions (i.e., exponential lifetimes, stationary arrivals), neither of which is desirable. They also fail to obtain an accurate estimate of arrival measure m.

6 CONSISTENT SAMPLING OF CHURN

This section studies how to achieve asymptotically accurate measurement of the lifetime distribution and remove the sampling bias in the observed arrival process.

6.1 U-RIDE

Our algorithm, which we call *Uniform RIDE* (U-RIDE), crawls the system at times $t_1, t_2, \ldots, t_M$, where M is the number of snapshots permitted by the overhead-accuracy tradeoff (see below). For each snapshot $k = 1, 2, \ldots, M$, the method identifies all live users and tracks ϵ-fraction of their residuals using recurring connection requests every Δ time units. Each user found in the system at time t_k is probed until $t_k + W$ or until it dies, whichever happens first. The measurement always completes at $t_M + W$. For $\epsilon = 1$ and the example in Figure 5, the crawl at t_1 yields the same residuals 6Δ and 2Δ as in RIDE. The second crawl, which starts at t_2, produces sample 4Δ from user 3. Since user 1 is alive during both crawls, it contributes a fourth residual 5Δ measured in $[t_2, t_2 + W]$. We explain the need for multiple counts in such cases later in this section.

Define the set of snapshot points $\mathcal{T}_M = \{t_1, t_2, \ldots, t_M\}$ to be the *sampling schedule* of U-RIDE. Set $t_k^* = (t_k - t_1) \bmod \delta$ and let $\mathcal{O}_M = \{t_1^*, t_2^*, \ldots, t_M^*\}$ be an *offset schedule*. When $\delta = 24$ hours, the offsets are times within a particular day. From this point on, we assume a periodic measure m with a time-average rate

$$\mu := \frac{1}{\delta} \int_0^\delta dm(t) = \frac{m(0, \delta)}{\delta}. \tag{41}$$

Definition 6. Schedule $\mathcal{T}_M$ is called *asymptotically uniform* if the empirical distribution of its offset schedule $\mathcal{O}_M$ converges to that of a uniform variable in $[0, \delta]$ as $M \to \infty$.

As we show below, uniformity of the schedule is the crux of the new method that allows it to recover target distribution $F_L(x)$ without bias. To see the intuition behind this result, define measure ν_k to be equal to the arrival measure m shifted by t_k, i.e., $\nu_k(a, b) = m(a + t_k, b + t_k)$. Note that the next result holds for time-reversed shifts as well, i.e., $\nu_k(a, b) = m(t_k - b, t_k - a)$.

Theorem 6. *For an asymptotically uniform schedule and periodic m, the aggregate time-shifted measure*

$$\theta_M := \frac{1}{M}\sum_{k=1}^{M} \nu_k \tag{42}$$

converges as $M \to \infty$ to a stationary measure θ with rate μ.

Proof. Observe that for any interval $[a,b]$,

$$\theta(a,b) := \lim_{M\to\infty} \theta_M(a,b) = E[m(a+T, b+T)], \tag{43}$$

where T is uniform in $[0,\delta]$. Then, for any s and periodic m

$$\begin{aligned}\theta(a+s, b+s) &= \frac{1}{\delta}\int_s^{\delta+s} m(a+t, b+t)dt = \frac{1}{\delta}\left(\int_\delta^{\delta+s} m(a+t,b+t)dt - \int_\delta^s m(a+t,b+t)dt\right)\\ &= \frac{1}{\delta}\left(\int_0^s m(a+y+\delta, b+y+\delta)dy - \int_\delta^s m(a+t,b+t)dt\right)\\ &= \frac{1}{\delta}\left(\int_0^s m(a+y, b+y)dy - \int_\delta^s m(a+t,b+t)dt\right)\\ &= \frac{1}{\delta}\int_0^\delta m(a+t,b+t)dt = \theta(a,b). \end{aligned} \tag{44}$$

Recalling (15), we get that θ must be a stationary measure. Its rate μ can be determined by noticing

$$\theta(0,\delta) = E[m(T, \delta+T)] = E[m(0,\delta)] = \mu\delta. \tag{45}$$

The logic for time-reversed measures is very similar. The corresponding manipulations are omitted for brevity. □

For each k, U-RIDE counts the number of users $N_U(x_j, t_k, n)$ seen at t_k with residual lifetimes no larger than x_j. Note that these are the same variables introduced earlier for RIDE, but no longer limited to snapshots at t_1. U-RIDE then uses the following estimator of the residual CDF

$$G_U(x_j) := \lim_{M\to\infty}\lim_{n\to\infty} \frac{\sum_{k=1}^M N_U(x_j, t_k, n)}{\sum_{k=1}^M N_U(\infty, t_k, n)}. \tag{46}$$

We next analyze the relationship between (32) and (46).

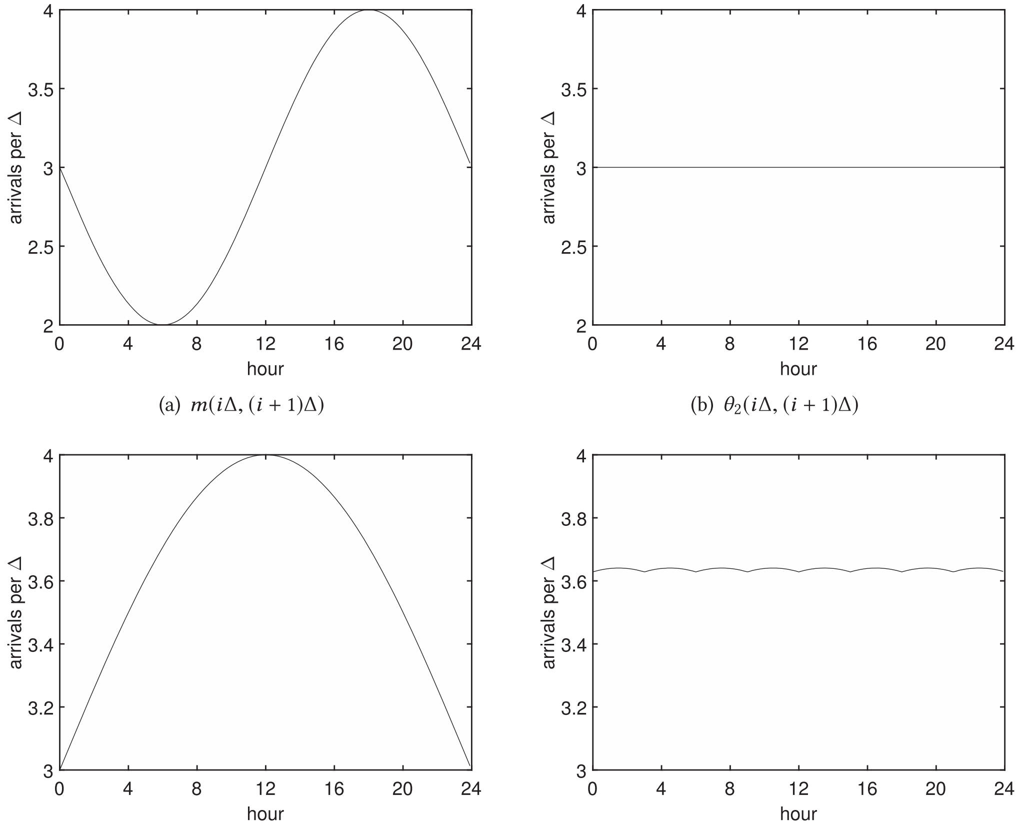
Theorem 7. *Assume a sequence of well-behaved periodic systems. Then, under any asymptotically uniform schedule $\mathcal{T}_M$, the limit in (46) equals the residual CDF of $F_L(x)$ in all points x_j, i.e., $G_U(x_j) = G(x_j)$.*

Proof. Define $\nu_k(a,b) = m(t_k - b, t_k - a)$ to be a time-reversed and shifted measure that corresponds to each point t_k. Recalling (36), we get that

$$\lim_{M\to\infty}\frac{1}{M}\sum_{k=1}^M \lim_{n\to\infty}\frac{N_U(x_j,t_k,n)}{n} = \lim_{M\to\infty}\frac{1}{M}\sum_{k=1}^M \int_0^\infty (\bar{F}_L(y) - \bar{F}_L(x_j+y))d\nu_k(y). \tag{47}$$

Due to uniformity of the schedule, Theorem 6 shows that the aggregate measure (42) converges to a stationary measure θ with rate μ. Therefore, the limit in (47) becomes

$$\int_0^\infty (\bar{F}_L(y) - \bar{F}_L(x_j+y))d\theta(y) = \mu\int_0^{x_j} \bar{F}_L(y)dy. \tag{48}$$



(a) $m(i\Delta, (i+1)\Delta)$

(b) $\theta_2(i\Delta, (i+1)\Delta)$

(c) $m(i\Delta, (i+1)\Delta)$

(d) $\theta_8(i\Delta, (i+1)\Delta)$

Fig. 6. Number of arrivals in every Δ bin (Δ = 6 minutes, $\delta = 24$ hours).

