

MIRAGE: Private, Mobility-based Routing for Censorship Evasion

Zachary Ratliff*, Ruoxing (David) Yang[†], Avery Bai[†], Harel Berger^{1‡}, Micah Sherr[†], James Mickens*

*Harvard University

Emails: zacharyratliff@g.harvard.edu, mickens@g.harvard.edu

[†]Georgetown University

Emails: ry216@georgetown.edu, yb243@georgetown.edu, micah.sherr@georgetown.edu

[‡]Ariel University

Email: harelb@ariel.ac.il

Abstract—In authoritarian and highly surveilled environments, traditional communication networks are vulnerable to censorship, monitoring, and disruption. While decentralized anonymity networks such as Tor provide strong privacy guarantees, they remain dependent on centralized Internet infrastructure, making them susceptible to large-scale blocking or shutdowns. To address these limitations, we present MIRAGE, a privacy-preserving mobility-based messaging system designed for censorship-resistant communication. MIRAGE uses a district-based routing scheme that probabilistically forwards messages based on the high-level mobility patterns of the population. To prevent leakage of individual mobility behavior, MIRAGE protects users’ mobility patterns with local differential privacy, ensuring that participation in the network does not reveal an individual’s location history through observable routing decisions.

We implement MIRAGE within *Cadence*, an open-source simulator that provides a unified framework for evaluating mobility-based protocols using approximated geographical encounters between nodes over time. We analyze the privacy and efficiency tradeoffs of MIRAGE and evaluate its performance against (1) traditional epidemic and random-walk-based routing protocols and (2) the state-of-the-art privacy-preserving geography-based routing protocol, using real-world trajectories—one from pedestrian movement patterns collected in various urban locations and another consisting of GPS traces from taxi operations. Our results demonstrate that MIRAGE significantly reduces message overhead compared to epidemic routing, and outperforms probabilistic flooding in terms of delivery rate, while providing stronger privacy guarantees than existing techniques.

I. INTRODUCTION

A rich literature exists on communication systems that try to protect the confidentiality and anonymity of messages. Unfortunately, many of these systems run atop centralized network infrastructure—infrastructure that is vulnerable to subversion by state-level actors or the technology companies which operate the infrastructure. For example, Tor [10]

provides IP-anonymous messaging via a distributed peer-to-peer routing layer which hides the senders and receivers of messages. However, Tor’s overlay network is deployed atop the traditional Internet routing layer. Thus, Tor is vulnerable to attacks by authoritarian governments who can censor, surveil, or completely shutdown the parts of the Internet under government control [1, 23, 28].

In response to these challenges, some private communication systems leverage *human-to-human* links to transmit messages [2, 3, 7, 22]. In these approaches, messages propagate between devices via short-range, peer-to-peer wireless protocols like WiFi Direct or Bluetooth Low Energy (BLE), and two devices can only exchange messages when their associated human owners come into direct physical proximity. Decentralized message forwarding atop ephemeral point-to-point links eliminates reliance on centralized routing infrastructure, providing robustness against the compromise of that infrastructure (due to attacker control or to natural disaster [21, 24]). Unfortunately, routing protocols built atop ephemeral, mobility-induced links may suffer from poor performance. For example, protocols that rely on epidemic flooding can induce network congestion, whereas random-walk-based protocols can suffer from high latency and low delivery rates [3].

To improve the performance of mobile ad-hoc routing, researchers have proposed forwarding mechanisms that leverage knowledge of predictable mobility patterns to guide message routing [2, 3, 9, 14]. For example, suppose that Alice encounters Bob and Charlie. Further suppose that Alice carries a message destined for a location that Bob (but not Charlie) is likely to visit. Alice can preferentially forward the message to Bob, such that Bob can deliver the message if he does in fact visit the message’s destination. By preferentially forwarding messages to individuals with higher probabilities of reaching the intended destinations, the routing layer improves delivery rates while reducing delivery latencies and message retransmissions.

Although mobility-aware routing improves network metrics, it significantly undermines user privacy by revealing a user’s mobility patterns to network-based attackers. For instance, if an attacker observes that a particular user frequently handles messages targeted at a specific geographic region, the attacker

¹Most work was done during the author’s time at Georgetown University.

may deduce that the user regularly visits that location. Such inference attacks could enable even modestly resourced adversaries to identify people from targeted groups, compromising user anonymity and exposing those users to significant risk.

In this paper, we introduce MIRAGE, a private, decentralized messaging system that leverages ephemeral point-to-point radio links to efficiently route messages between users. MIRAGE’s routing improves upon both epidemic-style approaches and random-walk protocols by exploiting historical mobility data collected from the population (§VII-B). For example, a simulation study of MIRAGE using real-world human movement data shows that MIRAGE can deliver $15\times$ the number of messages compared to random-walk protocols, and provides significantly better scalability (as measured by the number of concurrent messages in a network) than flooding. However (and importantly), MIRAGE provides provable privacy guarantees which ensure that a user’s participation in MIRAGE does not reveal *too much* about their individual mobility patterns, where we quantify a user’s privacy loss using differential privacy (§III).

II. OVERVIEW

MIRAGE represents physical space using a **map** M , partitioned into disjoint **districts**. The map M can be arranged in a standard grid pattern (e.g., as shown in Figure 1) or based on logical geographic divisions such as ZIP codes. Each district is thus represented as an element of the set M . At any given moment, each MIRAGE user is located in exactly one district. Given a message from Alice that is addressed to a district $d \in M$, MIRAGE attempts to route the message from Alice’s current district to d . MIRAGE does so by propagating the message outwards from Alice’s mobile device, using the ephemeral, point-to-point radio links that individual devices establish as those devices move through space (and through each other’s radio communication radii).

Alice models her mobility patterns using a **mobility profile** (§III), which is defined as a discrete probability distribution over the set of map districts M . To construct this profile, Alice’s device periodically records its current location by logging the visited district $d \in M$ into a multi-set of locations. Subsequently, MIRAGE categorizes each user’s mobility behavior by identifying their most frequently visited districts. These frequently visited districts indicate where a user is considered an *ideal* router for messages. Specifically, given a message m targeted at a district d_j within Alice’s set of frequently visited locations, Alice is considered an ideal candidate to deliver that message due to her regular presence in that area.

During a setup phase, users’ most visited districts undergo randomization to ensure local differential privacy (§VI-A), after which they are aggregated into a global set G representing *dynamic gossip parameters*. This global set captures the high-level mobility characteristics of the entire user population. Importantly, because each reported mobility profile adheres to local differential privacy, the central aggregator need not be trusted. We provide further details on this setup phase in §VI.

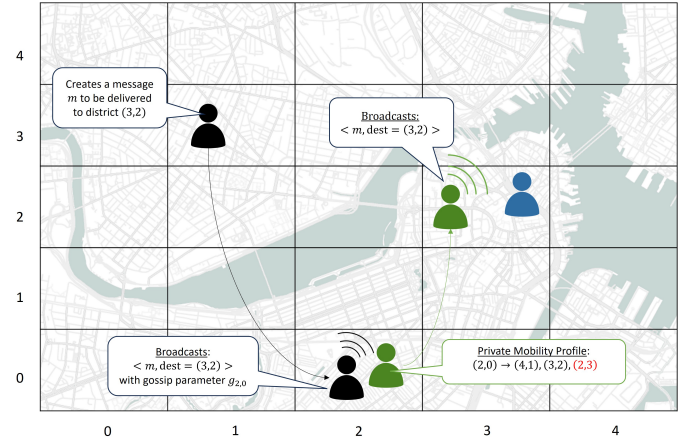


Fig. 1: An example map M consists of $5 \times 5 = 25$ districts. MIRAGE routes messages addressed to a specific district $d \in M$ through nodes likely to deliver successfully. For instance, a message targeted at district $(3, 2)$ originates in district $(1, 3)$ and is transferred to a node located in district $(2, 0)$. The initial broadcaster employs a dynamic gossip protocol with parameter $g_{(2,0)}$ to determine the number of recipients. A receiving node (green) in district $(2, 0)$ decides whether to accept the message based on its *private mobility graph*. In this graph, an edge exists from the current district to the destination district if the user frequently travels between those two locations. To preserve privacy, some edges may be artificially added (shown in red) or omitted, introducing uncertainty about the user’s true mobility patterns. Upon arrival at the destination district $(3, 2)$, the message is flooded locally to complete the delivery.

The set of dynamic gossip parameters G controls message forwarding frequency between districts to optimize successful delivery while minimizing unnecessary transmissions. Specifically, each parameter $g_{(d_i, d_j)} \in G$ is an empirical estimate of the probability that a randomly chosen user in district d_i will subsequently travel to district d_j . When a message carrier is in district d_i and propagates a message to nearby users, they use $g_{(d_i, \cdot)}$ to inform how many users to give the message to ensure a high probability of delivery to its final destination. For example, if users frequently move from district d_i to district d_j , the message may be propagated less while maintaining high delivery rates.

MIRAGE ensures that users accept and forward messages only if their private mobility graphs indicate a high likelihood of traveling between the current location and the message’s destination district. Specifically, a user accepts a message if their differentially private mobility graph (§VI-A) contains an edge from the current district to the destination district (see Figure 1). The differential privacy guarantees offered by MIRAGE provide plausible deniability, obscuring whether the user truly travels between those districts.

III. BACKGROUND

In this section, we review the relevant background information on mobility-based routing and differential privacy.

