Towards Latency-Optimal Distributed Relay Selection

Yongquan Fu, Yijie Wang, Xiaoqiang Pei
*Science and Technology of Parallel and Distributed Processing Key Laboratory,
School of Computer, National University of Defense Technology, Changsha, Hunan Province

***Abstract*—Latency-sensitive multiparty applications involve intensive communication between multiple participating nodes. Relays are usually adopted for matchmaking end hosts, filtering unwanted traffics, bypassing routing outages and so on. Speeding up the relay-communication becomes increasingly important to improve the QoE of clients. Currently, no rigorous guarantees have been made for the latency-optimal relay communication. We propose a novel framework to truthfully represent the relay communication in the latency space. Real-world data sets show that nearly 90% of node triples obey the average triangle inequality, while our new model allows for the asymmetry and triangle inequality violations to occur. We propose the general triangle to rigorously locate a candidate relay closer to multiple nodes, with which we systematically analyze the feasibility of finding an optimal relay node for arbitrarily sized groups. Our results show that distributed greedy methods are able to locate optimal relays with modest communication overhead and small search hops.**

***Index Terms*—Latency-sensitive applications; relay communication; inframetric model; greedy algorithm; concentric ring; average triangle inequality violation**

I. Introduction

A lot of multiparty communication applications among Internet-wide users have become a typical class of latency-sensitive applications, e.g., Web proxies, Voice-over-IP, multi-player network games. In these applications, participating users have stringent requirements on the Quality of Experiences (QoE) [1], [2], [3]. Users set up communication groups and then communicate with each other within the same group, as a result, each user needs to send/receive timely messages to/from other users in the same group.

To efficiently forward network traffic for the group communication, a popular approach is to set up **service nodes** as **relays** to process and forward real-time messages, which forms a star-like communication topology where the service node is the hub or switch. Further, with the fast penetration of cloud computing and Peer-to-Peer computing, researchers and application developers usually increase the number of nodes that serve as relays for the relay communication, in order to scale well with increasing end hosts.

In large-scale multiparty systems, the relays can be dynamically managed via unstructured overlays to offer better scalability over centralized maintenance. Then for each group-communication request, the relay is dynamically selected over the pool of candidate relays. As many multiparty applications are latency-sensitive, the communication latency between relay nodes and the users dominates users' online experiences. Unfortunately, selecting the relays with minimal network latencies towards end hosts is challenging: First, the overhead is high, as applications need to measure the network latencies between candidate relay nodes and each end host, taking a lot of available bandwidth resources of relays and end hosts. Second, the search takes long periods, since the probing process needs to be completed before selecting relays, as a result, the period for establishing the network connections is prolonged.

Selecting relay nodes using on-demand latency measurements like Meridian [4] bypasses the latency distortions, but is usually trapped at poor local minima because of the triangle inequality violation (TIV) and the clustering in the delay space [5], [6]. While network-coordinate based methods such as PeerWise [7] and IRS [8] select low-latency relays using the coordinate distances, which avoid direct probing costs, but are usually trapped at local minimum due to the coordinate errors.

In this paper, we provide a unifying analytical framework to rigorously study the latency-optimal distributed relay selection problem. Our framework comprises two models:

- The metric space model that assumes the triangle inequality and symmetry to hold. We present the average triangle inequality that bounds the latencies between the relays and the group of targets. This metric holds for nearly 90% of nodes based on the real-world data sets.
- We propose a generalized inframetric model [9] that allows for asymmetric RTTs and accommodates for a network delay space with violations of the average triangle inequality. Since the latency may be asymmetric [10].

We propose a distributed greedy method called RelayGreedy that extends our previous work on selecting nearest nodes for one target [6]. Upon receiving a relay selection request, RelayGreedy selects candidate relay nodes by sampling $log(N)$ (N stands for the number of service nodes) service nodes and recursively forwards the request to a candidate nearer to the targets. The key contribution of this work is on the theoretical analysis of RelayGreedy's performance. We rigorously show that RelayGreedy is able to find the optimal relays for varied numbers of hosts with high probability (w.h.p for short) [1] in both metric space and inframetric space models.

Our optimal relay-selection results can be useful in many popular user-facing cloud services:

[1] The event happens with a probability $1 - N^{-c}$, where N represents the number of service nodes and $c \geq 1$

- **Wide-area Web proxy**: Each Web proxy is a relay that receives clients' http requests, determines the destination's address, and redirects http messages to the destination, and forwards the http responses to the clients. Since Web surfing is highly sensitive to end-to-end latencies, selecting a latency-optimized proxy is important to ensure smooth Web experiences.
- **Voice-Over-IP**: VoIP applications usually adopt the relays to connect the end hosts that are behind NATs or firewalls. For example, Skype maintains a two-level overlay, where the top level consists of stable nodes called supernodes that provide rendezvous-relay service for decentralized hosts, the bottom level involves ordinary hosts that connect to one or several supernodes for the VoIP service.
- **Mutiplayer network game**: Network games typically establish interest groups for clients by matchmaking algorithms [11]. Clients must send game updates fast to all players in the same group. To meet the latency bound of updating game states, DonnyBrook [12] organizes high-bandwidth hosts as relays to forward latency-sensitive messages to group players.
- **Privacy communication**: Since relays decouple the connections between senders and receivers, linking the messages with the sources becomes difficult. For example, SplitX [13] places privacy-analysis proxies for redirecting clients' messages to analysts. To ensure real-time communication, clients send messages to a latency-optimized proxy, and the proxy relays messages to analysts as soon as possible.

In summary, our main contributions include:

- We present a unifying theoretical framework to rigorously analyze the latency-optimized relay selection problem. Our model relaxes the symmetry and triangle inequality, suiting the real-world data sets.
- We present a distributed greedy relay-selection algorithm RelayGreedy and provide rigorous theoretical analysis, we prove that RelayGreedy is able to find approximately optimal relays with modest maintenance overhead and small search hops.

The rest of the paper is organized as follows. Section II formalizes the problem of finding distributed relays. Section III then presents the intuitions and theoretical model to represent the relay-selection problem. Section IV then presents the distributed relay-search method. Then section V provides rigorous theoretical analysis. Section VI shows simulation results. Finally, section VII concludes the paper.

II. Problem Model

We next motivate the problem of selecting relays for distributed networking applications.

A. System Model

Relays passively receive messages for a group of nodes, process the messages, and forward messages to nodes in this group. Relays may be fixed (e.g., Web proxy) or dynamically selected from a set of candidate nodes. The relay then stores necessary contacting addresses of hosts in the same group. Selecting from a pool of dynamic relays scales well with increasing applications or end hosts.

TABLE I
Notation used.

Notation	Meaning
$\mathbf{T}$	Relay-communication clients
L	Number of targets
V	All nodes
S_V	Candidate relays
N	Total nodes
d	Pairwise delays of nodes in V
β	Latency-reduction threshold
ρ	Inframetric parameter

B. Scalable Latency-optimized Relay Selection

To provision real-time message delivery over relays, we need to reduce the latencies of the relay communication. We refer to the nodes for which we need to find the closest relay as the **targets**. The **targets** are those nodes that need to communicate with each other. These targets are usually placed at the edges of the Internet, which have variable delays (tens or hundreds of milliseconds) between each other. The **service nodes** are those relays that can relay messages for a set of decentralized targets. To assure a good quality of experience, selecting a latency optimal relay is important. As a result, the service nodes should be placed in geo-distributed data centers, rather than in one location. Table I lists key notations in the paper.

We state the relay-selection problem as follows: For a service node P and a group $\mathbf{T}$ of L targets, we propose to use the **average delay** $\bar{d}_{P\mathbf{T}}$ from node P to the set of targets $\mathbf{T}$

$$\bar{d}_{P\mathbf{T}} = \frac{1}{L}\sum\nolimits_{j\in\mathbf{T}} d_{Pj} \tag{1}$$

as the proximity metric for quantifying the performance of the relays for the targets $\mathbf{T}$, since the average delay is able to characterize the expected completion time of spreading messages to all targets from the relay.