Setting $x_j = \infty$ in (48), we obtain that the denominator of (46), normalized by Mn, converges to $\mu E[L]$. Dividing the two, we get the desired CDF in (32). □

The lifetime estimator for U-RIDE follows the same structure as (33), i.e., uses a k-point numerical derivative $g_U(x) = G'_U(x)$ and

$$F_U(x_j) := 1 - \frac{g_U(x_j)}{g_U(0)}, \tag{49}$$

where values of $k \in [2, 4]$ produce a good tradeoff between complexity, accuracy, and robustness to measurement noise.

6.2 U-RIDE Discussion

Note that $M \to \infty$ is only needed for the most difficult measures m, which are not likely to be encountered in practice. In fact, periodic measures that are *complement-symmetric* during shifts by $\delta/2$ (i.e., $m(0, t) = c - m(\delta/2, \delta/2 + t)$ for some constant c) become stationary with just $M = 2$. One example from the family of sine/cosine functions is shown in Figures 6(a)–6(b). Notice that just one extra snapshot at midpoint $t_2 = \delta/2$ produces an aggregate measure θ_2 that is stationary. A more challenging case is shown in Figures 6(c)–6(d), where we use a measure that is not complement-symmetric and $M = 8$ snapshots. Note that θ_M reduces peak-to-peak fluctuation of m by 81× and approximates stationary conditions rather well. Additionally, even if large M is needed in certain cases, the measurement cost can be offset by reducing ϵ, i.e., retaining a smaller fraction of hosts for monitoring purposes.

Two additional observations are in order. First, the proof of Theorem 7 shows that if a user is alive during multiple snapshots, it must be sampled at each instance and included in the corresponding totals *as if* they were independent users. In Figure 5, the top user survives to t_2, which indicates that U-RIDE must obtain two residuals from it. This can be explained by the fact that integration with respect to ν_k counts these users at each point t_k. Doing otherwise leads to incorrect estimation and bias in the result. Second, one might be tempted to simply apply RIDE at uniformly random time points and then take an average of the resulting CDFs. This would be equivalent to

$$\lim_{M\to\infty} \frac{1}{M} \sum_{k=1}^{M} \lim_{n\to\infty} \frac{N_U(x_j, t_k, n)}{N_U(\infty, t_k, n)} \tag{50}$$

as a replacement for (46). However, this estimator does not converge to anything that allows recovery of $F_L(x)$. As shown in the proof of Theorem 7, the numerator and denominator of (46) must be individually totaled across all snapshots and then divided. This small, yet important, detail has a strong impact on the accuracy.

6.3 Scheduling

The next piece of our algorithm is to find an asymptotically uniform schedule $\mathcal{T}_M$. If δ is known *a priori* and the arrival measure is reasonably smooth, spacing M points equally in $[0, \delta]$ approximates a uniform schedule rather well. For regular human activity, our experiments show that $M \in [2, 8]$ is often enough. However, when δ is unknown, the situation is more interesting. The rest of this subsection focuses on obtaining a provably convergent schedule for such cases.

We use an approach we call *Bernoulli scheduling*, in which

$$t_{k+1} = t_k + v_k\Delta + u_k, \quad k \geq 1, \tag{51}$$

where v_k is drawn from a geometric distribution with success probability p and u_k is drawn from a uniform distribution in $[0, \Delta]$. The former variable controls how often full snapshots are taken and determines duration of the entire measurement, which on average lasts $(M-1)\Delta(1/p + 1/2)$ time units. The latter variable ensures that schedule $\mathcal{T}_M$ is asymptotically uniform. From the property of BASTA (Bernoulli Arrivals See Time Averages) [26], it is straightforward to obtain the following:

Theorem 8. *As $M \to \infty$, Bernoulli scheduling becomes asymptotically uniform for any period δ.*

6.4 Replicating the Arrival Process

We now turn to the issue of building a simulated system of n' users with arrivals in points $\{\tau'_{ij}\}$ that would appear to the sampling process indistinguishable from the original network. Our proposed approach, shown in Figure 7, models user behavior using three states. The first one, which we call WAIT, delays the join time τ'_{ij} by a random offset $A_{ij} \sim F_A(x)$ from the start of the current δ-interval (e.g., midnight if $\delta = 24$ hours). The wait is followed by ON cycles that last for $L_{ij} \sim F_L(x)$ time units, possibly stretching into the next day. After departure, each user stays offline in the REST state for the remainder of the current δ-cycle.

For the simulated process in Figure 7, U-RIDE already has a consistently sampled $F_L(x)$. It thus needs two additional parameters – distribution of wait delays $F_A(x)$ and system size n'. This is our next topic. Assume that m is approximately stationary on small-enough intervals. To take advantage of this, it is beneficial for U-RIDE to minimize the inter-snapshot duration, i.e., space all crawl points equally within the first day. Supposing that δ is known, define $d = t_k - t_{k-1} = \delta/M$

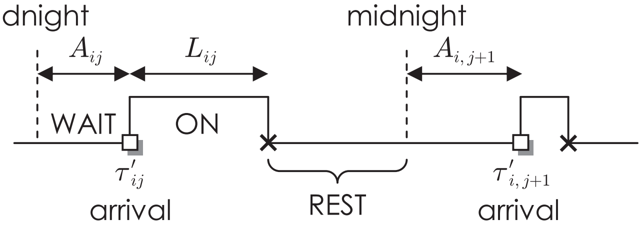


Fig. 7. Simulated user arrival process under periodic churn.

to be the distance between full scans. Then, assuming d is small (i.e., M is sufficiently large), it follows that $dm(t) \approx \mu_k dt$ for $t \in [t_k, t_{k+1}]$, where $\{\mu_k\}$ are some constants. From CBM analysis (30), the sampled arrival measure of U-RIDE is given by

$$m'(t_1, t_k) \approx n \int_{t_1}^{t_k} \bar{F}_L(d - t^*) dm(t) = n \sum_{j=1}^{k-1} \mu_j \int_{t_j}^{t_{j+1}} \bar{F}_L(d - t^*) dt = n \left[\int_0^d \bar{F}_L(t) dt \right] \sum_{j=1}^{k-1} \mu_j. \quad (52)$$

Note that m' is *not* normalized by n, i.e., it counts the actual number of observed hosts. Similar to (24), let the fraction of missed users in each interval be

$$\rho'_0 = \frac{1}{d} \int_0^d F_L(t) dt, \quad (53)$$

which is available to the estimator as long as $F_L(x)$ is. Then, (52) can be written as

$$m'(t_1, t_k) \approx nd(1 - \rho'_0) \sum_{j=1}^{k-1} \mu_j. \quad (54)$$

Noticing that $m(t_1, t_k) \approx d \sum_{j=1}^{k-1} \mu_j$, we get that the observed arrival measure is a scaled version of the unknown m

$$m'(t_1, t_k) \approx n(1 - \rho'_0) m(t_1, t_k). \quad (55)$$

Therefore, U-RIDE can recover $F_A(x)$ directly from the observed arrivals

$$F_A(x) = \frac{m(t_1, t_1 + x)}{m(t_1, t_1 + \delta)} \approx \frac{m'(t_1, t_1 + x)}{m'(t_1, t_1 + \delta)}. \quad (56)$$

As a sanity check, notice that stationary m produces uniform WAIT times in (56), i.e., $F_A(x) = x/\delta$. To determine n', define $Z_{ij} = \tau'_{i,j+1} - \tau_{ij}$ to be the jth inter-arrival delay of user i, where

$$Z_{ij} = \delta \left\lceil \frac{A_{ij} + L_{ij}}{\delta} \right\rceil - A_{ij} + A_{i,j+1}. \quad (57)$$

Now observe that the expected number of joins per period δ can be written as $n'\delta / E[Z_{ij}]$ or $m'(t_1, t_1 + \delta)/(1 - \rho'_0)$, where $m'(t_1, t_1 + \delta)$ is the number of detected arrivals in the first day of observation. Equating the two, we get

$$n' \approx \frac{E[Z_{ij}] m'(t_1, t_1 + \delta)}{\delta(1 - \rho'_0)} = E[D] \frac{m'(t_1, t_1 + \delta)}{1 - \rho'_0}, \quad (58)$$

where $D = \lceil (A + L)/\delta \rceil$, $A \sim F_A(x)$, $L \sim F_L(x)$. Therefore, U-RIDE can replicate churn (i.e., both arrival process and lifetimes) of well-behaved systems using sufficiently large M.

7 OVERHEAD

We now study the connection cost of CBM and compare it to that of U-RIDE.

7.1 CBM

Without loss of generality, we assume that start time $t_1 = 0$ and measurement window W is a multiple of period δ. Let each contact with a host carry unit cost. Then, we have the following result.