A. Mobility Profiles

A mobility profile p is a discrete probability distribution over a node's historical mobility patterns. We let $M = \{d_1, \dots, d_n\}$ be a finite set of partitioned districts. A node in the network periodically polls their location to obtain a point $d \in M$ and adds it to the multi-set V of *visited* districts. The mobility profile p is therefore the probability mass function:

$$p(d) = \begin{cases} \frac{|V_d|}{|V|} & \text{if } d \in V \\ 0 & \text{otherwise} \end{cases}$$

Given mobility profiles and a statistical measure of one's likelihood to visit a given district, highly efficient (albeit blatantly non-private) routing protocols are possible [3]. For example, user u_1 may pass a message addressed to district d to user u_2 if $p_{u_1}(d) < p_{u_2}(d)$, meaning that user u_2 is statistically more likely to deliver the message to district d than user u_1 . However, this comparison clearly leaks the values $p_{u_1}(d)$ and $p_{u_2}(d)$ in the clear, allowing any observer (including the participants themselves) to learn the mobility patterns of other users.

B. Differential Privacy

In this section, we review the relevant foundations of differential privacy.

Definition 1 (Differential Privacy [11]). *An algorithm $\mathcal{M} : \mathcal{X} \rightarrow \mathcal{Y}$ is ϵ -differentially private if for all neighboring inputs $x, x' \in \mathcal{X}$ and for all $S \subseteq \mathcal{Y}$:*

$$\Pr[\mathcal{M}(x) \in S] \leq e^\epsilon \cdot \Pr[\mathcal{M}(x') \in S]$$

Differential privacy imposes a multiplicative upper bound on the difference between output distributions when computing statistics on neighboring inputs.

We note that differential privacy is an information theoretic guarantee that holds for even computationally-unbounded adversaries. We can, therefore, precisely reason about an individual's privacy risk when their data is included in a differentially private computation. The following theorems describe the effect of post-processing and/or performing multiple differentially private multiple analyses on a dataset.

Lemma 1 (Post-processing). *If M is an ϵ -differentially private mechanism, and A is any arbitrary function, then $A(M(x))$ is ϵ -differentially private.*

Lemma 2 (Basic Composition). *If M_i is an ϵ_i -differentially private mechanism for $i = 1 \dots k$, then $(M_1(x), M_2(x), \dots, M_k(x))$ is $\sum_{i=1}^k \epsilon_i$ -differentially private.*

IV. THREAT MODEL AND PRIVACY GOALS

We consider an adversary capable of participating in the messaging protocol by observing and injecting messages, as well as controlling a subset of network nodes. However, we explicitly do *not* assume a highly resourced adversary capable of extensive physical surveillance, such as physically tracking users across locations. Instead, our threat model specifically considers attackers who can observe individual routing decisions (e.g., making inferences about Alice's location history based on the fact that Alice accepted or forwarded a message from Bob). We focus on this adversary model precisely because it reflects a realistic scenario in which attackers can periodically observe message passing patterns (e.g., during brief moments of physical proximity to a user) without engaging in continuous and resource intensive surveillance. Although this adversary has limited observation capabilities, it can still exploit routing decisions to infer sensitive details about users, such as repeated visits to specific locations or membership in targeted communities, enabling widespread surveillance of mobility patterns without constant monitoring.

Our primary security objective is to prevent adversaries from exploiting *routing decisions*—whether to transfer a message between two encountered users—to infer sensitive mobility patterns. Even limited leakage through routing behavior could allow attackers to uncover attributes such as community membership, repeated visits to sensitive locations, or involvement in private activities, leading to profiling, discrimination, or targeted surveillance. We therefore focus on defending against such routing-level inference attacks.

Formally, we define the routing decision as a potentially randomized function $f : \mathcal{U}^2 \times \mathcal{M} \rightarrow \{0, 1\}$, which takes as input a pair of users (u_1, u_2) and a message m , and outputs 1 (transfer) if the message should be exchanged or 0 (do not transfer) otherwise. We consider the user space \mathcal{U} to consist of user identifiers and associated mobility profiles. In our implementation, each user is represented by a string-valued user ID along with the user's corresponding mobility profile. Similarly, we take the message space \mathcal{M} to be a constant-sized bitstring representing message content. More generally, the user and message spaces may include richer sets of attributes such as timestamps, recent encounters, routing metadata, or message expiration information, depending on the specific requirements of the routing protocol.

When two users encounter each other, the decision to exchange a message is determined by the routing function f . In practice, evaluating f may involve an interactive protocol in which the users first exchange certain attributes or metadata (e.g., mobility profiles). For instance, consider a user Alice carrying a message m . When Alice encounters another user Bob, we say that $f((\text{alice}, \text{bob}), m) = 1$ if, after the interaction, Bob now carries the message m (potentially in addition to Alice), and 0 otherwise. While the routing protocol may depend on attributes such as message metadata (source, destination, expiration) and the users' mobility histories, preserving privacy demands that the acceptance probability not

change *too much* depending on the specific users involved. More precisely, we formalize this under differential privacy:

Definition 2 (DP Routing Function). A routing decision function $f : \mathcal{U}^2 \times \mathcal{M} \rightarrow \{0, 1\}$ for a mobility routing protocol is ϵ -differentially private if, for all $(u_1, u_2), (u'_1, u'_2) \in \mathcal{U}$, all messages $m \in \mathcal{M}$, and all outcomes $y \in \{0, 1\}$, we have:

$$\Pr[f((u_1, u_2), m) = y] \leq e^\epsilon \cdot \Pr[f((u'_1, u'_2), m) = y]$$

The above definition captures the requirement that the decision to transfer a message m from user u_1 to user u_2 should not depend too much on the specific identities (and thus the private mobility patterns) of the users involved in the exchange. The routing decision function for classic ad hoc routing protocols, such as flooding (forwarding all messages indiscriminately) or random walks (forwarding probabilistically, independently of mobility patterns), trivially satisfies Definition 2.

Finally, our threat model explicitly excludes general traffic analysis and correlation attacks. An adversary with multiple vantage points across the network might still infer general regional message origins through timing and broadcast frequency, even without mobility-informed routing (as with random walks or flooding). Defending against such attacks is beyond our scope. MIRAGE itself also does not enforce message integrity or authentication, however, one can design additional cryptographic mechanisms atop the MIRAGE routing layer to provide these complementary security guarantees.

V. CASE STUDY: STATISTICAL DISCLOSURE ATTACKS ON PPBR

In this section, we demonstrate that existing private mobility routing protocols, specifically the *probabilistic profile-based routing* (PPBR) scheme proposed by Aviv et al. [2], can inadequately protect user privacy. PPBR operates similarly to flooding-based protocols, but with the key difference that nearby users *silently accept* messages contingent on self-identifying as suitable carriers. Concretely, when Alice broadcasts a message m destined for district d , a nearby user (Bob) evaluates his suitability as a message carrier using a *marginal similarity score* defined as:

$$s(\text{bob}, d) = \frac{p_{\text{bob}}(d)}{p_{\text{general}}(d)}$$

where $p_{\text{bob}}(d)$ denotes Bob's probability¹ of visiting district d , and $p_{\text{general}}(d)$ represents the average user's probability of visiting district d , computed from a pre-calculated *general user profile* summarizing overall population mobility patterns. A high marginal similarity score for district d indicates that Bob is particularly suited to carry messages destined for that district, relative to the broader population. Consequently, Bob silently accepts a message m_d targeted at district d from user

u if d is among the top- k districts ranked by his marginal similarity scores:

$$f((u, \text{bob}), m_d) = \begin{cases} 1 & \text{if } d \in S_{\text{bob}} \\ 0 & \text{otherwise} \end{cases}$$

where S_{bob} is Bob's set of k districts corresponding to his highest marginal similarity scores.

At first glance, the silent acceptance mechanism seems to conceal individual routing decisions. However, subsequent forwarding or delivery actions inherently expose these decisions. For example, when Bob eventually broadcasts a message destined for district d , an adversary observing this action learns either that Bob was the original creator of the message, or that Bob silently accepted it from another user due to him having a high marginal similarity score for district d . Consequently, Bob's message-forwarding behavior inadvertently provides statistical information regarding his frequently visited districts.

To illustrate the severity of this privacy risk, consider a scenario in which Bob belongs to a minority group whose members predominantly live and work in a specific district d . As a result, district d consistently ranks among the highest in marginal similarity for all minority members. In contrast, members of the majority population rarely visit district d , leading to low marginal similarity scores for that location in their mobility profiles. Therefore, if an adversary observes Bob broadcasting a message m destined for district d , they can infer with high confidence that Bob is a member of the minority group, and thus frequently travels to that district.

This scenario highlights a fundamental vulnerability in PPBR. The *observable routing decisions* can be exploited to mount statistical disclosure attacks that reveal sensitive location patterns. The core issue is that PPBR lacks plausible deniability. When Bob accepts a message m for district d and later forwards that message, it directly leaks information about his true marginal similarity score and, by extension, his mobility habits.

To illustrate the severity of this leakage, we designed an experiment with a synthetic population of 1,000 users moving within a map divided into four quadrants and 100 districts. Of these users, 800 were *majority* users who resided in uniformly random districts within quadrant 3, representing general residential areas. The remaining 200 were *minority* users concentrated within a single targeted district located in quadrant 1, representing a geographically localized minority community. Each user's workplace was randomly assigned to districts within quadrant 4, representing a city's downtown or business district. Users moved probabilistically between their respective home districts and workplaces over 10,000 simulated epochs, generating the global mobility heatmap in Figure 2.