We next formalize the relay-selection problem, which seeks to find the service node that has the smallest average delay to targets $\mathbf{T}$. A distributed algorithm for selecting the relay is more suitable for large-scale platforms because of the relaxed requirements of collecting the all-pair latencies between the service nodes and the groups of targets. Therefore, we propose the distributed relay-selection problem as follows:

Definition 1: (**Distributed Relay Selection, DRS**) Given a set of targets $S_{\mathbf{T}}$ and a set of service nodes S_C, the problem is to select a node P_* from the set S_C as the relay for the set $S_{\mathbf{T}}$ via distributed algorithms, where the relay P_* has the minimal average RTT to the targets in $S_{\mathbf{T}}$

$$P_* = \underset{P\in S_C}{\arg\min}\, \bar{d}_{PS_{\mathbf{T}}} \tag{2}$$

where $\bar{d}_{PS_{\mathbf{T}}}$ denotes the average RTT value from node P to nodes in set $S_{\mathbf{T}}$.

The targets that require the relay communication are usually dynamically formed. As a result, DRS processes should be online. Further, the online relay selection has to satisfies several requirements: (i) **Accuracy**, to find a service node with the lowest delay in order to increase the quality of experiences of users. (ii) **Scalability**, to incur low bandwidth cost with increasing system size. (iii) **Speed**: to obtain the nearest service node quickly.

C. Related Work

Relay selection: Existing works usually focus on selecting a relay to forward messages for two nodes. In the network layer, the detour routing studies forward real-time network traffic among **two** end nodes in order to decrease the pairwise latency and to improve the route availability. Detour [14] and RON [15] choose the optimal relay nodes using timely probes in order to round off the effects of routing disconnection or congestions. Nakao et al. [16] use the AS topology and the geographical distance information to determine the relays for scalability.

In the application layer, the overlay routing studies seek to minimize the latency of forwarding overlay messages among **two** participating nodes. SOSR [17] chooses a random relay node from a set of online nodes, which does not optimize the end to end delays. Skype maintains an overlay of superpeers that serve as relays for bootstrapping the network connections for end hosts that are behind NATs. Su et al. [18] show that Akamai redirects users' requests mainly based on the network conditions and then propose a one-hop relay policy using the redirection conditions of the Akamai CDN networks. PeerWise [19] and IRS [20] construct routing overlays via a triangle inequality heuristic: each peer selects neighbors as those peers that can mutually reduce the network delays to some Internet nodes.

We generalize the relay-selection problem to multiple participating nodes. Our results provide a unifying framework to select latency-optimized relays for both network and application layers.

Metric and Inframetric Model: There exist two most related works with our study. Meridian [4] proposes the **concentric ring** structure to organize decentralized service nodes as an latency-aware overlay. Meridian proposes a general process to search the nearest node towards one or multiple targets by greedily forwarding among neighbors on the overlays. Meridian provides an interesting theoretical analysis in the metric space for one target, but does not generalize to multiple targets. Further, the metric space analysis does not match the latency metric that may contain asymmetry or TIVs.

HybridNN [6] generalizes Meridian to locate a node that is nearest to one target in a relaxed inframetric model. The inframetric model allows for asymmetry and triangle inequality violations for the pairwise latency, but can not model the average latency from one node to multiple targets. As a result, HybridNN's theoretical results are not suitable for the relay-selection problem.

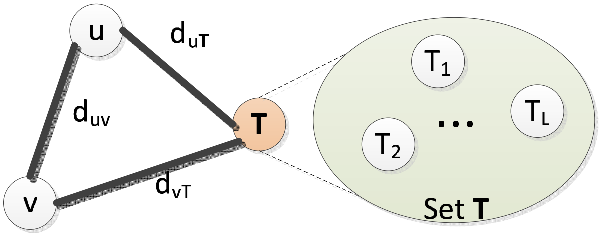


Fig. 1. The generalized triple $u, v, \mathbf{T}$.

III. Relay-selection in Latency Space

We first study the average latency in the metric space model. We next propose a relaxed model to adapt to general settings when the metric-space assumption does not hold.

A. Relay Selection in the Metric Space

Given a set V of nodes, the distance function d is called a metric space if d satisfies the identity of indiscernibles $d(u,u) = 0$, symmetry $d(u,v) = d(v,u)$, and triangle inequality $d(u,v) + d(v,T) \geq d(u,T)$, for $u, v, T \in V$. We propose a metric that helps find the set of service nodes that are close to a group of targets $\mathbf{T} = \{T_1, \ldots, T_L\}$. Prior work has shown that the wide-area latency can be approximately represented in the metric space [4].

1) Average Latency Triangle: We model the average latency in a general triple $(u, v, \mathbf{T})$, where the set $\mathbf{T}$ is regarded as a general vertex. Let the latency value $\left(d_{uv}, \bar{d}_{u\mathbf{T}}, \bar{d}_{v\mathbf{T}}\right)$ be three edges in this triple. Figure 1 illustrates this model.

We define the **average triangle inequality** for each general triple $(u, v, \mathbf{T})$:

$$g_{u,v,\mathbf{T}} = \max\left\{\frac{d_{uv}}{\bar{d}_{u\mathbf{T}} + \bar{d}_{v\mathbf{T}}}, \frac{\bar{d}_{u\mathbf{T}}}{d_{uv} + \bar{d}_{v\mathbf{T}}}, \frac{\bar{d}_{v\mathbf{T}}}{\bar{d}_{u\mathbf{T}} + d_{uv}}\right\} \leq 1 \tag{3}$$

and

$$\min\left\{\frac{d_{uv}}{\left|\bar{d}_{u\mathbf{T}} - \bar{d}_{v\mathbf{T}}\right|}, \frac{\bar{d}_{u\mathbf{T}}}{\left|d_{uv} - \bar{d}_{v\mathbf{T}}\right|}, \frac{\bar{d}_{v\mathbf{T}}}{\left|\bar{d}_{u\mathbf{T}} - d_{uv}\right|}\right\} \geq 1 \tag{4}$$

The maximum and minimum inequality generalizes the triangle inequality in the metric space model. We can trivially see that for metric-space model, the above inequalities both hold for each general triple.

Based on the above inequality equations, we can determine the closer nodes having smaller average latency towards the targets in $\mathbf{T}$. As a result, we can find a latency-optimized relay.

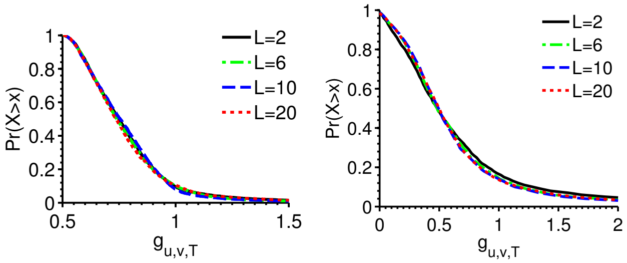
Lemma 3.1: For a service node u, if there exists a node v such that $\bar{d}_{v\mathbf{T}} \leq \beta \bar{d}_{u\mathbf{T}}$, where $\beta \in (0, 1]$, then the delay d_{uv} between u and v must be within the bounds

$$\left[(1-\beta)\,\bar{d}_{u\mathbf{T}}, (1+\beta)\,\bar{d}_{u\mathbf{T}}\right] \tag{5}$$

Proof: By the average triangle inequality of the triple $(u, v, \mathbf{T})$ and $\bar{d}_{v\mathbf{T}} \leq \beta \bar{d}_{u\mathbf{T}}$, we have

$$\begin{aligned} d_{uv} &\leq \bar{d}_{u\mathbf{T}} + \bar{d}_{v\mathbf{T}} \leq (1+\beta)\,\bar{d}_{u\mathbf{T}} \\ d_{uv} &\geq \bar{d}_{u\mathbf{T}} - \bar{d}_{v\mathbf{T}} \geq (1-\beta)\,\bar{d}_{u\mathbf{T}} \end{aligned} \tag{6}$$

which implies that $(1-\beta)\,\bar{d}_{u\mathbf{T}} \leq d_{uv} \leq (1+\beta)\,\bar{d}_{u\mathbf{T}}$. The proof is complete. ■



(a) Mat2500.

(b) Host479.

Fig. 2. The CCDF of the average triangle inequalities for the RTT data sets.