Theorem 9. *For well-behaved periodic systems with sufficiently large n and δ/Δ, the overhead of CBM is*

$$B_C \approx \frac{nE[L]}{\Delta}\left(\mu W + \int_0^W [G(W) - G(W-t)]dm(t)\right), \tag{59}$$

where μ is the average per-user arrival rate in (41) and $G(x)$ is the residual CDF from (32).

Proof. In the first half of the window $[0, W]$, CBM crawls the system W/Δ times. Let B_1 be the corresponding cost. Since $N_U(\infty, t, n)$ is the number of live users at time t, each of which must be crawled, we get

$$B_1 = \sum_{j=1}^{W/\Delta} N_U(\infty, x_j, n) \approx n \sum_{j=1}^{W/\Delta} \int_0^\infty \bar{F}_L(y)d\omega_j(y), \tag{60}$$

where measure $\omega_j(a, b) = m(x_j - b, x_j - a)$. If Δ is small compared to period δ and m is smooth, crawl instances x_j can be viewed as satisfying uniform scheduling. In that case, application of Theorem 6 allows further simplification of B_1 to $n\mu E[L]W/\Delta$.

In the second half $[W, 2W]$, CBM tracks users who have joined in the first half. This goes on until they die or their lifetime exceeds W. Let B_2 be the corresponding overhead. Suppose t is the arrival time of a user in the first half. To remain alive at W, its lifetime must satisfy $L > W - t$, while the cost of tracking this user in the second half is given by

$$c(t, L) \approx \frac{t + \min(L, W) - W}{\Delta}, \tag{61}$$

where the approximation arises from omission of rounding to an integer. Note that $c(L, t)$ overestimates the overhead for some users and underestimates for others, depending on the value of t^*. Since Δ is small and t^* of arriving users is roughly uniform in $[0, \Delta]$, the positive and negative errors tend to cancel each other out. Next, defining reward function

$$\zeta(t, L) = \begin{cases} c(t, L) & L > W - t \\ 0 & \text{otherwise} \end{cases}, \tag{62}$$

we get from (9) that

$$B_2 \approx n \int_0^W E[\zeta(t, L)]dm(t). \tag{63}$$

For constants $a, b > 0$ and non-negative L, note that

$$E[\min(L - b, a)\mathbf{1}_{L>b}] = \int_0^a \bar{F}_L(x + b)dx = E[L](G(a + b) - G(b)). \tag{64}$$

Setting $b = W - t$ and $a = t$, this result yields

$$E[\zeta(t, L)] = \frac{E[L]}{\Delta}(G(W) - G(W - t)). \tag{65}$$

Therefore,

$$B_2 = \frac{nE[L]}{\Delta} \int_0^W [G(W) - G(W - t)]dm(t). \tag{66}$$

Combining B_1 and B_2 gives the result in (59). □

In the stationary case, i.e., $m(a, b) = \mu(b - a)$, Theorem 9 reduces to [38, Theorem 8], i.e.,

$$B_C = \frac{nE[L]\mu}{\Delta}\left(W + \int_0^W [G(W) - G(y)]dy\right). \tag{67}$$

7.2 U-RIDE

Recall that U-RIDE crawls the system M times. For each snapshot k, a fraction ϵ of live users is selected and then checked until they disappear or window $[t_k, t_k + W]$ expires. Note that users alive at t_k and t_{k+1} produce two independent samples of residual lifetime, but the network overhead is still that of one user. Therefore, it is beneficial for U-RIDE to cluster its snapshots as close as possible, creating the largest overlap between the live users in adjacent crawls. As a result, the cost-optimal schedule places all M points uniformly within the first day, which also happens to be the best strategy for sampling m in the previous section.

Theorem 10. *Assume U-RIDE with an optimal uniform schedule and a sequence of well-behaved periodic systems. For sufficiently large n and M, the overhead of U-RIDE is*

$$B_U(\epsilon) \approx \frac{nE[L]}{\Delta}\left(\mu\Delta M + \epsilon\sum_{k=1}^{M}\int_{t_{k-1}}^{t_k} h_k(t)dm(t)\right), \tag{68}$$

where $h_k(t) = \bar{G}(t_k - t) - \bar{G}(t_M + W - t)$ *and* $t_0 = -\infty$.

Proof. The first part of the overhead comes from M crawls, which can be computed using the proof of Theorem 9

$$B_3 := \sum_{k=1}^{M} N_U(\infty, t_k, n) \approx nM\mu E[L]. \tag{69}$$

The second portion of the cost comes from tracking each live user. Since all crawl points are in the first day, it follows that W is larger than $t_M = \delta$ and thus each user is probed from its first appearance in the system until $t_M + W$ or until it dies, whichever happens first. Suppose a user arrives at time t such that $t_{k-1} < t \leq t_k$, i.e., k is the first snapshot that detects the user. For convenience, we assume $t_0 = -\infty$. Then, recording $M - k + 1$ residual lifetime samples from this user costs

$$c_k(t, L) = \frac{\min(t + L - t_k, t_M + W - t_k)}{\Delta}, \tag{70}$$

with the corresponding reward

$$\zeta_k(t, L) = \begin{cases} c_k(t, L) & L > t_k - t \\ 0 & \text{otherwise} \end{cases}. \tag{71}$$

This produces the tracking overhead as

$$B_4 \approx \epsilon n \sum_{k=1}^{M}\int_{t_{k-1}}^{t_k} E[\zeta_k(t, L)]dm(t). \tag{72}$$

Setting $b = t_k - t$ and $a = t_M + W - t_k$ in (64), we get

$$E[\zeta(t, L)] = \frac{E[L]}{\Delta}(G(t_M + W - t) - G(t_k - t)) \tag{73}$$

and consequently

$$B_4 \approx \frac{\epsilon nE[L]}{\Delta}\sum_{k=1}^{M}\int_{t_{k-1}}^{t_k}[G(t_M + W - t) - G(t_k - t)]dm(t),$$

which becomes (68) after combining with (69) and expressing the result via the tail distribution $\bar{G}(x)$. □

Two observations are in order. First, $M = 1$ and stationary $dm(t) = \mu dt$ yield the overhead of RIDE in (68), i.e.,

$$B_U(\epsilon) = \frac{nE[L]\mu}{\Delta}\left(\Delta + \epsilon \int_0^W \bar{G}(t)dt\right), \tag{74}$$

which matches [38, Theorem 9]. Second, define $N = 1/M \sum_{k=1}^{M} N_U(\infty, t_k, n)$ to be the average number of observed users per crawl. From (69), we get a variation of Little's Law in application to N – the average number of live entities in the system is the arrival rate μn times the expected delay each user spends online, i.e., $N \approx \mu n E[L]$. Using the observed arrival rate $\mu' = m'(0, \delta)/\delta$, this can be also written as $N \approx \mu' E[L]/(1 - \rho_0')$.

8 SIMULATIONS AND EXPERIMENTS

This section examines the accuracy of derived results and proposed methods in finite graphs (i.e., without $n, M \to \infty$).

8.1 Dataset

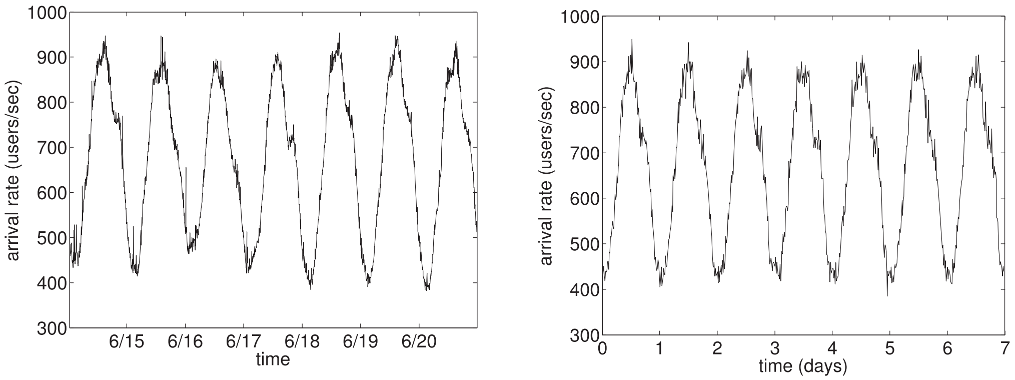
To obtain realistic arrival patterns and lifetimes from human-driven networks, we focus on large-scale P2P systems in this evaluation. One of the biggest such networks amenable to measurement was Gnutella [12], which in 2006-2007 reached a peak of 6.5M concurrent users [38]. While it has significantly shrunk in size after legal action against Limewire and Morpheus, other P2P networks (e.g., BitTorrent) routinely enjoy large populations of users even today; however, they are more difficult to sample at the same scale as Gnutella due to the limitations of the protocol (i.e., no system-wide crawl functionality). Since user behavior and period δ arguably do not change much from year to year, it is perfectly sufficient for our evaluation to engage an older dataset.