We then applied PPBR according to Aviv et al. [2], configuring it so that each user accepts messages addressed to districts corresponding to the top 1/10 fraction of entries in their marginal similarity score vector. Under this configuration, we conducted a hypothesis test with H_0 that the user belonged

¹Including a distance-decayed weighting factor that accounts for visits to neighboring districts; omitted here for brevity.

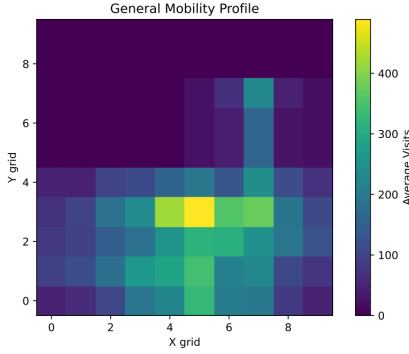


Fig. 2: Example mobility heat map for a population of 800 *majority* users and 200 *minority* users. Majority users reside in districts chosen uniformly at random within quadrant 3 and commute to districts in quadrant 4. Minority users all reside in a single district in quadrant 1 and also commute to districts in quadrant 4.

to the majority community and H_1 that the user belonged to the minority community, deciding in favor of H_1 if the user accepted messages destined for the targeted minority district.² This test perfectly identified minority users in our experiments, yielding a true positive rate (TPR) of 100% and a false positive rate (FPR) of 0%. Even under a more conservative PPBR configuration, in which users accepted messages for approximately the top 1/3 fraction of entries in their marginal similarity score vector, our hypothesis test still achieved very high accuracy, with a true positive rate of 100% and a false positive rate of only 0.4%.

In contrast, mobility networks that use DP routing functions (Definition 2) explicitly bound an adversary’s inference capabilities. If a sequence of routing decisions satisfies ϵ -DP, then any hypothesis test attempting to distinguish, for example, whether a user belongs to a minority or majority group based on the outcomes of these routing decisions, will have its TPR to FPR ratio bounded by e^ϵ . For instance, if a routing protocol satisfies ϵ -DP with $\epsilon = \ln(4)$, any hypothesis test achieving a TPR of 80% will necessarily incur an FPR of at least 20%.

VI. MIRAGE PRIVATE ROUTING

We describe how MIRAGE routes messages while protecting the privacy of individual message carriers.

A. Private Mobility Graphs

Upon installing the app, each user u constructs a mobility profile p_u , which captures their movement patterns across the map M . If the user’s location data is not immediately available, the app can locally collect this data over time by tracking the device’s location. Using the mobility profile p_u , MIRAGE identifies the $k \geq 2$ most frequently visited districts and encodes them as a *user mobility graph*.

²An adversary can test whether a user accepts a message for a targeted district by broadcasting the message to that user and then observing whether the user subsequently forwards it.

Definition 3 (User Mobility Graph). A user u ’s mobility graph, denoted as $G_u = (V_u, E_u)$, is defined as follows:

- V_u is the set of districts in the map M .
- An edge $(d_i, d_j) \in E_u$ exists between two vertices $d_i, d_j \in V_u$ if and only if both d_i and d_j are among the top k districts visited by user u , based on their mobility profile p_u .

Formally,

$$(d_i, d_j) \in E_u \iff d_i, d_j \in \text{Top}_k(p_u),$$

where $d \in \text{Top}_k(p_u)$ if and only if:

$$|\{d' \in V_u : p_u(d') > p_u(d)\}| < k,$$

and ties are broken arbitrarily to ensure exactly k districts are included in $\text{Top}_k(p_u)$.

Thus, each user’s mobility graph represents a graph over M where the nodes corresponding to the user’s top- k most visited districts form a clique. This clique naturally captures the individual’s routine movement between these districts, such as travel between home and work. An example user mobility graph is shown in Figure 3a.

To protect user privacy, MIRAGE randomizes the edges of the user mobility graph. Each edge $(d_i, d_j) \in E_u$ is encoded as a binary value $e_{i,j} = 1$ (if there is an edge) or $e_{i,j} = 0$ (otherwise). The standard randomized response mechanism is then applied to each edge $e_{i,j} \in E_u$:

$$\tilde{e}_{i,j} = \begin{cases} e_{i,j} & \text{with probability } p \\ 1 - e_{i,j} & \text{with probability } 1 - p \end{cases}$$

The resulting graph is a *private user mobility graph* $\tilde{G}_u = (M, \tilde{E}_u)$. An example private user mobility graph is shown in Figure 3b.

Lemma 3 (Private User Mobility Graph is ϵ -DP). *The private user mobility graph satisfies ϵ local differential privacy where $\epsilon = (k^2 - k) \cdot \ln(p/(1-p))$.*

Proof. The mechanism is the unary encoding (UE) scheme applied to the edges of the user’s mobility graph. The UE scheme achieves ϵ -DP for $\epsilon = 2k \ln(p/(1-p))$ when applied to a k -hot vector of arbitrary length [27]. We represent each user’s mobility graph as a vector over E_u . By definition, E_u has $k \cdot (k-1)/2$ edges, and thus the UE mechanism is applied to $(k \cdot (k-1)/2)$ -hot vector and the claim follows. \square

The private user mobility graph $\tilde{G}_u = (V_u, \tilde{E}_u)$ encodes the user’s noisy behavior in terms of transitioning between their most visited districts. This provides two key guarantees. First, it preserves the privacy of users’ most-visited districts while enabling plausible deniability through the randomized inclusion of additional transitions. Second, it captures movement patterns that can be aggregated into a global mobility graph (§VI-B) that is used to inform routing decisions. In particular, $\hat{e}_{i,j} = (\tilde{e}_{i,j} - 1 + p)/(2p - 1)$ is an unbiased estimator of a user’s true reported edge $e_{i,j}$. In the next section (§VI-B), we

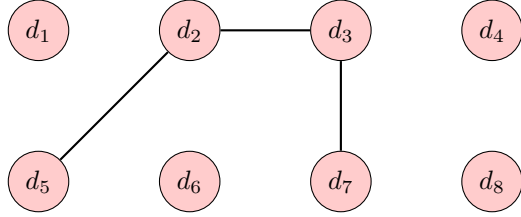
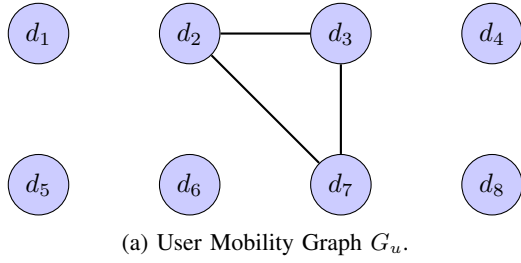


Fig. 3: Example of (a) user’s mobility graph and (b) a corresponding private mobility graph. Each edge of the non-private user mobility graph are perturbed using the randomized response mechanism.

discuss how the private mobility graphs are aggregated into a *global mobility graph* that captures the high-level mobility patterns of the population.

The parameters p and k govern the balance between graph fidelity and privacy. When p is close to 1, the user mobility graph is preserved with high fidelity, but privacy risks increase. When p is closer to 0.5, added noise in the transition edges provides stronger privacy protection. Similarly, as k grows toward $|M|/2$, user mobility graphs become more uniquely identifiable since users are more likely to have a distinctive set of top k visited districts.

B. Global Mobility Graph

MIRAGE leverages a *global mobility graph* to encapsulate the broader mobility trends of the general population. However, MIRAGE ensures that the collection and publication of this graph are conducted in a privacy-preserving manner, minimizing the need for trust. MIRAGE collects the private user mobility graphs (§VI-A) from a sample of users³ and generates a weighted transition graph that encodes the conditional probability of an individual in a source district s subsequently traveling to a destination district d . Leveraging this information, individuals can opportunistically forward messages at a rate proportional to the likelihood that a neighbor will later deliver the message to the target region of interest.

Definition 4 (Global Mobility Graph). *Let M denote the set of all districts, and let t represent a time epoch. The Global Mobility Graph is a weighted undirected graph $G = (M, E, P_t)$, where:*

- M is the set of vertices, with each vertex $d \in M$ representing a district.
- E is the set of undirected edges, where an edge $(u, v) \in E$ indicates potential movement between districts u and v .
- $P_t : E \rightarrow [0, 1]$ is a weight function where $P_t(u, v)$ represents the probability that a randomly selected individual in district u travels to district v within time epoch t . By construction, we assume symmetry so that $P_t(u, v) = P_t(v, u)$.

The *Global Mobility Graph* encapsulates the likelihood of individual movements between districts within the specified time epoch t .

The choice of t sets the temporal granularity of the mobility model. In MIRAGE, we set $t = 1$ day to align with typical human movement patterns, ensuring that forwarding decisions reflect realistic daily mobility while maintaining timely message delivery.

Given the above global mobility graph, we can use a gossip-like protocol with a dynamic parameter p , where the probability of forwarding a message depends on the current individual’s location and the message’s destination. Specifically, if Alice is in district u and holds a message destined for district v , she forwards the message according to a gossip protocol with forwarding probability $p = P_t(u, v)$, which represents the likelihood that a random individual in u will travel to v . This dynamic adjustment optimizes network bandwidth by tailoring the forwarding rate to the delivery likelihood. In regions where the probability of finding a suitable carrier is low, messages are forwarded more aggressively to improve delivery chances, whereas in regions with a high probability of encountering a suitable carrier, messages are forwarded less frequently, reducing redundant transmissions.