2) Suitability of the Metric-Space Assumption: We use two popular delay data sets to evaluate the relaying performance:

- **Mat2500**, a symmetric RTT matrix between 2500 DNS servers collected by the Meridian project [4] with the King method [21].
  Host479. A RTT delay matrix among Vuze BitTorrent clients [10]. The delay pairs d_{AB} and d_{BA} differ by more than a factor of four in about 40% of the cases [10].

We next study the triangle inequalities for real-world data sets. Figure 2 plots the Complementary Cumulative Distribution Function (CCDF) of the average triangle inequality. We see that for nearly 90% of all triples, the average triangle inequality is not larger than one, which implies that most triples obey the average triangle inequality. While in Host479, there are fewer triples having average triangle inequality below one than those in Mat2500. Since Host479 violates the symmetry of the metric space assumption.

Intuitively, the above procedure immediately leads to a greedy algorithm for recursively locating the relays minimizing the average RTT value to the targets by the objective (2). Unfortunately, if there exists a violation of the average triangle inequality, some local minimum may be found, since the node nearest to the targets may be out of the interval of (5).

B. Relay Selection in Inframetric Space

To relax the triangle inequality for a node triple consisting of three nodes, the inframetric model [9] is defined over each triple (u, v, T), where $d(u, T) \leq \rho \max\{d(u, v), d(v, T)\}$ for $\rho \geq 1$. Unfortunately, the inframetric model does not suit the violations of the average triangle inequality for the average RTTs. We therefore extend the inframetric model to represent **asymmetric** distances and **average-triangle-inequality-violations** among nodes.

For a set S_C of service nodes from the set V. Let $B_u(r)$ be a **closed ball** centered at node u **covering** the set of service nodes whose distances to node u are at most r:

$$B_u(r) = \{v | d(u, v) \leq r, u, v \in S_C\} \tag{7}$$

where r denotes the **radius**.

We next introduce the extended inframetric model:

Definition 2: Let a distance function $d : V \times V \rightarrow \Re^{+}$ denote the pairwise RTT values between nodes in V. Let $\mathbf{T} \subset V$ denote a set of targets. Let a function $\bar{d}$ denote the average RTT from service nodes to the targets in $\mathbf{T}$. The distance function d is called a $(\mathbf{T}, \rho)$-inframetric (where $\rho \geq 1$), *iff* d satisfies the following conditions:
(1) For any pair of nodes u_1 and u_2, where $u_1, u_2 \in V$, $d(u_1, u_2)$=0, then u_1=u_2;
(2) For a triple $(u, v, \mathbf{T})$, where $u, v \in V$ and the targets in $\mathbf{T}$,

$$\begin{array}{l} d_{uv} \leq \rho \min\left\{\max\left\{\bar{d}_{u\mathbf{T}}, \bar{d}_{\mathbf{T}v}\right\}, \max\left\{\bar{d}_{u\mathbf{T}}, \bar{d}_{v\mathbf{T}}\right\}\right\} \\ \bar{d}_{u\mathbf{T}} \leq \rho \min\left\{\max\left\{d_{uv}, \bar{d}_{v\mathbf{T}}\right\}, \max\left\{d_{uv}, \bar{d}_{\mathbf{T}v}\right\}\right\} \\ \bar{d}_{v\mathbf{T}} \leq \rho \min\left\{\max\left\{d_{vu}, \bar{d}_{u\mathbf{T}}\right\}, \max\left\{d_{vu}, \bar{d}_{\mathbf{T}u}\right\}\right\} \end{array} \tag{8}$$

hold.

The inframetric parameter ρ is defined for each generalized triple $(u, v, \mathbf{T})$. Different triples may have varying ρ values by Eq (8). When the RTT values among u, v and $\mathbf{T}$ are asymmetric, the minimum ρ for the triples $(u, v, \mathbf{T})$ and $(v, u, \mathbf{T})$ may also differ.

We next show that the inframetric model is able to bound the set of nodes close to the targets. For a set $\mathbf{T}$ of target nodes, let the set $B_{\mathbf{T}}(y)$ be those nodes whose **average** RTTs to targets in $\mathbf{T}$ are at most y as:

$$B_{\mathbf{T}}(y) = \left\{v \left| \bar{d}_{v\mathbf{T}} \leq y, v \in S_C \right.\right\} \tag{9}$$

Given a candidate relay u, we would like to obtain the set of candidate relays that are closer to the targets $\mathbf{T}$ than node u.

Lemma 3.2: Let $\bar{d}_{u\mathbf{T}} = r$. Assume that there exists a service node v that is at least β ($\beta \leq 1$) times closer to targets $\mathbf{T}$:

$$\bar{d}_{v\mathbf{T}} \leq \beta \bar{d}_{u\mathbf{T}} \tag{10}$$

Then node v must be included in the closed ball $B_u(\rho r)$.

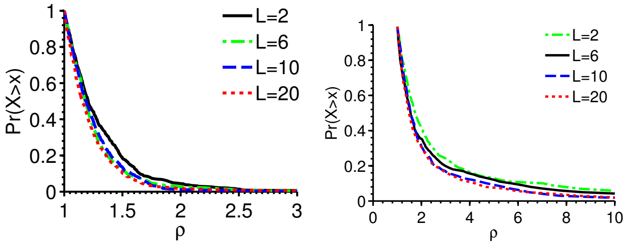
Proof: Since $\bar{d}_{v\mathbf{T}} \leq \beta r$, node v is included by the set $B_{\mathbf{T}}(\beta r) = \left\{x \left| \bar{d}_{x\mathbf{T}} \leq \beta r, x \in S_C \right.\right\}$. By the definition of the inframetric model, Node v is also covered by the ball
$B_u(\rho r) = \{x | d(u, x) \leq \rho r, u, x \in S_C\}$, since

$$\begin{array}{l} d_{uv} \leq \rho \min\left\{\max\left\{\bar{d}_{u\mathbf{T}}, \bar{d}_{\mathbf{T}v}\right\}, \max\left\{\bar{d}_{u\mathbf{T}}, \bar{d}_{v\mathbf{T}}\right\}\right\} \\ \leq \rho \max\left\{\bar{d}_{u\mathbf{T}}, \bar{d}_{v\mathbf{T}}\right\} \\ \leq \rho \bar{d}_{u\mathbf{T}} = \rho r \end{array} \tag{11}$$

As a result, the closed ball $B_u(\rho r)$ contains the node v. ■

We can see that the Lemma 3.2 generalizes the previous Lemma 3.1 that assumes the average triangle inequality to hold. Since $\beta \leq 1$ implies that $(1 + \beta)r \leq 2r$, if we set $\rho \geq 2$, the nodes in the interval $\left[(1 - \beta)\bar{d}_{u\mathbf{T}}, (1 + \beta)\bar{d}_{u\mathbf{T}}\right]$ are all covered by the ball $B_u(\rho r)$. Therefore, in each step, the inframetric model considers more nodes at candidates for the closest relay node.

Analysis of the Inframetric-ness of the Internet We empirically analyze the above inframetric model. Given a set of targets, we compute the minimal ρ that satisfies Eq (8). Figure 3 plots the CCDF of the inframetric parameter ρ of the RTT data sets. By varying the number L of targets, we see that the average RTT to a group of targets is comparable to the pairwise RTT values: most of the ρ values of the triples are lower than two. Therefore, selecting $\rho = 3$ is quite reasonable to model most of the triples.



(a) Mat2500.

(b) Host479.

Fig. 3. The CCDF of inframetric parameter ρ for the RTT data sets.

Implications Given Eq (11), if each node u obtains all nodes from the ball $B_u(\rho r)$, then node u is either able to find a node v that is β times closer to the targets, or node u must be the closest relay to the targets. Following the above recursive procedure at each intermediate node u, we are able to locate a relay that has the minimal average latency to the targets.

C. Growth in the Inframetric Model

We next define a generalized grid dimension to represent the average distances to a set of targets.

Definition 3: For a set of targets $\mathbf{T}$, assume that α satisfies

$$|B_{\mathbf{T}}(2r)| \leq 2^{\alpha} |B_{\mathbf{T}}(r)| \tag{12}$$

where $B_{\mathbf{T}}(r)$ denotes the set of service nodes whose average delay to the targets $\mathbf{T}$ is not larger than r. Then we call the smallest α when (12) holds is the **generalized grid dimension**.