Gnutella is an open-protocol P2P file-sharing network that organizes users into a two-tier overlay structure. Users, identified by (IP address, port) pairs, can serve as either ultrapeers or leaves. The former connect to each other, establishing the Gnutella overlay graph, and route search messages to find content. The latter attach to a handful of ultrapeers and do not provide any routing services to other members of the system. Note that Gnutella has no central administration and its global structure at any given time is hidden from the client software. However, leveraging the crawl option of Gnutella/0.6, it is possible to request neighbors of each visited ultrapeer and execute a BFS-like algorithm to capture snapshots of the entire system at different times t_k. Our dataset comes from Gnutella crawls that we performed in June 2007, where a full snapshot was taken every 3 minutes for $W = 7$ days. The average observed join rate was $\mu' = 675$/sec, which gave us a total of 408M instances of peer arrival. There were $U_1 = 50.5$M unique IPs in the dataset and $U_2 = 266$M unique (IP, port) combinations. Since NAT allows multiple users to share one publicly visible IP, it is likely that $n \geq U_1$. Similarly, the same user may appear in the network with different (IP, port) identities, which suggests $n \leq U_2$.

We start by replicating the arrival measure of Gnutella, which is necessary for the simulations below that evaluate the accuracy and bandwidth cost of CBM, RIDE, and U-RIDE.

8.2 Arrival Measure

Setting the delay between snapshots $d = 6$ minutes in U-RIDE (i.e., by discarding half of the crawls in the dataset), we obtain its observed arrival measure m' (or similarly bin-discretized rate) in Figure 8(a). The result clearly indicates periodic churn with $\delta = 24$ hours. For simulation purposes,



(a) Gnutella

(b) simulations

Fig. 8. Observed user arrival rate $m'(id, (i+1)d)/d$.

we use Pareto $F_L(x) = 1 - (1 + x/\beta)^{-\alpha}$ with $\alpha = 3$ and $\beta = 1$ (mean lifetime 0.5 hours). Our analysis in (53) shows that $\rho'_0 = 13\%$ of the users are missed with this choice of $F_L(x)$ and d. Invoking (58), we obtain $n' = 68.5$M users are needed to replicate the observed measure in Figure 8(a). Using the algorithm in Figure 7, we simulate n' non-stationary ON/OFF processes with A_{ij} drawn from the Gnutella $F_A(x)$. We then sample this system every d time units and plot the observed join rates in Figure 8(b). Note that the result is very similar to the one in Figure 8(a). Furthermore, normalization by $(1 - \rho'_0) = 0.87$ during recovery of n' in (58) plays an important role, which none of the previous methods do.

The procedure outlined in this section is quite flexible as it allows usage of our trace to create a simulated Gnutella network for any inter-snapshot delay that is a multiple of 3 minutes. This system can then be sampled using CBM, RIDE, or U-RIDE for the various comparisons below.

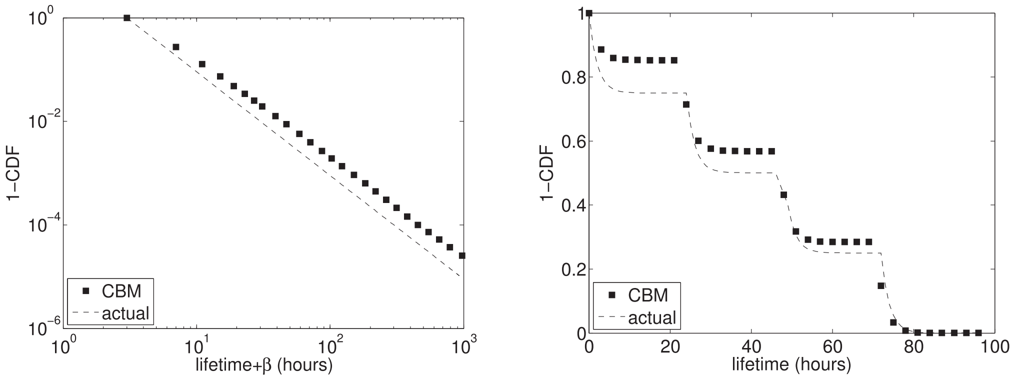
8.3 Accuracy of Lifetime Estimation

Given the replica system from the previous subsection, we consider two types of lifetimes: 1) Pareto with $F_L(x) = 1 - (1 + x/\beta)^{-\alpha}$, where shape α is 2 and scale β is such that $E[L] = 3$ hours; and 2) mixture $L = X_1 + X_2$, where X_1 is uniformly discrete among $\{0, \delta, 2\delta, 3\delta\}$ and $X_2 \in [0, \delta)$ is a truncated exponential random variable with mean 2 hours. The former case models users with heavy-tailed lifetimes, which is fairly standard in evaluating churn models [20], [44]. The latter case covers peers that leave their computers logged in for X_1 full days and then spend a random amount of time X_2 browsing the system before departure.

Using sampling interval $\Delta = 3$ hours, we show the output of CBM in Figure 9. In both cases, it overestimates the tail of the target distribution, which was predicted by (17). RIDE results are plotted in Figure 10, where the Pareto case (a) exhibits a similar amount of bias as in CBM; however, the mixture case (b) produces drastically different results. The estimated distribution is not only inaccurate, but also non-monotonic (i.e., not a CDF). Increasing overhead (i.e., lowering Δ) has no impact and RIDE remains biased regardless of manipulations to the sampling process. The corresponding plots for U-RIDE under Bernoulli scheduling with $p = 0.1$, $\epsilon = 0.01$, and $M = 8$ are provided in Figure 11. Notice that it correctly hits all points $\bar{F}_L(x_j)$ in both distributions.

8.4 Bandwidth Overhead

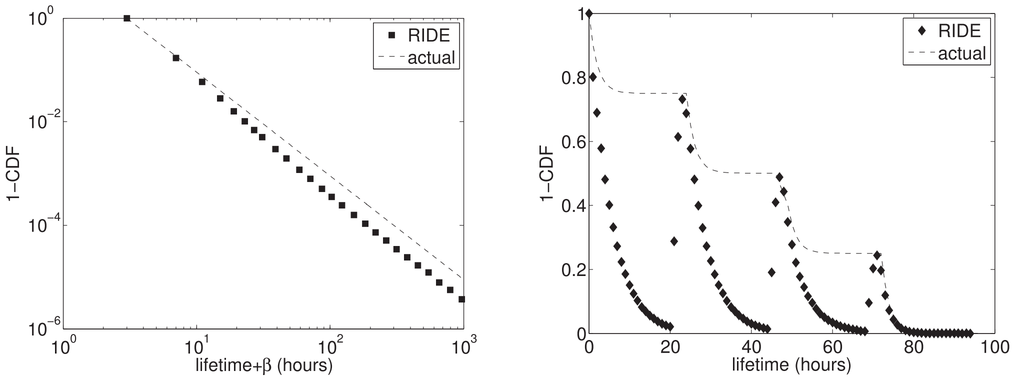
We now compare the sampling cost of CBM and U-RIDE, where the latter uses optimal scheduling (i.e., points $\{t_k\}$ equally spaced in the first day). Simulations use Gnutella's arrival measure,



(a) Pareto ($E[L] = 3$ hrs)

(b) mixture ($E[L] = 38$ hrs)

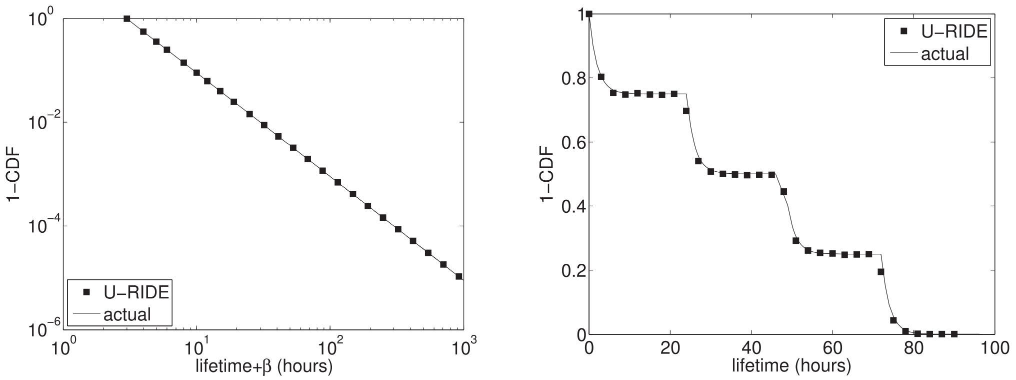
Fig. 9. CBM estimator (16) under Gnutella *m*.