However, the global mobility graph must be constructed in a privacy-preserving manner. While it may be acceptable to reveal aggregate, population-level insights about transition probabilities (e.g., discovering that with high probability, a random individual in district u will later travel to district v), the protocol must ensure that the specific mobility patterns of any given individual remain private. To achieve this, we use a differentially private algorithm for collecting and publishing statistics about the population’s mobility profiles.

Each user locally generates their private mobility graph as described in the previous section (§VI-A). The private mobility graphs are undirected and encode frequent co-travel between districts. These graphs are then sent to a centralized curator, which aggregates them into a directed global mobility graph $\tilde{G} = (M, E, P_t)$. For each directed edge $(d_i, d_j) \in E$, the weight $P_t(d_i, d_j)$ represents the estimated probability that a randomly selected user in district d_i will subsequently travel to district d_j within time epoch t . Formally, this is computed as:

$$P_t(d_i, d_j) = \frac{1}{|U|} \sum_{u \in U} \mathbb{I}((d_i, d_j) \in E_u)$$

³For instance, this graph could be built from users who voluntarily opt in to share their private mobility profiles.

where U is the set of users reporting their private mobility graphs to the aggregator, and $\mathbb{I}(\cdot)$ is the indicator function, which is equal to 1 if the edge (d_i, d_j) exists in the private graph G_u , and 0 otherwise.

The global mobility graph \bar{G} is therefore a complete directed graph, where edge weights reflect the proportion of users whose private mobility graphs indicate transitions between pairs of districts.⁴ Since \bar{G} is derived from locally DP private graphs, by post-processing (Lemma 1), \bar{G} is also differentially private.

We analytically determine the number of users required for MIRAGE to generate an accurate global mobility graph, given a specified privacy level. Let G denote the true global mobility graph, constructed without privacy constraints (i.e., direct aggregation without noise). Consider a single edge $e_{i,j} \in G$ with true frequency $\mu(e_{i,j})$ and private frequency estimate $\hat{\mu}(e_{i,j})$ in \hat{G} , obtained by summing over the per-user unbiased estimates $\hat{e}_{i,j} = (\tilde{e}_{i,j} - 1 + p)/(2p - 1)$. Each $\hat{\mu}(e_{i,j})$ is therefore an unbiased estimator of $\mu(e_{i,j})$. Using the standard analysis of randomized response and union bounding over all edges, we ensure that, with probability at least $1 - \beta$, every edge simultaneously satisfies $|\mu(e_{i,j}) - \hat{\mu}(e_{i,j})| \leq \alpha$ provided that the number of users n satisfies

$$n = \Omega\left(\frac{\ln(|M|/\beta)}{\alpha^2 \varepsilon^2}\right)$$

In Figure 4, we show the number of users required to achieve a specified accuracy guarantee for a global mobility graph with $|M| = 100$ districts under different privacy levels ε . Computing an accurate global mobility graph, while maintaining meaningful privacy protections ($\varepsilon < 1$), is feasible given a sufficiently large number of users who opt in to sharing their private mobility graphs with the central curator. Relaxing the per-edge accuracy requirements reduces the number of users necessary, albeit potentially at the cost of diminished routing effectiveness.

Finally, we remark that requiring a subset of users to collect and send their private mobility profiles to a centralized curator may initially seem counter to the privacy-preserving goals of MIRAGE. However, many private mesh network messengers, such as Briar [7] and Moby [22], operate as hybrid systems, leveraging traditional network infrastructure during normal conditions while utilizing mesh networking capabilities during blackouts or periods of severe censorship. Similarly, a system utilizing MIRAGE could rely on traditional network infrastructure during a setup phase, in which the curator aggregates the contributed private mobility graphs into a global mobility graph and distributes it to participants. Once this setup is complete, the system can pivot to the mesh routing capabilities of MIRAGE, and users need only download the global mobility graph once before operating in a fully decentralized mode.

⁴Although the edge weights are symmetric and \bar{G} can be viewed as an undirected graph, we retain the directed formulation for conceptual clarity, since $P_t(u, v)$ is naturally interpreted as the probability that a randomly selected user in district u will later travel to v .

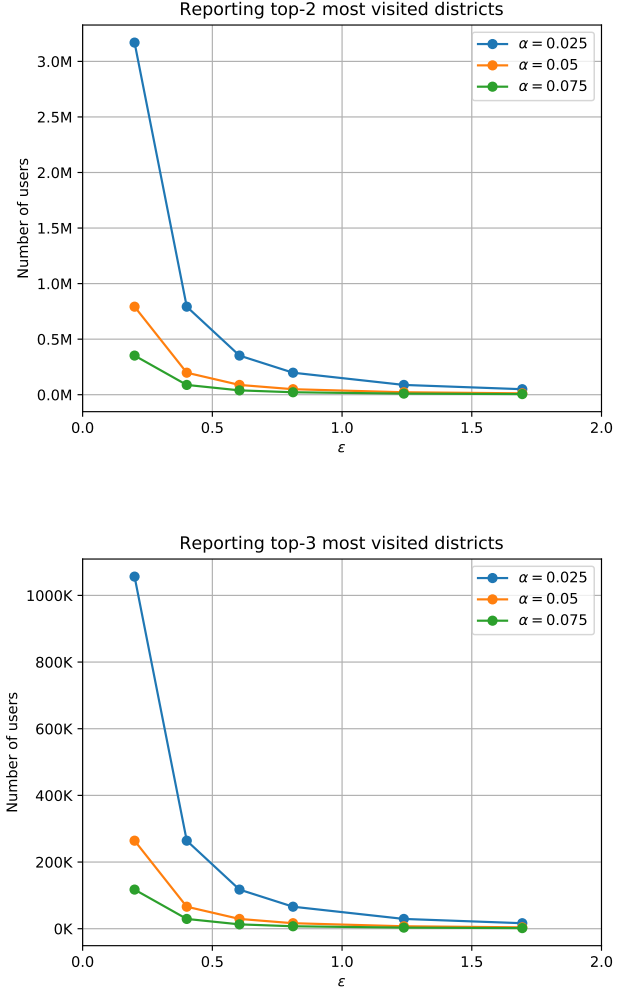


Fig. 4: Number of users required to estimate the true edge weight in the global mobility graph within additive error α with probability at least 99% for a given ε . Both analyses assume a map with 100 districts.

C. Routing

Each user will attempt to route messages based on the global mobility graph (§VI-B) and their private mobility profile (§VI-A). When a user Alice has a message destined to some district $d \in M$, she will attempt to exchange it with multiple message carriers. Upon coming in contact with another user Bob, Alice broadcasts the message, and Bob responds with `accept` or `reject` indicating whether or not he will attempt to route the message. Bob will accept the message according to the output of the decision function f defined below:

$$f((\text{alice}, \text{bob}), m) = \begin{cases} 1 & \text{if } \tilde{e}_{i,j} = 1 \\ 0 & \text{otherwise} \end{cases}$$

where $\tilde{e}_{i,j} \in \tilde{G}_{\text{bob}}$ is the edge in Bob's private mobility

graph, i is the district where Alice and Bob are currently encountering one another, and j is the destination district for message m . Since \tilde{G}_{bob} is differentially private and f is a post-processing function, it follows that the decision function f is also differentially private (Lemma 1).

Theorem 1. *Let $f : \mathcal{U}^2 \times \mathcal{M} \rightarrow \{0, 1\}$ be a routing decision function that for any users $u_1, u_2 \in \mathcal{U}$ who encounter each other in district i , and any message $m \in \mathcal{M}$ destined to district j ,*

$$f((u_1, u_2), m) = \begin{cases} 1 & \text{if } \tilde{e}_{i,j} = 1 \\ 0 & \text{otherwise} \end{cases}$$

where $\tilde{e}_{i,j} \in \tilde{G}_{u_2}$ is an edge in u_2 's private mobility graph \tilde{G}_{u_2} . Then f is an ε -differentially private routing decision function.

Proof. The proof follows from the fact that the outcome of routing decision function depends only on user u_2 's ε -DP private mobility graph \tilde{G}_{u_2} . By post-processing (Theorem 1), f is also ε -DP. \square

While other decision functions are possible, it is essential that f not depend on any internal state⁵ that could be correlated with Bob's mobility profile, such as the number of messages currently in his queue.

Under this decision function, users accept messages when a directed edge exists between d_i and d_j in their private mobility graph. Notably, when Bob responds to Alice with `accept`, one of two situations must hold:

- 1) **True Positive:** Bob's actual (non-private) mobility graph indicates that he frequents district d_j .
- 2) **False Positive:** Bob's actual (non-private) mobility profile indicates that he does *not* frequent district d_j , but his perturbed private mobility graph reports that he does.

Alice can anticipate that case (2) occurs with probability at most $(1 - p)$, where p is the probability of preserving the true value in the randomized response. Consequently, to maximize the likelihood that her message reaches district d_j , Alice must compensate for the noise introduced by differential privacy by forwarding her message to multiple carriers. Given that the global mobility graph (§VI-B) encodes the probability $P_t(d_i, d_j)$ that a randomly selected user in district d_i will later travel to district d_j , Alice forwards her message to approximately $\min\{n, n_{\max}\}$ users where n_{\max} is a fixed constant and

$$n = \begin{cases} \left\lceil \frac{p \cdot P_t(d_i, d_j) + (1-p) \cdot (1 - P_t(d_i, d_j))}{p \cdot P_t(d_i, d_j)} \right\rceil, & \text{if } P_t(d_i, d_j) > 0, \\ n_{\max}, & \text{otherwise.} \end{cases}$$

The constant n_{\max} limits the number of transfers to prevent excessive message propagation. The expression inside the ceiling operator follows directly from Bayes' Theorem and

⁵For instance, if Bob resides in a densely populated area, he may encounter more individuals, potentially leading to a larger number of messages in his local queue. If f utilizes this information, differential privacy is not guaranteed to hold.

equals the inverse of the conditional probability that a user's *true* mobility graph contains the edge from d_i to d_j , given that their *private* (noisy) mobility graph reports that edge. This value represents the expected number of recipients required so that, on average, one will genuinely travel to d_j . The parameter p explicitly captures the noise introduced by randomized response, allowing the protocol to maintain delivery reliability despite potential false positives.