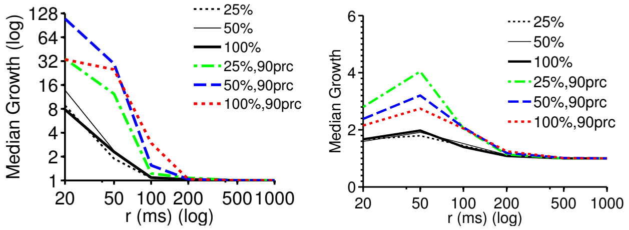
The generalized grid dimension bounds the cardinality of nodes whose delays to targets $\mathbf{T}$ are within certain intervals, which is especially suitable to represent nodes close to the targets.

We next extend the grid dimension in the metric space to the growth metric in the inframetric space that is more general to represent the latency space. The growth metric represents the ratio between the number of nodes covered by two sets $B_{\mathbf{T}}(\rho r)$ and $B_{\mathbf{T}}(r)$ with respect to a set $\mathbf{T}$ of targets:

Definition 4 (Growth [9]): Given a $(\mathbf{T}, \rho)$-inframetric model, for any $r \in \Re^{+}$, $(\gamma_\rho)^{\mathbf{T}} \in \Re^{+}$ and $u \in S_V$, if $|B_{\mathbf{T}}(\rho r)| \leq (\gamma_\rho)^{\mathbf{T}} |B_{\mathbf{T}}(r)|$, the $(\mathbf{T}, \rho)$-inframetric model is said to have a growth value $(\gamma_\rho)^{\mathbf{T}} \geq 1$.

A low growth value $(\gamma_\rho)^{\mathbf{T}}$ means that the number of nodes covered by the set $B_{\mathbf{T}}(\rho r)$ is comparable to the number of nodes covered by $B_{\mathbf{T}}(r)$ of smaller radius. As we expand the radius of the set centered for a set of targets, new nodes in $B_{\mathbf{T}}$ "come into view" at a constant rate. This implies that each node can find a node that is closer to the targets than node u by uniformly sampling a modest number of nodes.

For any node u, we compute the growth by determining the ratio of the cardinality between the set $B_{\mathbf{T}}(\rho r)$ and the set $B_{\mathbf{T}}(r)$ for a variable r. We compute the median and 90th percentile growth values for 25%, 50% and 100% of nodes from the whole data sets. From Figure 4, the median growth of both data sets is relatively small, and declines quickly with increasing radii. As a result, we can choose a modest growth value to represent common growth trends. On the other hand, the 90th percentile growth shows divergent dynamics for different data sets. Therefore, outliers exist for the growth dynamics in the $(\mathbf{T}, \rho)$-inframetric model.



(a) Mat2500.

(b) Host479.

Fig. 4. The distributions of the growth metric $(\gamma_\rho)^{\mathbf{T}}$ for the RTT data sets. The number of targets $L = 2$.

Having defined the growth, we are able to quantify the differences of the cardinality of balls with identical centers but different radii.

Lemma 3.3: Given a $(\mathbf{T}, \rho)$-inframetric with growth $\gamma_\rho^{\mathbf{T}} \geq 1$, for any $x \geq \rho$, $r > 0$ and any node u, the cardinality of the set $B_{\mathbf{T}}(r)$ is at most x^{α_ρ} times smaller than that of the ball $B_{\mathbf{T}}(xr)$, where $\alpha_\rho \in \left[\log_\rho \gamma_g^{\mathbf{T}}, 2\log_\rho \gamma_g^{\mathbf{T}}\right]$.

IV. RelayGreedy: Greedy is Optimal

We propose RelayGreedy, a method for selecting the relay near to the targets based on the intuitions of the average triangle inequality and the inframetric model. RelayGreedy guarantees the accuracy and scalability of the relay-selection procedure.

RelayGreedy manages the closed balls via the concentric ring [4], [6]. Each node u's concentric ring provides multi-level closed balls centered at each node u. When each node receives a request of selecting relays, this node obtains candidate neighbors that may be closer to the targets Then this node recursively selects the neighbor that has the lowest average RTT to the targets. The search terminates when no such better nodes can be found. Finally, the found relay is returned for delivering messages for the group of targets. Algorithm 1 summarizes the steps.

A. Summary of Theoretical Results

We first define the approximation ratios to quantify the accuracy of the found relays.

Definition 5: Let $\bar{d}_*$ denote the minimum average delay from the optimal service node u_* to a set of targets $\mathbf{T}$. If the average delay $\bar{d}_{A\mathbf{T}}$ from the found service node A to $\mathbf{T}$ is smaller than $\omega\bar{d}_*$, the found nearest service node A is called an ω**-approximation** $(\omega \geq 1)$.

The smaller ω, the closer the found node is to the optimal relay for the targets. We next show that in RelayGreedy, each node u is able to find one node v that is β times closer to the targets w.h.p.

Theorem 4.1: Let B_{ui} be the ball $B_u\left(2^i\right)$. Let $S_{ui} = B_{ui} \backslash B_{(u,i-1)}$ be the i-th ring in the concentric ring. Given a $(\mathbf{T}, \rho)$-inframetric d with growth $(\gamma_\rho)^{\mathbf{T}} \geq 1$, a node u,

Algorithm 1: Basic steps of locating the relay.

```
Relay($u$, $\mathbf{T}$, $\rho$, $\beta$)
input  : Current node $u$, the set of targets $\mathbf{T}$, the inframetric
         parameter $\rho$, the latency-reduction threshold $\beta$.
output: The next-hop node that is closer to $\mathbf{T}$.
Obtain the RTT values from node $u$ to targets;
Compute the average RTT value $r$ from node $u$ to targets:
$r \leftarrow \bar{d}_{u\mathbf{T}}$;
Node $u$ selects neighbors, denoted as $S_u$, from rings numbered
within $[1, \lceil \log_2(\rho r) \rceil]$ in its own concentric ring;
Node $A \leftarrow$ the closest neighbor to $\mathbf{T}$ from the concentric ring;
if $\bar{d}_{A\mathbf{T}} \leq \beta r$ then
    Relay($A$, $\mathbf{T}$, $\rho$, $\beta$);
else
    return $A$;
end
```

and targets $\mathbf{T}$. Let $r = \bar{d}_{u\mathbf{T}}$. The number of nodes in each ring is $O(\log(N))$. There exists a ring whose number is in $[1, \lceil \log_2(\rho r) \rceil]$ satisfying that selecting all neighbors on that ring will find one node covered by $B_{\mathbf{T}}(\beta r)$ with at least a probability $(1 - N^{-c})$, where N denotes the number of service nodes, $c > 1$.

Theorem 4.1 is proved in section V-A. We next characterize the accuracy and efficiency of Algorithm 1 as follows.

Theorem 4.2: Given a set of targets $\mathbf{T}$, Algorithm 1 stops in at most $\log_{\frac{1}{\beta_{real}}}(\Delta_d)$ steps, where $\beta_{real} < 1$ denotes the average delay reduction per step and Δ_d is the ratio of the maximum delay to the minimum delay in the delay space. Algorithm 1 has an $\frac{1}{\beta}$-approximation with a probability $1 - N^{-c_2}$, N denotes the number of service nodes and $c_2 > 1$.

Theorem 4.2 is proved via the following corollaries 5.4 and 5.5. Our results show that RelayGreedy is optimal and terminates quickly with modest requirements. As a result, RelayGreedy efficiently solves the relay-selection problem.

Parameter recommendation: Based on the analysis, we set RelayGreedy's delay reduction threshold β to 1 by default, in order to obtain the best possible relays. Further, since we only need $O(\log(N))$ neighbors for each ring to find closer neighbors towards the targets, we set the default number of neighbors in each concentric ring to 8.

V. Theoretical Framework

Having presented the ideas of the distributed relay-search process, we next derive performance guarantees under our proposed models for the latency metric. The building block of the analysis is locating a node that is closer to the targets with high probability in a distributed environment based on the random sampling process. We next analyze the number of required samples required to find a neighbor closer to the targets with high probability. Finally, we obtain the performance bounds for Algorithm 1.