(a) Pareto ($E[L] = 3$ hrs)

(b) mixture ($E[L] = 38$ hrs)

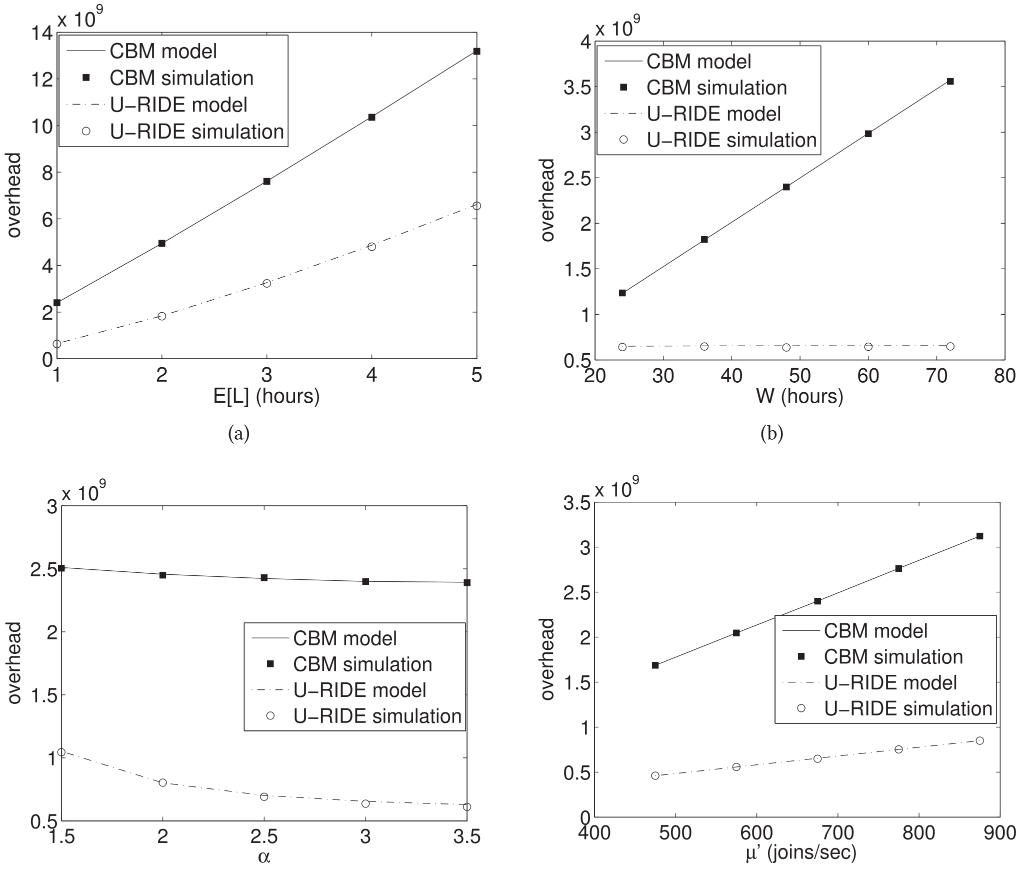
Fig. 10. RIDE estimator (33) under Gnutella *m*.



(a) Pareto ($E[L] = 3$ hrs)

(b) mixture ($E[L] = 38$ hrs)

Fig. 11. U-RIDE estimator (49) with Bernoulli scheduling and Gnutella *m*.



(a)

(b)

(c)

(d)

Fig. 12. Connection overhead under Gnutella m and Pareto lifetimes (default parameters $W = 48$ hours, $\alpha = 3$, $E[L] = 1$ hour, $\mu' = 675$/sec).

$n' = 68.5$M non-stationary processes, Pareto lifetimes, $\epsilon = 1$, and $M = 12$. Figure 12(a) shows that both models (59) and (68) match simulations very well and that the cost is indeed a linear function of $E[L]$. The two methods require between 1B and 13B connections (i.e., $1 - 13$ TB of data using 1 KB per crawl request), depending on the average lifetime. Figure 12(b) confirms that CBM is linear in W, but U-RIDE is insensitive to the observation window length. This is because it probes users until they die, which happens well before the window expires. In Figure 12(c), both methods reduce overhead for lighter-tailed α, which comes from tracking users in proportion to their residual lifetimes (i.e., larger α means users depart quicker). Finally, Figure 12(d) shows that both techniques are linear in observed rate μ'. These conclusions are consistent with (59) and (68).

Once subsampling is taken into account, U-RIDE achieves a more impressive advantage over CBM. Table 1 shows that overhead reduction reaches $1 - 2$ orders of magnitude depending on the parameters of the network. From the proof of Theorem 10, U-RIDE obtains $\epsilon n\mu E[L]M$ residual samples. For $\epsilon = 0.001$ and $M = 8$, this translates into 22K observations, which is more than enough to recover $F_L(x)$ with accuracy similar to that in Figure 11. As n become larger, it is possible to scale $\epsilon \sim 1/n \to 0$, in which case the ratio of CBM cost to that of U-RIDE converges to a simple

Table 1. Overhead under Gnutella Arrivals, Pareto Lifetimes, $E[L] = 1$ hour, $\Delta = 3$ minutes, and $M = 8$

α	W	Overhead ratio of CBM to U-RIDE			
		$\epsilon = 1$	$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$
1.1	48 hrs	1.1	10	59	114
	72 hrs	1.6	12	78	166
	96 hrs	1.7	13	92	216
2	48 hrs	3.7	29	96	124
	72 hrs	5.3	42	140	182
	96 hrs	6.9	55	184	241

Table 2. Measurement Statistics in the Top-10 Subsets by Country and ISP

Country	Samples		Unique IPs	ISP	Samples		Unique IPs
US	120M	48%	21M	FDC Servers	21.5M	8.6%	3.6M
Brazil	36M	14%	6.4M	Level 3	18.2M	7.3%	3.0M
Canada	16M	6.4%	2.6M	Telecom. de Santa Catarina	11.3M	4.5%	2.1M
UK	13M	5.3%	2.0M	Telecom. de Bahia	8.7M	3.5%	1.5M
Germany	6.0M	2.4%	1.0M	SBC Communications	8.2M	3.3%	1.3M
Australia	5.0M	2.0%	0.9M	Verizon	6.2M	2.5%	1.0M
Japan	4.6M	1.9%	0.9M	Telecomunicacoes de Sao Paulo	5.5M	2.2%	1.0M
Netherlands	4.5M	1.8%	0.9M	Shaw	4.8M	1.9%	9.3M
Poland	4.4M	1.7%	0.8M	Cablevision	4.1M	1.6%	0.8M
Austria	4.3M	1.7%	0.7M	Cox	4.0M	1.6%	0.7M

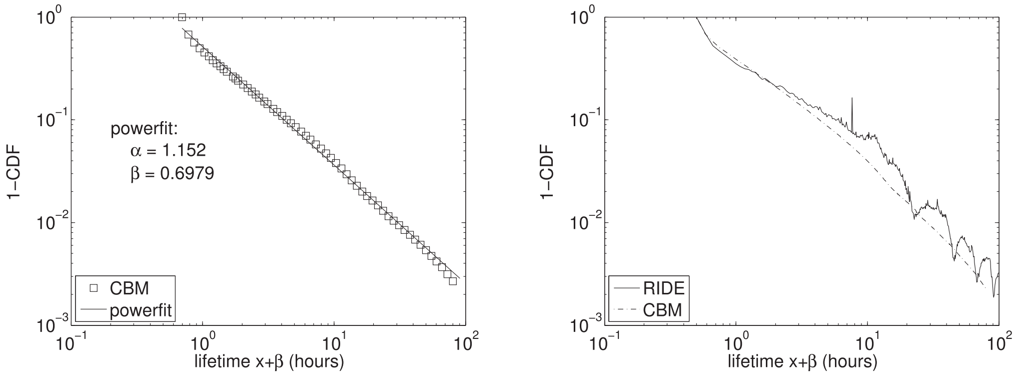
formula $W/(M\Delta) \geq 1$. For a 7-day measurement in Figure 8(a), $\alpha = 2$, and $M = 8$, this means trading 8.8 TB of bandwidth in CBM for just 21 GB in U-RIDE (i.e., a reduction by a factor of 420).

8.5 Gnutella

We now return to the Gnutella dataset and run the studied lifetime estimators over it. We split the observations based on two criteria: geographic location and service provider. Table 2 lists the statistics of top-10 subsets in both categories. While the collected lifetime samples concentrate in just a few countries, with almost 50% in the US, the distribution of users among service providers is more even, with none of the ISPs producing more than 9% of the observations.

To compare accuracy of lifetime estimation, we first need to obtain $F_L(x)$ as ground-truth. While this task is impossible with absolute accuracy, our earlier results (see Theorem 3) have shown that CBM has a diminishing bias as $\Delta \to 0$. In particular, this condition can be considered to hold if $\Delta \ll E[L]$, which is satisfied given Gnutella's $E[L]$ on the order of hours and our $\Delta = 3$ minutes. Figure 13(a) plots the CBM result on a log-log scale along with a power-law fit. The curve indicates that lifetimes follow a power-law distribution with shape $\alpha = 1.15$ and $\beta = 0.69$ (i.e., $E[L] = 4.6$ hours). This value of α is consistent with application of CBM in prior work [3]. In contrast, RIDE in Figure 13(b) produces a non-monotonic function (i.e., an invalid CDF) whose tail is significantly more noisy than in Figure 13(a).