Memoryless TTLs. MIRAGE implements a time-to-live (TTL) mechanism to ensure messages do not circulate indefinitely within the network. However, explicitly embedding TTL values risks revealing proximity information: if Bob receives a message from Alice with a high TTL, he may infer Alice was recently near (or even is) the message originator.

To mitigate this privacy risk, MIRAGE adopts *memoryless TTLs*. Each carrier independently attempts message delivery within discrete epochs of fixed duration. At the end of an epoch, if delivery has not occurred, the carrier discards the message with probability q . With probability $1 - q$, the carrier instead forwards the message to at most n_{\max} peers. The exact number of peers selected may vary according to the dynamic gossip parameters detailed in §VI-C. Subsequent carriers repeat this probabilistic retention and forwarding procedure independently.

We analyze this forwarding strategy as a classical branching process, characterized by the replication factor $R \leq n_{\max} \cdot (1 - q)$. To guarantee eventual message extinction, the process must be subcritical, thus requiring $R < 1$ or equivalently $q > 1 - 1/n_{\max}$. Under this constraint, the expected number of carriers decays exponentially with the number of epochs. That is, the number of carriers of a given message at epoch j is given by $\mathbb{E}[X_j] \leq n_{\max} \cdot R^j$. By setting $\mathbb{E}[X_j] = 1$, we find the expected message lifetime in epochs is approximately $-\ln(n_{\max})/\ln(R)$. Thus, the system parameters (n_{\max}, q) can be tuned explicitly to ensure predictable message extinction.

Message Delivery. When a message carrier arrives at district d and holds a message m addressed to that district, they initiate an epidemic-style broadcast of the message throughout the district. Nearby nodes that receive the broadcast will rebroadcast the message, continuing to propagate it to other nodes within the district. This process is repeated for a fixed duration (e.g., one day) to maximize the likelihood that the intended recipients within the district eventually receive the message.

VII. SIMULATION STUDY

To evaluate MIRAGE's performance, we utilize Cadence [5], an open-source discrete event human movement simulator that runs routing algorithms on top of real-world human movement datasets. Cadence simulates message exchanges among nodes, where each node's movements in a virtualized space is governed by a real-world trace of a human's movements. Replaying those traces using different routing algorithms allows us to make fair comparisons between routing protocols.

We enhanced Cadence by adding support for MIRAGE and Aviv et al.’s probabilistic profile-based routing (PPBR) [2]. Our implementation is available at <https://doi.org/10.5281/zenodo.16953762>.

Cadence takes as input a human movement dataset that consists of a set of *events* $\mathcal{E} = \{e\}$, where each e is a tuple $\langle n, t, l \rangle$, with n being a node identifier, t a timestamp, and l a location. Cadence considers an *encounter* to occur between two events e_i and e_j if (1) $n_i \neq n_j$, (2) $t_i = t_j$, and (3) $\|l_i - l_j\|_2 \leq \tau_l$, where $\|\cdot\|_2$ denotes the L^2 distance and τ_l is tunable distance threshold.⁶ Message transfers can occur only during encounters. Conceptually, encounters capture the notion of two nodes being in close proximity (i.e., having a distance apart no greater than τ_l) at a moment in time, and represent an opportunity for message transfers. Our choice for τ_l (see Table I) is loosely informed by the ranges of WiFi Direct and BLE.

A. Simulation Setup

Datasets. To evaluate MIRAGE, we consider two human movement datasets:

- **YJMob100K** [30] is an anonymized human mobility dataset that describes the movements of individuals in a city in Japan over 75 days, the last 15 of which occurred during an unspecified emergency. Locations were collected using mobile phone location data.
- **T-Drive** [31, 32, 33] is a collection of taxicab trajectories recorded in Beijing, China, in 2008.

We simulate the first 500 users from each dataset to meet Cadence’s performance limits. Summary statistics for the two datasets are given in Table I.

We chose these particular datasets as they capture large city-scale areas similar to where we imagine MIRAGE might be deployed. These two contrasting datasets—pedestrian movements and taxi trajectories—capture diverse scales and modalities of human mobility. Despite their differences, both exhibit high frequencies of close-proximity interactions, making them well-suited for studying contact-driven protocols. Pedestrian data offers insight into dense, human-scale dynamics, while taxi data reflects structured, vehicular mobility in a car-centric city. Together, they allow us to evaluate MIRAGE’s generalizability, ability to preserve privacy, and behavioral invariants across movement modalities.

MIRAGE Configuration. To define districts for the YJMob100K and T-Drive datasets, we apply k -means clustering to the initial 10% of each user’s location events, ordered chronologically. We set the number of clusters to be 100, which we posit is a reasonable choice for defining areas of

a city. Our set of districts is therefore defined as the set of cluster centers.⁷ Districts are generated once for each dataset. Cadence maps a location to a district by assigning it to the district with the closest centroid.

We use the 2nd 10% of each user’s chronologically ordered location events from each dataset to construct a representative global mobility graph. To simulate possible error in the global mobility graph (e.g., sampling error, or a group of malicious participants purposefully contributing bogus location data), we compute the global mobility graph from the population exactly, and add Gaussian noise with standard deviation $\sigma = 0.01$ and mean $\mu = 0$ to each edge.

Cadence simulates MIRAGE using the remaining 80% of the chronologically ordered location data in each dataset. Given the relatively few number of users (500) in the tested datasets, we use a replication factor of three and allow a node to forward its message to $\min\{3n, n_{\max}\}$ peers (see §VI-C).

Comparison of Routing Protocols. As baseline comparisons, we consider *maximal flooding*, *handoff*, *probabilistic flooding*, and *probabilistic profile-based routing (PPBR)* [2] protocols. In maximal flooding, nodes transmit all stored messages to every node that they encounter. At the other extreme, in handoff, a message-carrying node passes all of its stored messages to the first node that it encounters, and then deletes all local copies. To prevent messages from vacillating between two nearby nodes, each node remembers the message identifiers of messages it has seen, and does not accept messages it has already received. Maximal flooding and handoff are both deterministic (non-randomized) protocols.

Our implementation of probabilistic flooding uses respective message passing and deletion probabilities⁸ of $1/2$ and $4/5$, meaning that a node will attempt to share each message that it carries with an encountered node with probability $1/2$, and when such a transfer occurs, it will delete its local copy of the message with probability $4/5$. If a node does not delete its local copy, the message remains in its message queue and may be transferred again when it next encounters another node.

We also instrument Cadence with probabilistic profile-based routing (PPBR) [2]. In PPBR, each node uses its privately-computed location profile to locally assess whether it is a good candidate to forward a message without revealing this decision to others (though, this decision can be implicitly revealed by later message broadcasts as discussed in §V). To limit message duplication and enhance scalability, each carrier announces the message to only k_{PPBR} encountered nodes, of which only one should probabilistically accept the message and carry it away. This balances delivery reliability with privacy and efficiency, avoiding the overhead of epidemic maximal flooding

⁶Since human movement datasets are often sparse and do not contain locations for all nodes for every reported timestamp, Cadence infers the position of a node at any time between two reported events using a linear movement model: given two consecutive events $e_a = \langle n, t_a, l_a \rangle$ and $e_b = \langle n, t_b, l_b \rangle$ of a node n , with $t_a < t_b$, the inferred position l_α of an event α at some intermediate time $t_\alpha \in (t_a, t_b)$ is defined as $l_\alpha = l_a + \frac{l_b - l_a}{t_b - t_a} (t_\alpha - t_a)$. Cadence infers the locations of all nodes for every time in the dataset and computes the encounters between nodes.

⁷In practice, districts should be defined independently of user mobility data—ideally using public sources—to avoid privacy risks, as clustering itself can reveal sensitive patterns if not done with differential privacy. While we used clustering to identify city centers and define districts within our datasets, a similar outcome could likely be achieved with minimal public information (e.g., known city centers, ZIP codes, metro or bus stops).

⁸We chose these probabilities to achieve reasonable message dissemination while avoiding mirroring the epidemic flooding of maximal flooding.

TABLE I: Human movement datasets. We consider the first 500 nodes in each dataset.

Dataset	Modality	Median Encounters	τ_l
YJMob100K [30]	Personal	117	50 m
T-Drive [33]	Vehicle	136	50 m

while preserving user control through localized, profile-based decisions. However, as shown in §V, PPBR is vulnerable to statistical disclosure attacks, and as we demonstrate below, can lead to epidemic maximal flooding in certain cases.

Unlike MIRAGE’s district-based addressing scheme, PPBR adopts a grid-based routing system. Consistent with prior work, we configure the grid overlay to consist of $200\text{m} \times 200\text{m}$ grid squares, which Aviv et al. suggest approximates the size of a city block [2]. Following Aviv et al., we set $k_{\text{PPBR}} = 10$.

For all routing algorithms, we configure Cadence with a message queue size of 500, meaning each node can store up to 500 message copies. When the queue is full, Cadence evicts messages using a FIFO strategy.