A. Sampling a Node that is Closer to the Targets

We next analyze how many neighbor nodes a node u needs to sample to find w.h.p at least one node that lies in the set $B_{\mathbf{T}}(\beta r)$.

Let a constant $r = \beta \bar{d}_{u\mathbf{T}}$. In order to reduce the average distance to the targets by at least β times, each distributed relaying step needs to sample at least a node from the set $B_{\mathbf{T}}(r)$, where the set $B_{\mathbf{T}}(r)$ be the set of nodes whose average delays to targets $\mathbf{T}$ are not larger than r.

Theorem 5.1 bounds the number of required samples to find a node in the $B_{\mathbf{T}}(r)$ in the metric space.

Theorem 5.1: Let the constants $\beta \in (0, 1]$, $\alpha > 1$, $c > 1$, $r > 0$. Let N be the number of nodes. Assume that the current node is u. The targets are represented by $\mathbf{T}$. When we sample $O(\ln N)$ nodes from the ball $B_u\left(2^j\right)$, where $j = \left\lceil \log\left(\bar{d}_{u\mathbf{T}}(1+\beta)\right) \right\rceil$, with a probability at least $1 - 1/N^c$ (i.e., w.h.p), we can locate at least a node v whose average distance to the targets $\mathbf{T}$ is at most β times of that from the current node u to the targets $\mathbf{T}$.

Proof: We show the relation between the set $B_{\mathbf{T}}(r)$ and the ball with the center at node u. First, we select the smallest j that satisfies $B_{\mathbf{T}}(r)$

$$\bar{d}_{u\mathbf{T}} + r \leq 2^j \tag{13}$$

In other words, the integer j is

$$j = \left\lceil \log\left(\bar{d}_{u\mathbf{T}}(1+\beta)\right) \right\rceil \tag{14}$$

Therefore, we have

$$\bar{d}_{u\mathbf{T}} + r \geq 2^{j-1} \tag{15}$$

Furthermore, since for any node $v \in B_{\mathbf{T}}(r)$, it holds that

$$\bar{d}_{v\mathbf{T}} \leq r \tag{16}$$

Therefore, we can see that

$$\begin{aligned} d_{uq} &\leq \bar{d}_{u\mathbf{T}} + \bar{d}_{v\mathbf{T}} \\ &\leq \bar{d}_{u\mathbf{T}} + r \\ &\leq 2^j \end{aligned} \tag{17}$$

which implies that $B_{\mathbf{T}}(r)$ is covered by the ball $B_u\left(2^j\right)$:

$$B_{\mathbf{T}}(r) \subseteq B_u\left(2^j\right) \tag{18}$$

We next bound the cardinality between the ball $B_u\left(2^j\right)$ and that of $B_{\mathbf{T}}(r)$. First, for any node $v \in B_u\left(2^j\right)$, i.e., $d_{uv} \leq 2^j$, by the average triangle inequality, we have

$$\begin{aligned} \bar{d}_{v\mathbf{T}} &\leq d_{uv} + \bar{d}_{u\mathbf{T}} \\ &\leq 2^j + \bar{d}_{u\mathbf{T}} \end{aligned} \tag{19}$$

As a result, we can see that

$$B_u\left(2^j\right) \subset B_{\mathbf{T}}\left(2^j + \bar{d}_{u\mathbf{T}}\right) \tag{20}$$

Second, by $\bar{d}_{u\mathbf{T}} = \frac{1}{\beta} r$ and (15), we can bound the coverage relation as

$$\begin{aligned} & B_{\mathbf{T}}\left(2^j + \bar{d}_{u\mathbf{T}}\right) \subset B_{\mathbf{T}}\left(2^{j+1} - r\right) \\ & \subset B_{\mathbf{T}}\left(\left(\frac{r}{\beta} + r\right) \times 4 - r\right) \\ & = B_{\mathbf{T}}\left(\left(3 + \frac{4}{\beta}\right) r\right) \\ & = B_{\mathbf{T}}(\gamma r) \end{aligned} \tag{21}$$

where $\gamma = 3 + 4/\beta$. Therefore, we have

$$B_u\left(2^j\right) \subset B_{\mathbf{T}}(\gamma r) \tag{22}$$

By the generalized grid dimension, we know that

$$\left|B_{\mathbf{T}}\left(\gamma r\right)\right| \leq \gamma^{\alpha}\left|B_{\mathbf{T}}\left(r\right)\right| \tag{23}$$

As a result, by combining (20), (21) and (23), we can compute the cardinality relation between the ball $B_u\left(2^j\right)$ and the set $B_{\mathbf{T}}\left(r\right)$ as:

$$\left|B_u\left(2^j\right)\right| \leq \gamma^{\alpha}\left|B_{\mathbf{T}}\left(r\right)\right| \tag{24}$$

Then, suppose that we randomly sample a node from the ball $B_u\left(2^j\right)$, this node is covered by the set $B_{\mathbf{T}}\left(r\right)$ with a probability at least

$$\frac{\left|B_{\mathbf{T}}\left(r\right)\right|}{\left|B_u\left(2^j\right)\right|} \geq \frac{\left|B_{\mathbf{T}}\left(r\right)\right|}{\left|B_{\mathbf{T}}\left(\gamma r\right)\right|} \geq \frac{1}{\gamma^{\alpha}} \tag{25}$$

As a result, when we select k nodes uniformly at random from the ball $B_u\left(2^j\right)$, the **failure probability** p_f that all k nodes are not in the set $B_{\mathbf{T}}\left(r\right)$ is at most

$$p_f = \left(1 - 1/\gamma^{\alpha}\right)^k \tag{26}$$

Suppose that we need to control the failure probability p_f to be under $1/N^c$ for $c > 1$, i.e., $p_f = N^{-c}$, we need to ensure the size k to fulfill the constraint

$$\begin{aligned} k &= -c\ln N/\ln\left(1 - 1/\gamma^{\alpha}\right) \\ &= O\left(\ln N\right) \end{aligned} \tag{27}$$

which ensures to find a β times closer neighbor at each relay search step. ■

For the inframetric space, we next analyze how many neighbor nodes a node u needs to sample to find w.h.p at least one node that lies in the set $B_{\mathbf{T}}\left(\beta r\right)$.

We first show the inclusion relation of balls with different centers, which generalizes the inclusion of balls around a node pair in the *metric space* [22].

Lemma 5.2 (Sandwich lemma): Given a $(\mathbf{T}, \rho)$-inframetric model, for any pair of nodes u and v, and $d_{uv} \leq r$, then

$$B_{\mathbf{T}}\left(r\right) \subseteq B_u\left(\rho r\right) \subseteq B_{\mathbf{T}}\left(\rho^2 r\right) \tag{28}$$

Proof: For any node $v \in B_{\mathbf{T}}\left(r\right)$, node v satisfies

$$d_{uv} \leq \rho\max\left\{\bar{d}_{u\mathbf{T}}, \bar{d}_{v\mathbf{T}}\right\} \leq \rho r \tag{29}$$

which follows that

$$B_{\mathbf{T}}\left(r\right) \subseteq B_u\left(\rho r\right) \tag{30}$$

For any node $y \in B_u\left(\rho r\right)$, we have

$$\bar{d}_{y\mathbf{T}} \leq \rho\max\left\{\bar{d}_{u\mathbf{T}}, d_{yu}\right\} \leq \rho^2 r \tag{31}$$

which implies that

$$B_u\left(\rho r\right) \subseteq B_{\mathbf{T}}\left(\rho^2 r\right) \tag{32}$$

■

Next with the Sandwich lemma 5.2, we are able to analyze how many neighbor nodes a node u needs to sample to find w.h.p at least one node lies in the set $B_{\mathbf{T}}\left(\beta r\right)$.