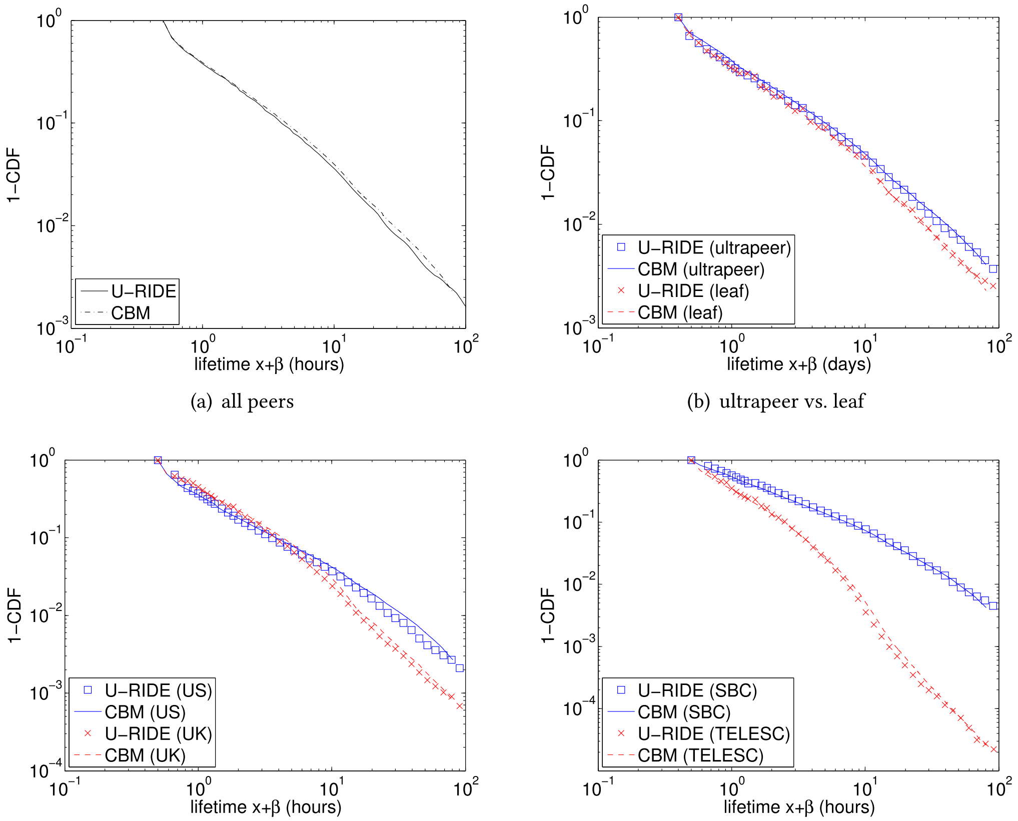
For U-RIDE, we use $p = 1/20$ and collect 24 full snapshots (approximately one for each hour) during the first day. We then apply the corresponding estimator to the original dataset of all peers and plot in Figure 14(a) the curves computed by U-RIDE and CBM. Observe that the two match



(a) CBM with a power fit

(b) RIDE

Fig. 13. Estimated lifetime distribution of all observed peers.



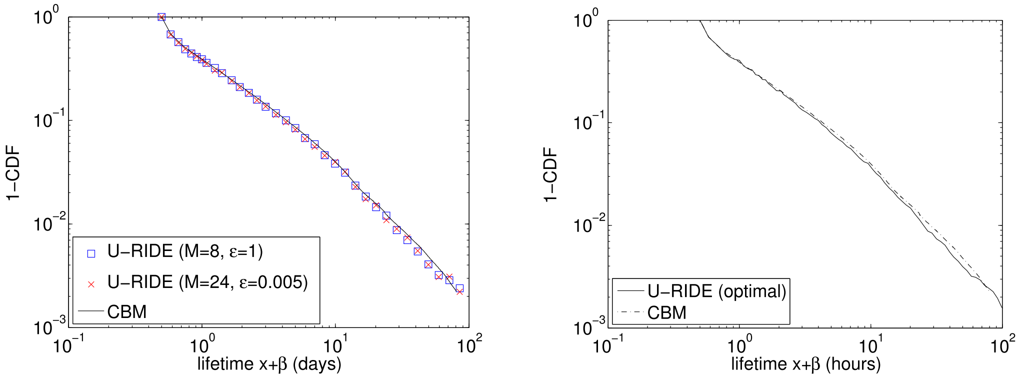
(a) all peers

(b) ultrapeer vs. leaf

(c) geographic location

(d) service provider

Fig. 14. Comparison of U-RIDE with CBM with $M = 24, \epsilon = 1$ in different datasets.



(a) different subsampling rates

(b) $M = 8$, $\epsilon = 0.005$

Fig. 15. Effect of overhead reduction using ϵ.

closely, meaning that the former provides no worse estimation than the latter, *but at dramatically lower cost*. Figure 14(b) shows a similar comparison among ultrapeers and leaves. We next apply U-RIDE to four subsets from Table 2. For geographic location, we use the US and UK to contrast their $F_L(x)$; and for service provider, we select a US carrier SBC Communications and a Brazilian company Telecomunicacoes de Santa Catarina SA (TELESC). Figures 14(c)–14(d) indicate solid agreement between U-RIDE and CBM in all studied zones, as well as show that US peers exhibit heavier tails than those in the UK and especially in Brazil. Additional results based on criteria such as the time zone, protocol version, and software vendor also confirm accuracy of the proposed technique.

Note that the U-RIDE results above did not use subsampling. However, Figure 15(a) shows that other choices of M and ϵ can also produce accurate estimation. In what follows, we explore the parameter space of M and ϵ to strike a balance between accuracy and overhead. To assess accuracy, we employ the *Weighted Mean Relative Difference* (WMRD), which is often used for comparing distribution functions [9], [18], [23]. Given an estimated CDF $F_Q(x)$ and a target distribution $F_L(x)$, the WMRD is defined as

$$w = \frac{\sum_{j=1}^{W/\Delta} |F_Q(x_j) - F_L(x_j)|}{\sum_{j=1}^{W/\Delta} (F_Q(x_j) + F_L(x_j))/2}, \tag{75}$$

where $x_j = j\Delta$. For the evaluation, let r_{CU} be the overhead ratio between CBM and U-RIDE, while r_{CR} be that between CBM and RIDE. To put this in perspective, RIDE exhibits $w = 0.2$ and overhead ratio $r_{CR} = 9.8$ in Figure 13(b), while U-RIDE achieves $w = 0.048$ and $r_{CU} = 4.6$ in Figure 14(a), where both indirect methods use their most inefficient versions with $\epsilon = 1$. Next, we illustrate a more useful scenario.

We run U-RIDE with a set of 72 combinations of parameters M (from 1 to 288) and ϵ (from 0.0001 to 1). To find the optimal choice for M and ϵ, we admit only such pairs that keep $w < 0.1$ and simultaneously $r_{CU} > 100$. Among the five candidates that pass this criteria, we select the pair with the smallest WMRD. The resulting choice is $M = 8$ and $\epsilon = 0.005$, which produces $r_{CU} = 126$ and $w = 0.055$. Figure 15(b) plots the estimated tail using the optimized parameters. Both numerical WMRD score and the figure indicate a very good match, despite the heavy thinning. Since CBM does not admit similar reduction in overhead through subsampling [38, Theorem 7], U-RIDE emerges as a significantly more efficient solution for estimating lifetime distributions in large, non-stationary distributed systems. Furthermore, when inter-crawl duration Δ is large, it is also more accurate.