Metrics. To compare the efficacy and efficiency of the different routing algorithms, we consider the following metrics:

- **Delivery rate:** the fraction of sent messages that reach their intended destination during the simulation;
- **Network load:** the total number of messages transfers that occur during a simulation;
- **Delivery efficiency:** delivery rate divided by network load; and
- **Message latency:** the time between when a sender sends a message and when that message is received by its intended destination.

At one extreme, we note that maximal flooding provides high delivery rates and low message latencies, but does so at the cost of high network load. At the other extreme, the handoff protocol guarantees at most one copy of a given message exists in the network at any time, and thus has optimal network load; but, as shown in §VII-B, it produces poor delivery rates.

Workload. We configure Cadence such that each node originates 10 messages to 10 other nodes, selected uniformly at random without replacement. This constrains the total number of messages in the network to be between 5000 (there exists a single copy of each message; recall that there are 500 nodes who each send 10 messages) and 5000^2 (each node has a copy of every message). The origination time of each message is chosen uniformly at random from the range defined by the first time the message’s originator appears in the dataset to the last time it appears. We consider a message to be delivered if and only if it is received by the intended receiver.

Although MIRAGE uses a district-based addressing scheme, for fair comparisons with our baseline protocols, we consider a MIRAGE message delivered only if it is received by the message’s intended receiver. For MIRAGE, we assume that the sender (1) has *a priori* knowledge of the intended receiver’s most frequented district, and (2) uses that district as the message’s targeted district.

B. Simulation Results

We fix $k = 2$ based on the premise that users typically only commute between a small set of frequented locations (e.g., home and work). We achieve similar results for $k \in \{4, 6\}$ shown in Appendix A.

Delivery Rates. The median delivery rates achieved by the five routing protocols when using the YJMob100K and T-Drive datasets are respectively presented in Figures 5a and 6a. For MIRAGE, we vary p and show the resulting value of ε along the x-axis. We note that ε is dependent on both p and k , and is computed as $\varepsilon = k(k-1)\ln(p/(1-p))$. In all experiments, we perform 10 runs of MIRAGE, probabilistic flooding, and PPBR. Shaded regions show the inter-quartile ranges (IQRs); in some cases, these can be difficult to perceive due to small IQRs. The maximal flooding and handoff protocols are deterministic and have no IQR.

For the pedestrian-based YJMob100K dataset, MIRAGE significantly outperforms probabilistic flooding and handoff. For example, when $p = 0.6$, MIRAGE’s delivery rate (36.9%) was $4.1\times$ and $11.5\times$ that of probabilistic flooding (9.1%) and handoff (3.2%), respectively. The delivery rates for MIRAGE and PPBR are comparable, with MIRAGE slightly outperforming PPBR for $p = 0.55$, but moderately underperforming when $p \in \{0.6, 0.65\}$. For the vehicle-based T-Drive dataset, MIRAGE’s delivery rate generally matches that of the probabilistic flooding protocol (e.g., 25.5% vs. 25.9% when $p = 0.6$). PPBR had a much higher delivery rate for this dataset (67.8%), but this is due to PPBR’s inability to avoid flooding; PPBR effectively functions as maximal flooding (see below). The handoff strategy again exhibited poor performance with a delivery rate of only 3.4%.

For both datasets, MIRAGE’s delivery rate decreases with an increasing p . This is expected, since as p approaches 0.5, the probability that any edge $(d_i, d_j) \in V_u \times V_u$ exists in a user u ’s mobility graph G_u also approaches 0.5. In such cases, MIRAGE essentially becomes probabilistic flooding.

Network Load. Although MIRAGE did not achieve the delivery rate of maximal flooding (86.9% for YJMob100K and 68.9% for T-Drive), maximal flooding results in a large network load, as is observable in Figures 5b and 6b. In contrast, the handoff protocol has optimum network load—with only one copy per message—but has an untenable delivery rate. MIRAGE incurs a significantly decreased network load compared to maximal flooding, especially for higher values of p . As discussed above, network load decreases with p since values of p closer to 0.5 result in flood-like behavior.

For the YJMob100K dataset, MIRAGE and PPBR exhibit similar network loads (e.g., 838k vs. 769k, respectively, when $p = 0.6$). However, when run against the T-Drive dataset, PPBR behaves similarly to maximal flooding. In PPBR, a carrier of a message transmits the message to k_{PPBR} peers. A peer accepts a communicated message if it believes it is best suited among than the other $k_{\text{PPBR}} - 1$ recipients to deliver the message to its intended destination. This determination is heuristic and is based on the node’s previous movement

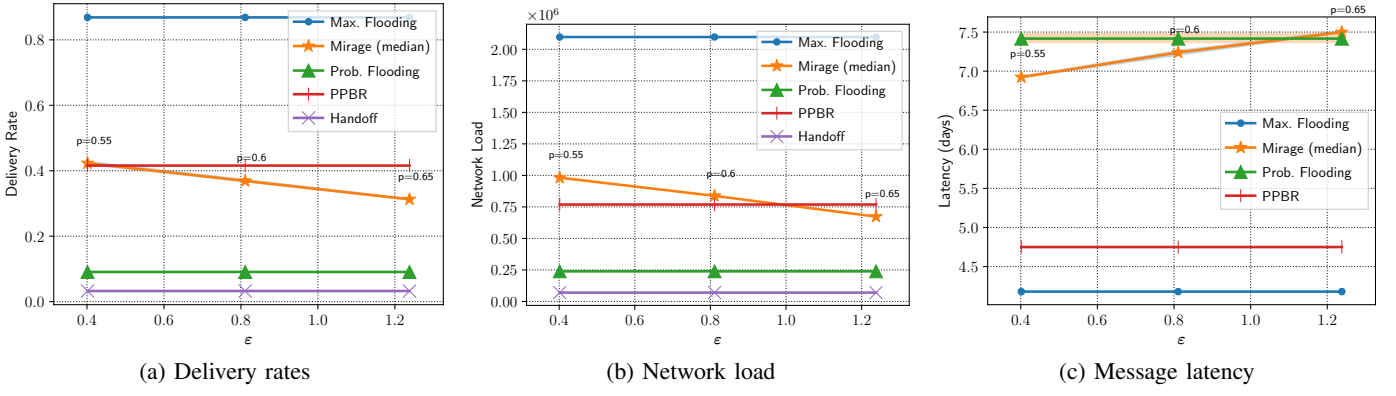


Fig. 5: Performance of MIRAGE and other routing protocols on the YJMob100K dataset, with $k = 2$. Shaded regions show the IQRs for MIRAGE and probabilistic flooding over 10 runs.

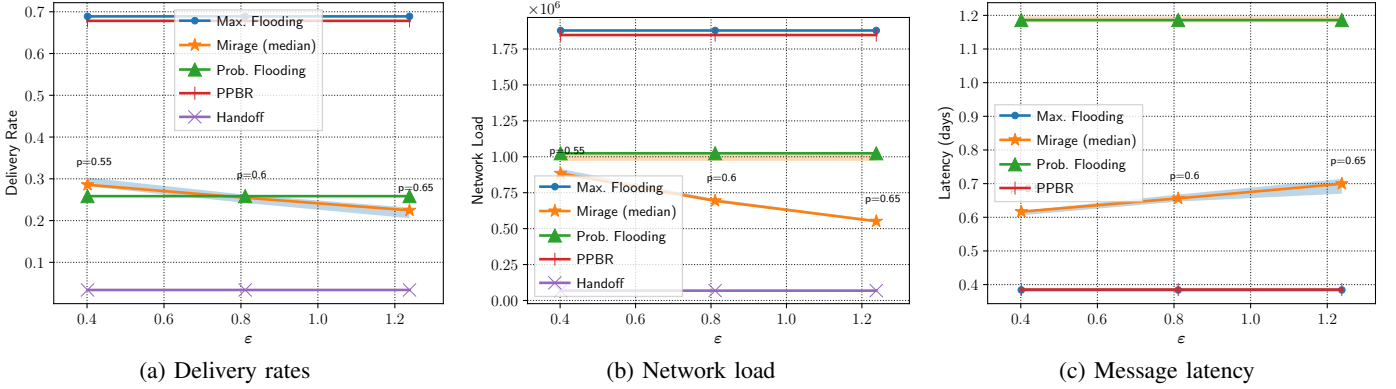


Fig. 6: Performance of MIRAGE and other routing protocols on the T-Drive dataset, with $k = 2$. Shaded regions show the IQRs for MIRAGE and probabilistic flooding over 10 runs.

patterns and its estimation of the uniqueness of its likelihood of visiting the destination location specified in the message. In the case of the T-Drive dataset, PPBR nodes overestimate their suitability for delivering messages, resulting in epidemic-like behavior. In contrast, as described in §VI-C, message acceptance is more limited in MIRAGE, where a node accepts a message only if there is a corresponding edge in its private mobility graph.

Delivery Efficiency. In Appendix B, we use *delivery efficiency* to examine MIRAGE’s utility as a function of privacy (ϵ). Delivery efficiency captures both delivery rate and network load, and constitutes an intuitive measure of MIRAGE’s routing efficacy. In brief, we confirm that decreased privacy (increased ϵ) leads to reduced network load (see also Figures 5b and 6b) and overall more efficient routing.