Theorem 5.3 (Sampling in balls): Given a $(\mathbf{T}, \rho)$-inframetric d with growth $\gamma_{\rho}^{\mathbf{T}} \geq 1$, a node u and targets $\mathbf{T}$ satisfying $\bar{d}_{u\mathbf{T}} \leq r$. For any $\beta \in (0, 1]$, let N denote the number of service nodes, and $c > 1$. If u selects $O\left(\ln N\right)$ nodes uniformly at random with replacement from the ball $B_u\left(\rho r\right)$, then with a probability larger or equal than $1 - N^{-c}$ one of sampled nodes will lie in $B_{\mathbf{T}}\left(\beta r\right)$.

Proof: From Lemma 5.2, we have $B_{\mathbf{T}}\left(\beta r\right) \subseteq B_{\mathbf{T}}\left(r\right) \subseteq B_u\left(\rho r\right)$. Nodes in the set $B_{\mathbf{T}}\left(\beta r\right)$ are also covered by the ball $B_u\left(\rho r\right)$. Therefore, we only need to sample enough nodes in $B_u\left(\rho r\right)$ in order to sample a node located in $B_{\mathbf{T}}\left(\beta r\right)$.

Since $\left|B_u\left(\rho r\right)\right| \leq \left|B_{\mathbf{T}}\left(\rho^2 r\right)\right| = \left|B_{\mathbf{T}}\left(\frac{\rho^2}{\beta}\beta r\right)\right|$, we have

$$\left|B_u\left(\rho r\right)\right| \leq \left|B_{\mathbf{T}}\left(\frac{\rho^2}{\beta}\beta r\right)\right| \leq \left(\frac{\rho^2}{\beta}\right)^{\alpha_{\rho}}\left|B_{\mathbf{T}}\left(\beta r\right)\right| \tag{33}$$

where $\alpha_{\rho} \in \left[\log_{\rho}\gamma_g^{\mathbf{T}}, 2\log_{\rho}\gamma_g^{\mathbf{T}}\right]$.

Therefore, the probability of uniformly sampling a node from $B_u\left(\rho r\right)$ that lies in the set $B_{\mathbf{T}}\left(\beta r\right)$ is:

$$\frac{\left|B_{\mathbf{T}}\left(\beta r\right)\right|}{\left|B_u\left(\rho r\right)\right|} \geq \frac{\left|B_{\mathbf{T}}\left(\beta r\right)\right|}{\left(\frac{\rho^2}{\beta}\right)^{\alpha_{\rho}}\left|B_{\mathbf{T}}\left(\beta r\right)\right|} = \frac{1}{\left(\frac{\rho^2}{\beta}\right)^{\alpha_{\rho}}} \tag{34}$$

Let $\gamma = \left(\rho^2/\beta\right)$. Let the number of samples be l. The probability of failing to sample a node in the set $B_{\mathbf{T}}\left(\beta r\right)$ is at most

$$\left(1 - 1/\gamma^{\alpha_{\rho}}\right)^l$$

In order to obtain the failure probability to be within N^{-c}, where $c > 1$, i.e., $\left(1 - 1/\gamma^{\alpha_{\rho}}\right)^l \leq N^{-c}$, the number l of samples must be at least

$$\begin{aligned} l &= -\tfrac{c}{\ln\left(1 - 1/\gamma^{\alpha_{\rho}}\right)}\ln N \\ &= -\tfrac{c}{\ln\left(1 - \left(\beta/\rho^2\right)^{\alpha_{\rho}}\right)}\ln N \\ &= O\left(\ln N\right) \end{aligned}$$

As a result, with a probability of at least $\left(1 - N^{-c}\right)$, the current node is able to locate a neighbor that lands in the set $B_{\mathbf{T}}\left(\beta r\right)$. ■

We next extend the sampling conditions to the concentric-ring settings, proving Theorem 4.1:

Proof: Assume that each ring contains $O\left(\ln N\right)$ nodes and the nodes at each ring are uniformly sampled from the whole set of nodes that fall into that ring. Let

$$r_* = \beta r \tag{35}$$

We select the minimum positive integer i such that

$$\rho\max\left\{r, r_*\right\} = \rho r \leq 2^i \tag{36}$$

holds. As a result, the inequality

$$\rho r > 2^{i-1} \tag{37}$$

also holds, because otherwise, $(i - 1)$ will become the minimum integer satisfying Eq (36). Besides, we can see that $i = \left\lceil \log_2\left(\rho r\right)\right\rceil$.

For a node j from the set $B_{\mathbf{T}}\left(r_*\right)$, i.e.,

$$\bar{d}_{\mathbf{T}j} \leq r_* \tag{38}$$

By Definition 2, we know that

$$d_{uj} \leq \rho\max\left\{\bar{d}_{u\mathbf{T}}, \bar{d}_{\mathbf{T}j}\right\} \leq \rho\max\left\{r, r_*\right\} = \rho r \overset{Eq(36)}{\leq} 2^i \tag{39}$$

Therefore, node j is covered by the ball B_{ui}. As a result, the set $B_{\mathbf{T}}(r_*)$ is covered by B_{ui}, i.e.,

$$B_{\mathbf{T}}(r_*) \subseteq B_{ui} \tag{40}$$

In other words, in order to obtain a sample from the set $B_{\mathbf{T}}(r_*)$, we only need to select sufficient nodes from the ball B_{ui}.

(1) By multiplying two at both sides of Eq (37), we have

$$2^i < 2\rho r \tag{41}$$

Therefore, by multiplying ρ at both sides of Eq (41), it follows that

$$B_{\mathbf{T}}\left(\rho 2^i\right) \subset B_{\mathbf{T}}\left(\rho\left(2\rho r\right)\right) \tag{42}$$

Moreover, for any node $j \in B_{ui}$, we know that

$$\bar{d}_{\mathbf{T}j} \le \rho \max\left\{\bar{d}_{u\mathbf{T}}, d_{uj}\right\} \le \rho \max\left\{r, 2^i\right\} \overset{Eq(36)}{=} \rho 2^i \tag{43}$$

by the Definition 2. As a result, the ball B_{ui} is covered by the set $B_{\mathbf{T}}\left(\rho 2^i\right)$:

$$B_{ui} \subseteq B_{\mathbf{T}}\left(\rho 2^i\right) \tag{44}$$

Combining Eq (42) and Eq (44), we know that B_{ui} is covered by $B_{\mathbf{T}}\left(\rho\left(2\rho r\right)\right)$:

$$B_{ui} \subset B_{\mathbf{T}}\left(\rho\left(2\rho r\right)\right) \tag{45}$$

Since $r = \frac{r_*}{\beta}$ based on Eq (35), Eq (45) can be transformed to be:

$$B_{ui} \subset B_{\mathbf{T}}((2\rho^2/\beta)r_*) \tag{46}$$

By the definition of the growth metric, we calculate the cardinality difference between the ball B_{ui} and $B_{\mathbf{T}}((2\rho^2/\beta)r_*)$ as follows:

$$|B_{ui}| < \left(2\rho^2/\beta\right)^{\alpha} |B_{\mathbf{T}}(r_*)| \tag{47}$$

As a result, the probability of uniformly sampling a node from B_{ui} that lies in the ball $B_{\mathbf{T}}(r_*)$ is:

$$\frac{|B_{\mathbf{T}}(r_*)|}{|B_{ui}|} > \frac{|B_{\mathbf{T}}(r_*)|}{\left(2\rho^2/\beta\right)^{\alpha} |B_{\mathbf{T}}(r_*)|} = \frac{1}{\left(2\rho^2/\beta\right)^{\alpha}} \tag{48}$$

(2) Suppose that the size of the ring is

$$\begin{aligned}
&\left(2\rho^2/\beta\right)^{\alpha} \log\left(N/N^{-c}\right) \\
&= (c+1)\left(\rho/\beta\right)^{\alpha} \log\left(N\right) \\
&= O\left(\log\left(N\right)\right)
\end{aligned}$$

By Theorem 4.1 in [4], we can see that some node from a ring S_{ul}, $l \le i$ lands in the set $B_{\mathbf{T}}(r_*)$ with a failure probability $\left(N^{-c}\right)/N^2 < N^{-c}$. The proof is complete. ■

For a set of targets T, Theorem 4.1 shows that we only need $O\left(\log\left(N\right)\right)$ neighbors per ring to find at least one node that lies in the closed set $B_{\mathbf{T}}(\beta r)$ w.h.p.