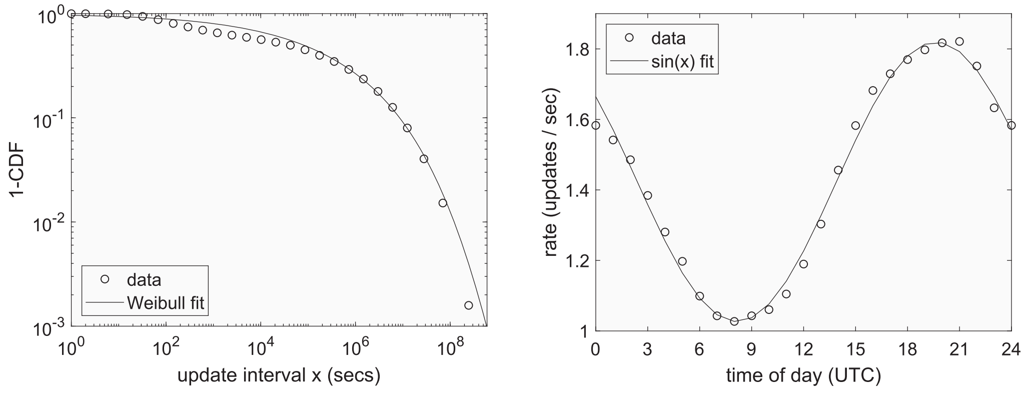
8.6 Wikipedia

Our second example deals with estimation of inter-update delays in web-crawling scenarios [22], [23]. Assume a collection of web pages (i.e., sources) that need to be indexed by a search engine. This procedure typically consists of downloading the pages, parsing their text, constructing a reverse index that maps keywords to pages that contain them, and performing ranking on the resulting database. One of the main challenges in providing useful search results is *data churn*, which refers to random updates (e.g., from webmasters and/or regular users) that modify the original content and make the index of a search engine outdated. Because HTTP is a pull-based protocol, updates cannot be communicated directly to web crawlers, which causes an inherent delay before they are picked up during the next re-crawl. Maintaining a perfectly fresh copy of every page would be ideal; however, this is impossible in practice due to the enormous size of the web and frequent change in content.

A common question in this line of work [4], [6], [22], [23] is to model the relationship between the rate at which pages are revisited and the staleness it produces, where frequent crawls reduce staleness, but also consume large amounts of bandwidth. While this can be formalized using a number of different metrics [22], the most common question is how to compute the crawl rate $\lambda(p)$ that results in a given staleness probability $p > 0$, where p is defined as the fraction of time that the search-engine index deviates from the source. The model for p typically assumes [4], [6], [22] that the crawler visits pages with a known inter-download delay $D \sim F_D(x)$ whose rate $\lambda(p) = 1/E[D]$ needs to be decided. In the computation below, we use exponential $F_D(x)$, which is a common scenario of interest [4], [6], [22], [23].

In order to solve for $\lambda(p)$, the crawler needs to know the inter-update distribution $F_L(x)$, or more specifically its residual $G(x)$ [22]. Unfortunately, neither function is directly available to outside observers. Thus, the only feasible approach is to sample the update process by periodically downloading each page and comparing its content at times t and $t + \Delta$, where as before $\Delta > 0$ is a lower-bound on the return delay to the same page. If a modification is detected in the interval $[t, t + \Delta]$, we count a new update; otherwise, we extend the previous update interval by Δ time units. Note that updates are analogous to lifetimes in our earlier P2P examples. Likewise, CBM can be applied to this problem by crawling each page in the set every Δ time units and directly measuring inter-update delays, some of which are still missed and the others are rounded to a multiple of Δ. RIDE and U-RIDE operate almost the same as before, i.e., by measuring residual delays to the next update from some initial point t or multiple such points $t_1, \ldots, t_M$. Finally, because pages do not go offline, the arrival process here is identical to the update process.

Unlike P2P networks, which are fully decentralized and do not expose timestamps of user arrival, model verification for web-crawling scenarios can be performed using certain publicly shared traces that record all updates and thus provide ground-truth for the various estimation methods. An excellent candidate for this type of analysis is Wikipedia [40], which is one of the most frequently visited sites on the Internet, with 16B page views per month, 36M collaborating editors, and 49M updates per month [42]. Wikipedia page modification is driven by non-stationary activity of humans, which may be argued applies to other web resources as well (e.g., Facebook, Twitter). As a result, our goal is to assess how accurately the three studied methods (i.e., CBM, RIDE, and U-RIDE) can sample the Wikipedia inter-update distribution $F_L(x)$, when used by a web-crawler that does not have access to website internals, and use this information to compute rate $\lambda(p)$. We also examine the associated cost (i.e., number of page downloads).



(a) tail of inter-update distribution $F_L(x)$

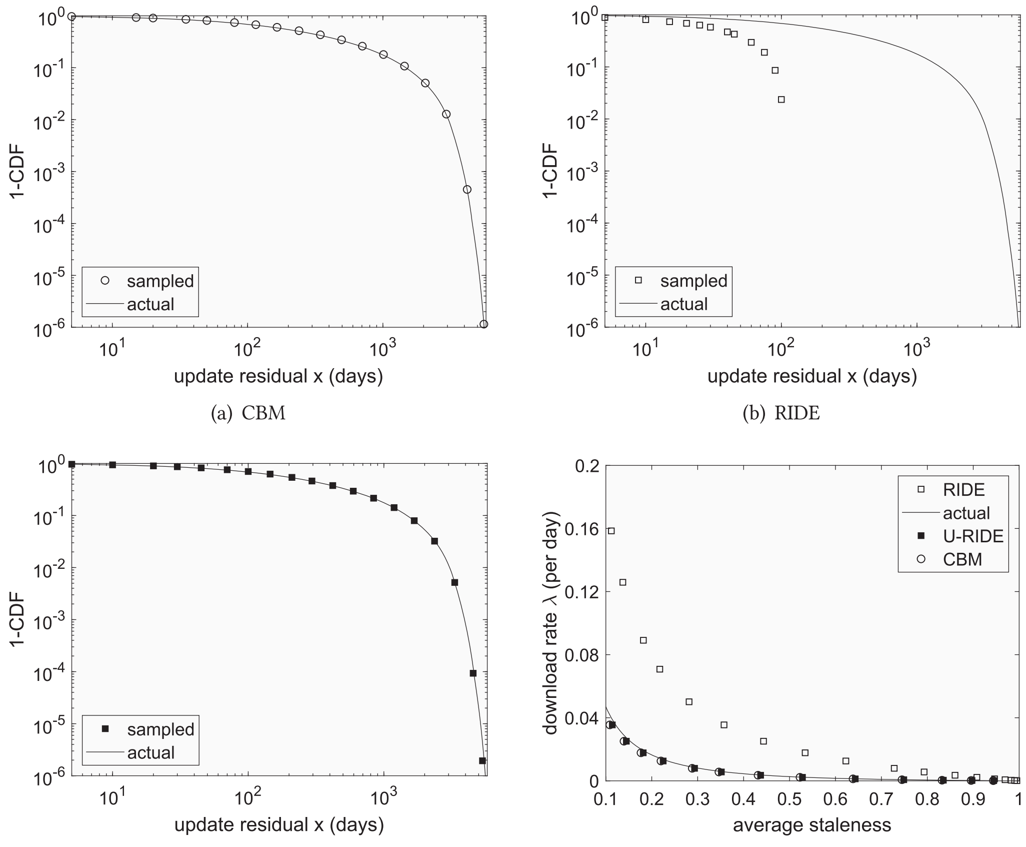
(b) non-stationary update rate

Fig. 16. Properties of Wikipedia.

We focus on the April 2019 dump of English Wikipedia [41], which contains 47M pages and 821M updates over a period of 19 years. The distribution of inter-update delay $F_L(x)$ across all pages is shown in Figure 16(a). The result indicates a good match to the Weibull distribution $1 - e^{(-x/\nu)^k}$ with $\nu = 3.4 \times 10^5$ and $k = 0.6$. The update rate during the day is shown in Figure 16(b), which confirms sine-like periodic behavior, where the highest and lowest rates differ by a factor of 1.8. This is slightly less than in Gnutella's Figure 8(a), where it was 2.1, but significant nevertheless. We next sample the update process of Wikipedia using the three candidate methods. Considering that the average delay between adjacent updates to the same page is $E[L] = 56$ days, we select $\Delta = 5$ days to be small enough to keep CBM accurate. The method crawls each available page in Wikipedia every Δ time units and rounds the observed inter-update delays to the nearest multiple of Δ. Because our target metric of staleness p requires the residual distribution $G(x)$ from (32), we integrate the sampled $F_L(x)$ to get $G(x)$. RIDE executes its residual sampling at one snapshot point, which is selected randomly in December 2018, which happens to be 9:52 am on 12/6/18. It uses subsampling probability $\epsilon = 0.16$, i.e., 16% of the pages are monitored. U-RIDE applies Bernoulli scheduling with $M = 16$ points within the observation window. To keep the number of samples equal to that of RIDE, we set $\epsilon = 0.005$ for U-RIDE. Since both RIDE and U-RIDE directly produce an estimate of $G(x)$, no further conversion is needed to calculate p.

Comparison of the true $G(x)$ against the estimates from CBM, RIDE, and U-RIDE is shown in Figures 17(a)–17(c). While the CBM and U-RIDE curves are indistinguishable from the correct result, RIDE produces a significant deviation. This highlights the pitfalls of using a single snapshot—the residuals at a fixed point t are not generally representative of the whole distribution $G(x)$ when the underlying system is non-stationary. The tail produced by RIDE in Figure 17(b) not only is truncated to 100 days, but also does not resemble $G(x)$ even in this limited range. Using a binary search to deduce $\lambda(p)$ from each estimated $G(x)$, we plot the model-suggested download rate of the crawler for different levels of staleness in Figure 17(d). Compared to the correct value of $\lambda(p)$, there is a significant amount of over-estimation stemming from the RIDE result. For example, it overshoots the correct rate by 4× at $p = 10\%$ and by 7.8× at $p = 50\%$.