Message Latency. Figures 5c and 6c show the median message latencies for the two datasets, considering only the delivered messages. We omit the handoff protocol since its inclusion would be misleading given its very low ($\leq 3.4\%$) delivery rate. Maximal flooding achieves the lowest message latency, but as shown above, does so while imposing a large network load. We find that MIRAGE provides faster message delivery than probabilistic flooding, which we attribute to MIRAGE’s more targeted message passing strategy. For values of p closer to 0.5, this “targeting” effect diminishes as

users’ private mobility graphs become more noisy, moving MIRAGE’s latency closer to that of flooding. PPBR delivers messages faster than MIRAGE (especially for the T-Drive dataset when its behavior matches that of maximal flooding).

Summary of simulation study: Maximal flooding offers high delivery rates and low latencies, but is unscalable due to high network load. In contrast, MIRAGE’s use of (noised) mobility-patterns enables *more efficient routing* and lower network loads. MIRAGE delivered significantly more ($4\times$) messages than probabilistic flooding when using the pedestrian dataset, and had comparable delivery performance on the vehicular dataset. For both datasets, MIRAGE was able to deliver messages faster than probabilistic flooding. For the pedestrian movement dataset, the routing performance of MIRAGE and PPBR were similar, although PPBR exhibited lower latency. However, PPBR’s propensity to revert to flooding was evident when using vehicular-based movement data, resulting in far greater network loads than MIRAGE.

VIII. RELATED WORK

Aviv, Sherr, Blaze, and Smith introduced *probabilistic profile-based routing* (PPBR) [2], a mobility-aware message routing scheme based on population-level mobility patterns. In PPBR, participants are provided with a general mobility profile that estimates the average movement behavior of the population across a given geographic region. Each user then computes a *marginal similarity score* for a destination district $d \in M$, quantifying their likelihood of visiting d relative to the average person. A higher marginal similarity score indicates a better suitability as a message carrier for that district. Users self-select as message carriers by silently accepting or rejecting messages based on a locally chosen threshold τ , calibrated so that each user accepts messages for roughly a $1/k$ fraction of possible destinations.

Although PPBR reduces direct coordination between sender and receiver, later forwarding behavior still leaks sensitive information. As discussed in §V, once a user forwards a message, an observer can infer that the user either originated the message or accepted it because their marginal similarity score exceeded the threshold. This lack of plausible deniability leaves PPBR vulnerable to statistical disclosure attacks, enabling adversaries to recover both past and future mobility patterns from observed forwarding behavior.

Beyond PPBR, several peer-to-peer messaging systems have been proposed for censorship-resistant communication. Briar [7] is a decentralized application that operates without centralized servers, using Bluetooth, Wi-Fi, and Tor to support messaging during Internet outages. While Briar provides end-to-end encryption and strong metadata protection, it is limited to communication between pre-established trusted contacts, requiring manual credential exchange [4]. This constraint reduces its applicability in spontaneous, large-scale mobility routing scenarios, where efficient forwarding among untrusted peers is essential.

Moby [22] takes a different approach, offering blackout-resistant secure messaging that combines Internet connectivity when available with ad-hoc peer-to-peer communication during outages. Moby ensures end-to-end encryption, forward secrecy, and sender-receiver anonymity, enhancing resilience against censorship and denial-of-service attacks. However, Moby’s routing protocol relies on trust-based epidemic dissemination, where trust is inferred from prior direct or mutual contacts. In contrast, MIRAGE does not require trust-based forwarding but instead employs differentially private mobility profiles to probabilistically optimize message routing while preserving privacy. This enables MIRAGE to function effectively in dynamic environments without requiring prior contact relationships, making it more suitable for large-scale, infrastructure-free communication in adversarial settings.

Finally, there has been considerable work that examines methods for evading Internet-based censorship (Khattak et al. provide a good, albeit dated, survey of the Internet evasion landscape [18]). Proposed evasion strategies generally apply mimicry [20, 26] (trying to appear as an allowed protocol),

tunneling (embedding a covert channel within a non-blocked cover protocol) [12, 16], refraction networking [17, 25, 29] (redirecting Internet traffic via on-path routers), and proxy-based approaches [6, 13, 19] (e.g., VPNs or Tor bridges). Mimicry is known to be easily detected by an adversary [15], and most deployed evasion systems are proxy-based, likely due to the relative ease of instantiating proxies—especially for browser-based proxies such as Snowflake [6]. However, all of these approaches rely on centralized Internet infrastructure that may be susceptible to subversion by state-level actors. At the extreme, nation states may choose to (and have chosen to [8]) shut down this infrastructure to curtail communication, especially during times of civil unrest. MIRAGE avoids existing infrastructure by instead using human-to-human links to route messages.

IX. DISCUSSION

MIRAGE Setup Phase. MIRAGE improves message routing efficiency by leveraging user mobility histories. This involves a setup phase in which users voluntarily share locally DP user mobility graphs with a central aggregator. The accuracy of MIRAGE’s global mobility graph depends on the size of this opt-in user subset, and configurations with more restrictive privacy budgets require larger participation rates to achieve comparable accuracy. Alternatively, if reliable public mobility data is available, the global mobility graph can be constructed without this setup phase. However, publicly available data may be inaccessible, unreliable, or subject to control by the governing authority.

Sybil attacks, involving adversaries creating a large number of fake user accounts, could attempt to degrade the accuracy of MIRAGE’s global mobility graph by introducing fabricated mobility profiles. Although such attacks could negatively impact the overall utility and reliability of routing decisions, they do not compromise individual privacy. Since each user determines their routing decisions based on locally DP mobility profiles, the presence of sybil-generated data in the global graph cannot leak information about genuine users’ individual mobility patterns.

Importantly, our analysis and evaluation assumes that users opting into the setup phase are drawn uniformly at random from the general population. However, in practice, this assumption might not hold. Certain groups of users—such as dissidents, activists, or politically sensitive individuals—may perceive higher risks associated with sharing mobility information, even under differential privacy guarantees, and thus might be less likely to participate. Conversely, users less concerned about surveillance or detection may disproportionately opt in, potentially biasing the resulting global mobility graph. Such biases could reduce the effectiveness of MIRAGE by impacting routing decisions in politically sensitive or high-risk areas.

Future work could explore eliminating the centralized setup phase altogether. For instance, private mobility graphs could be distributed among users through flooding, enabling each node to independently collect a sufficient subset of private mobility graphs to compute the global mobility graph locally.

However, such decentralized approaches introduce new privacy considerations, as each user’s locally computed global mobility graph would inherently depend on their individual mobility patterns and interactions. Nodes interacting with a limited subset of users could obtain biased or incomplete versions of the global graph, potentially reducing routing effectiveness and allowing an adversary to infer a user’s prior location history based on their version of the global mobility graph. Thus, careful consideration is required when designing fully decentralized mechanisms to ensure privacy and accuracy tradeoffs are maintained.

Another intriguing extension to MIRAGE involves allowing individual users to set personalized privacy budgets for their mobility graphs. This approach could accommodate varied privacy preferences based on individual risk assessments and exposure to surveillance. For example, users operating in environments with higher surveillance risks might prefer stronger privacy guarantees, accepting reduced accuracy in their reported mobility graphs. Conversely, users in less risky contexts might choose weaker privacy settings to contribute more precise data. Implementing personalized privacy budgets would enhance user autonomy but would also introduce complexities in aggregating heterogeneous data. Furthermore, even the act of selecting a privacy budget may leak sensitive information since choosing strong privacy settings could potentially identify individuals as belonging to higher-risk or targeted groups.

Handling changes in mobility patterns. MIRAGE protects user mobility patterns under local differential privacy by randomizing each user’s mobility graph. However, user mobility patterns are rarely completely static; individuals may relocate, change employment, or adopt new recreational routines. Such changes can lead to discrepancies between a user’s actual mobility behavior and their initially generated mobility graph. To address this, MIRAGE could be extended to dynamically detect shifts in individual mobility patterns locally, such as by continuously monitoring mobility data on the user’s device. Upon identifying significant divergence from the originally randomized mobility graph, MIRAGE would generate and randomize a new user mobility graph based on the updated mobility trends. Importantly, by composition (Theorem 2), each re-randomization results in an additional ϵ added to the total privacy loss. Therefore, after k updates, MIRAGE would still ensure $k\epsilon$ -DP.

ACKNOWLEDGMENTS

We thank the anonymous reviewers and our shepherd for their feedback and suggestions. This work is partially supported by the Georgetown University Callahan Family Chair Fund and the Farr Faculty Award. This material is partially based upon work supported by DARPA under Contract No. FA8650-22-C-6424. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

Zachary Ratliff’s work is supported in part by Cooperative Agreement CB20ADR0160001 with the Census Bureau, and in part by Salil Vadhan’s Simons Investigator Award.