B. *Algorithm Analysis*

Having proven the number of samples required at each step, we can prove Algorithm 1's performance. We first derive an upper bound on the number of hops.

Corollary 5.4: Algorithm 1 stops in at most $\log_{\frac{1}{\beta_{real}}}\left(\Delta_d\right)$ steps, where $\beta_{real} < 1$ denotes the average delay reduction per step and Δ_d is the ratio of the maximum delay to the minimum delay in the delay space.

Corollary 5.4 follows the similar proofs of Theorem 4.4 in [6], since both relay on the β-times-closer greedy optimization.

Next, we prove that Algorithm 1 locates approximately optimal relays as the threshold β approaches 1. When $\beta = 1$, Algorithm 1 locates the optimal results w.h.p.

Corollary 5.5: Algorithm 1 locates an $\frac{1}{\beta}$-approximation relays with respect to the optimal relay node for any set of targets with a probability $1 - N^{-c_2}$, where N denotes the number of service nodes and $c_2 > 1$.

Proof: Suppose that u_* is the ground-truth nearest server to the targets $\mathbf{T}$. Suppose that a node u forwards the relay request to another node v by Algorithm 1, the *progress* of the relay process is calculated as the ratio of $\frac{\bar{d}_{u\mathbf{T}}}{\bar{d}_{v\mathbf{T}}}$, which is at least $\frac{1}{\beta}$ by Theorem 4.1 in the inframetric space (or by Theorem 5.1 in the metric space).

First, let p be the probability of finding a neighbor v that is β times closer to the targets at a step. Based on the sampling conditions of Theorem 4.1, $p \ge 1 - N^{-c}$. As a result, the failure probability of l steps is at most

$$\begin{aligned}
&1 - p^l = \left(1 - \left(1 - N^{-c}\right)^l\right) \\
&\approx 1 - e^{-l/N^c} \\
&\approx 1 - \left(1 - l/N^c\right) \\
&\approx \left(N^{-c_2}\right)
\end{aligned} \tag{49}$$

due to the Taylor's expansion, where $c_2 = c - \log_N l > 1$ since the number l of search steps satisfies $l \ll N$ by the corollary 5.4. As a result, the probability of finding a neighbor satisfying the sampling condition in Theorem 5.3 after l steps is at least $1 - N^{-c_2}$, i.e., w.h.p.

Second, assume that Algorithm 1 locates a node u_x as the nearest server and has an approximation ratio larger than $\frac{1}{\beta}$ i.e., $\bar{d}_{u_x\mathbf{T}} > \frac{1}{\beta}\bar{d}_{u_*\mathbf{T}}$. We disprove the approximation ratio by contradiction.

Since $\beta \le 1$, we see that $\bar{d}_{u_x\mathbf{T}} > \bar{d}_{u_*\mathbf{T}}$. As a result, we can locate a new node β times closer to the targets than u_x w.h.p. As a result, the search process can be continued, which contradicts the fact that the search process stops at node u_x. Therefore, the approximation ratio of the found node must be at most $\frac{1}{\beta}$, which completes the proof. ■

VI. Simulation

In this section, we compare RelayGreedy's performance with state-of-art methods.

Experimental Setup We have implemented a simulator. The simulator randomly selects a set of nodes as the service nodes (800 by default) and randomly selects nodes in the data sets as the targets that need to select the relay to forward messages sent and received between targets. In the simulator, for each relaying request, we sample a set of targets uniformly at random from the whole set of nodes and choose a service node to receive the request for these targets. The simulation consists of 10,000 relay requests for different sets of targets. We repeat the above simulation five times.

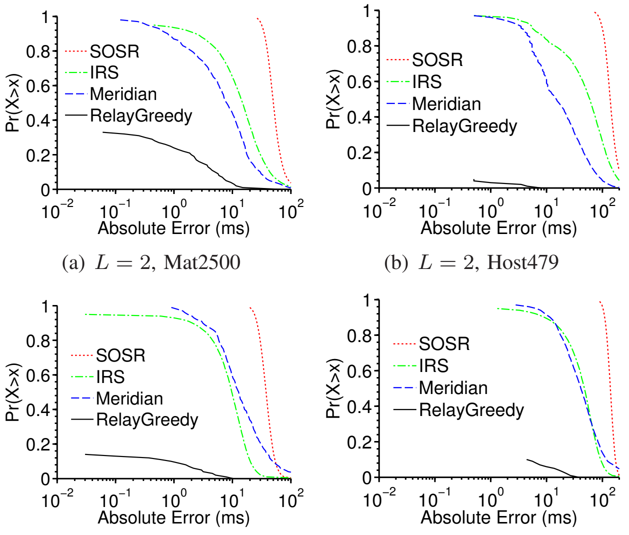
Comparison We compare the **absolute error** of different methods:

$$\left|\bar{d}_{p\mathbf{T}} - \bar{d}_{P^*\mathbf{T}}\right| = \frac{1}{L}\left|\sum_{j=1}^{L}\left|d_{PT_j}\right| - \sum_{j=1}^{L}\left|d_{P^*T_j}\right|\right|$$

where L denotes the number of targets, P denotes the estimated relay, P^* is the ground-truth optimal relay, and $(T_1, \ldots, T_L)$ represents the set of targets. We also evaluated the sensitivity of parameters for RelayGreedy, which is reasonably robust against the parameter choices. The simulator compares RelayGreedy with the following methods: (i) **SOSR** [17], selects a node uniformly at random from the whole set of nodes as the relay; (ii) **Meridian** [4], recursively locates a neighbor with concentric rings managed with gossip protocols. For fair comparison, we select Meridian's candidate neighbors based on the same rules as RelayGreedy; (iii) **IRS** [20], constructs a routing overlay where each node serves as the relay for its neighbors. Neighbors are selected via gossip protocols as those have the highest embedding errors, corresponding to those causing TIVs in the latency space.

We report the comparison results for all methods in Figure 6. We see that the SOSR method has the largest absolute errors among all methods, since the relay node nearest to the targets may be orders of magnitude closer to the targets than a randomly sampled node because of the clustering structure of the network latency space. IRS incurs relatively high absolute errors. This is because embedding errors are also caused by poor convergence or inherent embedding distortions [23]. RelayGreedy outperforms other methods by orders of magnitudes. In more than 60% of all tests, RelayGreedy is able to find the ground-truth nearest relay nodes to the targets. Further, for more than 90% of all experiments, RelayGreedy's absolute error is less than 10 ms. Comparatively, Meridian yields much worse performance than RelayGreedy, since Meridian's ring maintenance process converges quite slowly because of the clustering phenomenon in the network latency space [6], which makes the relay search process be trapped at much poorer local minima.

We next analyze the scalability of the methods with increasing service nodes. We fix the number of neighbors at each ring for both Meridian and RelayGreedy. Figure 6 (a) shows the accuracy of found relays by Meridian and RelayGreedy. Meridian has steady accuracy since it is able to find stable local minimum with $O(\log N)$ samples, where N denotes the size of service nodes. RelayGreedy decreases the absolute errors with increasing service nodes, since relays are placed in wider areas in the network latency space, so that RelayGreedy is able to find relays that are closer to the targets.



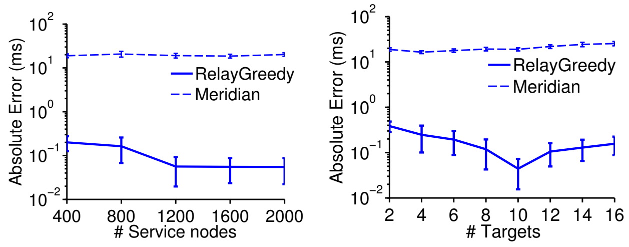
(a) $L = 2$, Mat2500

(b) $L = 2$, Host479

(c) $L = 10$, Mat2500

(d) $L = 10$, Host479

Fig. 5. The CCDF distributions of the absolute errors for relay selection methods.



(a) Vary Service nodes.