While CBM has good estimation accuracy for this choice of Δ, it requires an exorbitant amount of download bandwidth to keep probing each page for the entire window. Our results show that it obtains 311M inter-update delay samples, which requires 8.6B page downloads. At 3.7 KB per page, this is equivalent to 32 TB of network traffic. In contrast, U-RIDE operates with just 515K



(a) CBM

(b) RIDE

(c) U-RIDE

(d) download rate $\lambda(p)$

Fig. 17. Estimation accuracy of the residual distribution $G(x)$ and resulting download rate.

samples, which are collected using 55M downloads (204 GB). The yields a reduction in sampling cost by a factor of 156, but without affecting estimation accuracy.

9 CONCLUSION

The article studied the tradeoff between accuracy and overhead in sampling churn in distributed systems with non-stationary arrivals. We proposed a novel approach for modeling the arrival/departure process of such systems, which was both sufficient and necessary for estimation to be feasible, showed that existing methods were biased under the conditions of this model, and introduced a sampling algorithm that achieved consistent estimation of both the lifetime distribution and the observed arrival measure, while offering a substantial reduction in bandwidth compared to some of the best previous techniques.

REFERENCES

[1] M. G. Baker, J. H. Hartman, M. D. Kupfer, K. W. Shirriff, and J. K. Ousterhout. 1991. Measurements of a distributed file system. In *ACM SOSP*. 198–212.

[2] R. Bhagwan, S. Savage, and G. M. Voelker. 2003. Understanding availability. In *IPTPS*. 256–267.

[3] F. E. Bustamante and Y. Qiao. 2003. Friendships that last: Peer lifespan and its role in P2P protocols. In *Web Content Caching and Distribution*.

[4] Junghoo Cho and Hector Garcia-Molina. 2000. Synchronizing a database to improve freshness. In *ACM SIGMOD*. 117–128.

[5] Junghoo Cho and Hector Garcia-Molina. 2003. Estimating frequency of change. *ACM Trans. Internet Technol.* 3, 3 (Aug. 2003), 256–290. Issue 3.

[6] Junghoo Cho and Alexandros Ntoulas. 2002. Effective change detection using sampling. In *VLDB*. 514–525.

[7] Y. S. Chow and H. Teicher. 1988. *Probability Theory: Independence, Interchangeability, Martingales* (2nd ed.). Springer-Verlag.

[8] J. Chu, K. Labonte, and B. N. Levine. 2002. Availability and locality measurements of peer-to-peer file systems. In *ITCom Conference*, Vol. 4868. 310–321.

[9] N. Duffield, C. Lund, and M. Thorup. 2003. Estimating flow distributions from sampled flow statistics. In *ACM SIGCOMM*. 325–336.

[10] Zakir Durumeric, Eric Wustrow, and J. A. Halderman. 2013. ZMap: Fast internet-wide scanning and its security applications. In *USENIX Security*. 605–620.

[11] Z. Ge, D. R. Figueiredo, S. Jaiswal, J. Kurose, and D. Towsley. 2003. Modeling peer-peer file sharing systems. In *IEEE INFOCOM*. 2188–2198.

[12] Gnutella. [n.d.].

[13] B. Godfrey, S. Shenker, and I. Stoica. 2006. Minimizing churn in distributed systems. In *ACM SIGCOMM*. 147–158.

[14] S. Guha, N. Daswani, and R. Jain. 2006. An experimental study of the skype peer-to-peer VoIP system. In *IPTPS*.

[15] J. Heidemann, Y. Pradkin, R. Govindan, C. Papadopoulos, G. Bartlett, and J. Bannister. 2008. Census and survey of the visible internet. In *ACM IMC*. 169–182.

[16] KaZaA. [n.d.].

[17] S. Krishnamurthy, S. El-Ansary, E. Aurell, and S. Haridi. 2005. A statistical theory of chord under churn. In *IPTPS*. 93–103.

[18] A. Kumar, M. Sung, J. Xu, and J. Wang. 2004. Data streaming algorithms for efficient and accurate estimation of flow size distribution. In *ACM SIGMETRICS*. 177–188.

[19] Derek Leonard and Dmitri Loguinov. 2010. Demystifying service discovery: Implementing an internet-wide scanner. In *ACM IMC*. 109–122.

[20] D. Leonard, V. Rai, and D. Loguinov. 2005. On lifetime-based node failure and stochastic resilience of decentralized peer-to-peer networks. In *ACM SIGMETRICS*. 26–37.

[21] D. Leonard, Z. Yao, X. Wang, and D. Loguinov. 2005. On static and dynamic partitioning behavior of large-scale networks. In *IEEE ICNP*. 345–357.

[22] Xiaoyong Li, Daren B. H. Cline, and Dmitri Loguinov. 2015. On sample-path staleness in lazy data replication. In *IEEE INFOCOM*. 1104–1112.

[23] Xiaoyong Li, Daren B. H. Cline, and Dmitri Loguinov. 2015. Temporal update dynamics under blind sampling. In *IEEE INFOCOM*. 1634–1642.

[24] J. Liang, R. Kumar, and K. W. Ross. 2006. The FastTrack Overlay: A measurement study. *Computer Networks* 50, 6 (Apr. 2006), 842–858.

[25] D. Liben-Nowell, H. Balakrishnan, and D. Karger. 2002. Analysis of the evolution of peer-to-peer networks. In *ACM PODC*. 233–242.

[26] A. Makowski, B. Melamed, and W. Whitt. 1989. On averages seen by arrivals in discrete time. In *IEEE CDC*. 1084–1086.

[27] Christopher Olston and Sandeep Pandey. 2008. Recrawl scheduling based on information longevity. In *WWW*. 437–446.

[28] G. Pandurangan, P. Raghavan, and E. Upfal. 2003. Building low-diameter peer-to-peer networks. *IEEE J. Sel. Areas Commun.* 21, 6 (Aug. 2003), 995–1002.

[29] D. Qiu and R. Srikant. 2004. Modeling and performance analysis of bittorrent-like peer-to-peer networks. In *ACM SIGCOMM*. 367–378.

[30] S. Resnick. 1987. *Extreme Values, Regular Variation, and Point Processes*. Springer-Verlag.

[31] M. Ripeanu, I. Foster, and A. Iamnitchi. 2002. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE Internet Comput. J.* 6, 1 (Jan.-Feb. 2002), 50–57.

[32] D. Roselli, J. R. Lorch, and T. E. Anderson. 2000. A comparison of file system workloads. In *USENIX Annual Technical Conference*. 41–54.

[33] S. Saroiu, P. K. Gummadi, and S. D. Gribble. 2002. A measurement study of peer-to-peer file sharing systems. In *SPIE/ACM Multimedia Computing and Networking*, Vol. 4673. 156–170.

[34] Moritz Steiner, Ernst W. Biersack, and Taoufik Ennajjary. 2007. Actively monitoring peers in kad. In *IPTPS*.

[35] D. Stutzbach and R. Rejaie. 2006. Understanding churn in peer-to-peer networks. In *ACM IMC*. 189–202.

[36] Guang Tan and Stephen Jarvis. May 2007. Stochastic analysis and improvement of the reliability of DHT-based multicast. In *IEEE INFOCOM*. 2198–2206.

[37] Jing Tian and Yafei Dai. 2007. Understanding the dynamic of peer-to-peer systems. In *IPTPS*.

[38] X. Wang, Z. Yao, and D. Loguinov. 2009. Residual-based estimation of peer and link lifetimes in P2P networks. *IEEE/ACM Trans. Networking* 17, 3 (Jun. 2009), 726–739.

[39] X. Wang, Z. Yao, Y. Zhang, and D. Loguinov. 2009. Robust lifetime measurement in large-scale P2P systems with non-stationary arrivals. In *IEEE P2P*. 101–110.

[40] Wikipedia. [n.d.].

[41] Wikipedia Dumps. [n.d.].

[42] Wikipedia Stats. [n.d.].

[43] Mohan Yang, Haixun Wang, Lipyeow Lim, and Min Wang. 2010. Optimizing content freshness of relations extracted from the web using keyword search. In *ACM SIGMOD*. 819–830.

[44] Z. Yao, D. B. H. Cline, X. Wang, and D. Loguinov. 2014. Unifying models of churn and resilience for unstructured P2P graphs. *IEEE Trans. Parallel and Distributed Systems* 25, 9 (Sep. 2014), 2475–2485.

[45] Zhongmei Yao, Xiaoming Wang, Derek Leonard, and Dmitri Loguinov. 2007. On node isolation under churn in unstructured P2P networks with heavy-tailed lifetimes. In *IEEE INFOCOM*. 2126–2134.

Received October 2017; revised October 2019; accepted October 2019