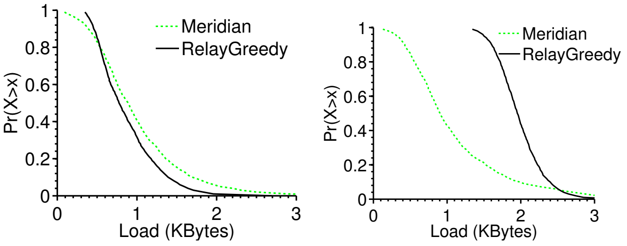
(b) Vary targets

Fig. 6. The mean absolute errors of the found relays as we vary the number of targets and the number of service nodes.

We next test the performance of finding the relays as we vary the size of targets. Figure 6 (b) plots the mean absolute errors. RelayGreedy has orders of magnitudes lower absolute errors compared to Meridian, since RelayGreedy employs several performance-guaranteed techniques to avoid bad local minimums. We see that RelayGreedy significantly decreases the mean absolute errors until number L of targets reaches 10. This is because increasing targets reduces the ρ numbers of the extended inframetric model, ' which increases the probability of finding suitable candidates to find good relays according to Algorithm 1. The absolute errors increases moderately for RelayGreedy when the number L of Targets is beyond 10, due to higher errors of RTT prediction with network coordinates.

We next compare the messaging costs between our method and Meridian, which is defined as the sum of the sizes of the recursive-search messages exchanged among neighbors and the delay probe messages from service nodes to targets: $\sum_{l=1}^{L}\left(|m_l^q| + |m_l^p|\right)$, where L represents the number of search steps, $|m_l^q|$ and $|m_l^p|$ denote the sizes of the search messages and delay probe messages of the l-th hop, respectively. For simplicity, we set the size of each message to 50 bytes. We record the messaging costs of 10,000 relay search procedures.

From Figure 7, we see that RelayGreedy requires higher messaging costs than Meridian when the number of targets is 10, since RelayGreedy have much larger number of candidate neighbors to probe. While Meridian's concentric ring may have insufficient neighbors for some rings, resulting in lower messaging costs than those of RelayGreedy.



(a) $L = 2$

(b) $L = 10$

Fig. 7. Query messaging costs of RelayGreedy and Meridian on the Mat2500 data set.

VII. Conclusion

We have addressed the problem of selecting the latency-optimal relay to forward real-time messages to multiple target nodes. We rigorously formalize the problem of finding the relay that is nearest to the targets, with both metric space and the inframetric models. Our metric space model is suitable for modelling most of the triples in the latency space, while our refined inframetric model is general enough to allow for asymmetric distances and the triangle inequality violations. We next present a simple greedy algorithm to find the optimal relay node with high probability, by recursively sampling a modest number of nodes in a distributed manner. We believe that our theoretical and algorithmic results are general enough for diverse user-facing cloud services.

Acknowledgements

We appreciate a lot to Dr. Ernst Biersack for his constructive suggestions. We gratefully thank the anonymous reviewers for their comments. This work was supported by the National Natural Science Foundation of China (Grant No. 61402509 and 61379052), the National High Technology Research and Development 863 Program of China (Grant No.2013AA01A213), the Natural Science Foundation for Distinguished Young Scholars of Hunan Province (Grant No.14JJ1026), Specialized Research Fund for the Doctoral Program of Higher Education (Grant No.20124307110015)

References

[1] X. Lu, H. Wang, J. Wang, J. Xu, and D. Li, "Internet-based Virtual Computing Environment: Beyond the Data Center as a Computer," *Future Generation Comp. Syst.*, vol. 29, no. 1, pp. 309–322, 2013.

[2] T. Flach, N. Dukkipati, A. Terzis, B. Raghavan, N. Cardwell, Y. Cheng, A. Jain, S. Hao, E. Katz-Bassett, and R. Govindan, "Reducing Web Latency: the Virtue of Gentle Aggression," in *Proc. of SIGCOMM*, 2013, pp. 159–170.

[3] Y. Chen, R. Mahajan, B. Sridharan, and Z.-L. Zhang, "A Provider-side View of Web Search Response Time," in *Proc. of SIGCOMM*, 2013, pp. 243–254.

[4] B. Wong, A. Slivkins, and E. G. Sirer, "Meridian: a Lightweight Network Location Service Without Virtual Coordinates," in *Proc. of SIGCOMM 2005*, pp. 85–96.

[5] V. Vishnumurthy and P. Francis, "On the Difficulty of Finding the Nearest Peer in P2P Systems," in *Proc. of IMC 2008*, pp. 9–14.

[6] Y. Fu, Y. Wang, and E. Biersack, "HybridNN: An Accurate and Scalable Network Location Service based on the Inframetric Model," *Future Generation Comp. Syst.*, vol. 29, no. 6, pp. 1485–1504, 2013.

[7] C. Lumezanu, R. Baden, D. Levin, N. Spring, and B. Bhattacharjee, "Symbiotic Relationships in Internet Routing Overlays," in *Proc. of the 6th USENIX NSDI*, 2009, pp. 467–480.

[8] C. Ly, C.-H. Hsu, and M. Hefeeda, "Irs: A detour routing system to improve quality of online games," *IEEE Transactions on Multimedia*, vol. 13, no. 4, pp. 733–747, 2011.

[9] P. Fraigniaud, E. Lebhar, and L. Viennot, "The Inframetric Model for the Internet," in *Proc. of INFOCOM 2008*, pp. 1085–1093.

[10] D. R. Choffnes, M. Sanchez, and F. E. Bustamante, "Network positioning from the edge - an empirical study of the effectiveness of network positioning in p2p systems," in *Proc. of IEEE INFOCOM 2010*, pp. 291–295.

[11] S. Agarwal and J. R. Lorch, "Matchmaking for Online Games and Other Latency-sensitive P2P Systems," in *Proc. of SIGCOMM*, 2009, pp. 315–326.

[12] A. Bharambe, J. R. Douceur, J. R. Lorch, T. Moscibroda, J. Pang, S. Seshan, and X. Zhuang, "Donnybrook: Enabling large-scale, high-speed, peer-to-peer games," in *Proc. of the ACM SIGCOMM*, 2008, pp. 389–400.

[13] R. Chen, I. E. Akkus, and P. Francis, "Splitx: high-performance private analytics," in *Proc. of ACM SIGCOMM*, 2013, pp. 315–326.

[14] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan, "Detour: A case for Informed Internet Routing and Transport," *IEEE Micro*, vol. 19, pp. 50–59, 1999.

[15] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "Resilient Overlay Networks," in *Proc. of SOSP'01*, pp. 31–41.

[16] A. Nakao and L. Peterson, "Scalable Routing Overlay Networks," *ACM SIGOPS Operating Systems Review*, 2006.

[17] K. P. Gummadi, H. Madhyastha, S. D. Gribble, H. M. Levy, and D. J.Wetherall, "Improving the Reliability of Internet Paths with One-hop Source Routing," in *Proc. of the 6th conference on Symposium on Opearting Systems Design and Implementation - Volume 6*, 2004.

[18] A.-J. Su, D. R. Choffnes, A. Kuzmanovic, and F. E. Bustamante, "Drafting behind Akamai: Travelocity-based detouring," in *Proc. of SIGCOMM'06*.

[19] C. Lumezanu, R. Baden, D. Levin, N. Spring, and B. Bhattacharjee, "Symbiotic Relationships in Internet Routing Overlays," in *Proc. of NSDI'09*, pp. 469–480.

[20] C. Ly, C. Hsu, and M. Hefeeda, "Improving Online Gaming Quality Using Detour Paths," in *Proc. of ACM Multimedia*, pp. 66–64.

[21] K. P. Gummadi, S. Saroiu, and S. D. Gribble, "King: Estimating Latency Between Arbitrary Internet End Hosts," in *Proc. of the 2nd SIGCOMM Workshop on Internet measurment*, 2002, pp. 5–18.

[22] D. R. Karger and M. Ruhl, "Finding Nearest Neighbors in Growth-restricted Metrics," in *Proc. of the thiry-fourth annual ACM symposium on Theory of computing*, 2002, pp. 741–750.

[23] S. Lee, Z.-L. Zhang, S. Sahu, and D. Saha, "On Suitability of Euclidean Embedding for Host-based Network Coordinate Systems," *IEEE/ACM Trans. Netw.*, vol. 18, no. 1, pp. 27–40, 2010.