REFERENCES

- [1] Al Jazeera. Poll results prompt iran protests. <https://www.aljazeera.com/news/2009/6/14/poll-results-prompt-iran-protests>, June 2009. [Online; accessed 23-December-2024].
- [2] Adam J Aviv, Matt Blaze, Micah Sherr, and Jonathan M Smith. Privacy-aware message exchanges for humanets. *Computer communications*, 48:30–43, 2014.
- [3] Adam J Aviv, Micah Sherr, Matt Blaze, and Jonathan M Smith. Evading cellular data monitoring with human movement networks. In *5th USENIX Workshop on Hot Topics in Security (HotSec 10)*, 2010.
- [4] Djohara Benyamina, Abdelhakim Hafid, and Michel Gendreau. Wireless mesh networks design — a survey. *IEEE Communications Surveys & Tutorials*, 14(2):299–310, 2012.
- [5] Harel Berger, Micah Sherr, and Adam Aviv. Cadence: A simulator for human movement-based communication protocols. In *Proceedings of the 16th Cyber Security Experimentation and Test Workshop*, pages 26–31, 2023.
- [6] Cecylia Bocovich, Arlo Breault, David Fifield, Serene, and Xiaokang Wang. Snowflake, a Censorship Circumvention System Using Temporary WebRTC Proxies. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 2635–2652, 2024.
- [7] Briar Project. Briar project: Secure messaging, anywhere. <https://briarproject.org/>.
- [8] Alberto Dainotti, Claudio Squarcella, Emile Aben, Kimberly C. Claffy, Marco Chiesa, Michele Russo, and Antonio Pescapé. Analysis of country-wide internet outages caused by censorship. In *ACM SIGCOMM Conference on Internet Measurement Conference (IMC)*, November 2011.
- [9] James A Davis, Andrew H Fagg, and Brian Neil Levine. Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks. In *Proceedings Fifth International Symposium on Wearable Computers*, pages 141–148. IEEE, 2001.
- [10] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The Second-Generation Onion Router. In *USENIX Security Symposium (USENIX)*, August 2004.
- [11] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pages 265–284. Springer, 2006.
- [12] David Fifield, Chang Lan, Rod Hynes, Percy Wegmann, and Vern Paxson. Blocking-resistant Communication through Domain Fronting. In *Privacy Enhancing Technologies Symposium (PETS)*, 2015.

- [13] Sergey Frolov, Jack Wampler, Sze Chuen Tan, J. Alex Halderman, Nikita Borisov, and Eric Wustrow. Conjure: Summoning Proxies from Unused Address Space. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 2215–2229, London United Kingdom, November 2019. ACM.
- [14] Matthias Grossglauser and Martin Vetterli. Locating mobile nodes with ease: learning efficient routes from encounter histories alone. *IEEE/ACM Transactions on Networking*, 14(3):457–469, 2006.
- [15] Amir Houmansadr, Chad Brubaker, and Vitaly Shmatikov. The Parrot is Dead: Observing Unobservable Network Communications. In *IEEE Symposium on Security and Privacy (Oakland)*, 2013.
- [16] Amir Houmansadr, Thomas Riedl, Nikita Borisov, and Andrew Singer. I Want My Voice to be Heard: IP over Voice-over-IP for Unobservable Censorship Circumvention. In *Network and Distributed System Security Symposium (NDSS)*, 2013.
- [17] Josh Karlin, Daniel Ellard, Alden W. Jackson, Christine E. Jones, Greg Lauer, David P. Mankins, and W. Timothy Strayer. Decoy Routing: Toward Unblockable Internet Communication. In *USENIX Workshop on Free and Open Communications on the Internet (FOCI)*, 2011.
- [18] Sheharbano Khattak, Tariq Elahi, Laurent Simon, Colleen M. Swanson, Steven J. Murdoch, and Ian Goldberg. SoK: Making Sense of Censorship Resistance Systems. *Proceedings on Privacy Enhancing Technologies (PoPETS)*, 2016(4):37–61, July 2016.
- [19] Patrick Tser Jern Kon, Aniket Gattani, Dhiraj Saharia, Tianyu Cao, Diogo Barradas, Ang Chen, Micah Sherr, and Benjamin E. Ujcich. NetShuffle: Circumventing Censorship with Shuffle Proxies at the Edge. In *IEEE Symposium on Security and Privacy (S&P)*, 2024.
- [20] Hooman Mohajeri Moghaddam, Baiyu Li, Mohammad Derakhshani, and Ian Goldberg. SkypeMorph: Protocol Obfuscation for Tor Bridges. In *ACM Conference on Computer and Communications Security (CCS)*, 2012.
- [21] New Scientist. Earthquake shakes the internet. <https://www.newscientist.com/article/mg19325852-300-earthquake-shakes-the-internet/>, January 2007. [Online; accessed 23-December-2024].
- [22] Amogh Pradeep, Hira Javaid, Ryan Williams, Antoine Rault, David Choffnes, Stevens Le Blond, and Bryan Alexander Ford. Moby: A blackout-resistant anonymity network for mobile devices. *Proceedings on Privacy Enhancing Technologies*, 2022(3):247–267, 2022.
- [23] Reuters. U.S. calls on Big Tech to help evade online censors in Russia, Iran. <https://www.reuters.com/technology/us-calls-big-tech-help-evade-online-censors-russia-iran-2024-09-05/>, September 2024. [Online; accessed 23-December-2024].
- [24] The Telegraph. Unprecedented cyber attack takes Liberia’s entire internet down. <http://www.telegraph.co.uk/technology/2016/11/04/unprecedented-cyber-attack-takes-liberias-entire-internet-down/>, November 2016. [Online; accessed 23-December-2024].
- [25] Benjamin VanderSloot, Sergey Frolov, Jack Wampler, Sze Chuen Tan, Irv Simpson, Michalis Kallitsis, J. Alex Halderman, Nikita Borisov, and Eric Wustrow. Running Refraction Networking for Real. *Proceedings on Privacy Enhancing Technologies*, 2020(4), August 2020.
- [26] Qiyang Wang, Xun Gong, Giang T. K. Nguyen, Amir Houmansadr, and Nikita Borisov. CensorSpoofer: Asymmetric Communication using IP Spoofing for Censorship-Resistant Web Browsing. In *ACM Conference on Computer and Communications Security (CCS)*, 2012.
- [27] Tianhao Wang, Jeremiah Blocki, Ninghui Li, and Somesh Jha. Locally differentially private protocols for frequency estimation. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 729–745, 2017.
- [28] Mingshi Wu, Jackson Sippe, Danesh Sivakumar, Jack Burg, Peter Anderson, Xiaokang Wang, Kevin Bock, Amir Houmansadr, Dave Levin, and Eric Wustrow. How the great firewall of china detects and blocks fully encrypted traffic. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 2653–2670, 2023.
- [29] Eric Wustrow, Scott Wolchok, Ian Goldberg, and J. Alex Halderman. Telex: Anticensorship in the Network Infrastructure. In *USENIX Security Symposium (USENIX)*, 2011.
- [30] Takahiro Yabe, Kota Tsubouchi, Toru Shimizu, Yoshihide Sekimoto, Kaoru Sezaki, Esteban Moro, and Alex Pentland. Yjmob100k: City-scale and longitudinal dataset of anonymized human mobility trajectories. *Scientific Data*, 11(397), April 2024.
- [31] Jing Yuan, Yu Zheng, Xing Xie, and Guangzhong Sun. Driving with knowledge from the physical world. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 316–324, 2011.
- [32] Jing Yuan, Yu Zheng, Chengyang Zhang, Wenlei Xie, Xing Xie, Guangzhong Sun, and Yan Huang. T-drive: driving directions based on taxi trajectories. In *Proceedings of the 18th SIGSPATIAL International conference on advances in geographic information systems*, pages 99–108, 2010.
- [33] Yu Zheng. T-Drive Trajectory Data Sample, August 2011. Available at <https://www.microsoft.com/en-us/research/publication/t-drive-trajectory-data-sample/>.

APPENDIX A

ADDITIONAL RESULTS FOR VARIOUS p PARAMETERS

Figures 7 and 8 show additional performance results for MIRAGE and other routing protocols running on the YJMob100K dataset when MIRAGE uses $k = 4$ and $k = 6$, respectively. The results are similar to that when $k = 2$ (see §VII-B), with delivery rates increasing slightly with greater values of

k . The same trend appears in Figures 9 and 10, which show performance results for $k = 4$ and $k = 6$ using the T-Drive dataset.

APPENDIX B DELIVERY EFFICIENCY

We examine MIRAGE’s utility as a function of p using a combined metric that considers both delivery rate and network load. In Figure 11, we plot *delivery efficiency* as a function of ε (and thus p), where delivery efficiency is defined as the delivery rate divided by the network load. Intuitively, delivery efficiency reflects how proficient MIRAGE is able to route messages: a higher delivery efficiency means messages are delivered at lower cost (i.e., network load).

With increasing p —and hence increasing ε —MIRAGE generally exhibits higher delivery efficiency.⁹ This is expected, as decreased privacy leads predictably to reduced network load and more efficient routing. We observe the same effect when k increases (not shown), since increasing k likewise causes an increase in ε .

APPENDIX C ARTIFACT APPENDIX

This appendix describes the steps required to reproduce the results of our experimentation—specifically, the findings of our simulation study (Section VII). Our evaluation of MIRAGE uses a modified version of Cadence [5], an open-source discrete-time event-based human-movement simulator. Specifically, we modified Cadence to support both MIRAGE and PPBR [2] routing, and added additional reporting and output modules to the simulator. The artifacts described in this appendix include the modified version of Cadence, the human mobility datasets used to perform our simulations, and Python scripts used to configure Cadence and execute simulations that mirror those of the simulation study described in Section VII of our paper.

We emphasize that many of the routing protocols described in this paper are probabilistic (non-deterministic), and hence results will not perfectly replicate those in the paper. However, because the variance of the protocols’ performance is small, following the artifact specifications and workflow described below should yield results that reproduce the scientific findings of our paper.

A. Description & Requirements

1) *Access*: The artifact package is available through Zenodo at <https://doi.org/10.5281/zenodo.16953762>. The project is actively being maintained, with future updates appearing in our GitHub repository at <https://github.com/GUsecLab/cadence>.

⁹An exception occurs when $p = 0.8$ for the T-Drive dataset. At such a high value of p , users’ local mobility graphs become very sparse, causing users to refuse to accept most messages. This appears to be the case here, as the network load dropped significantly from its level when $p = 0.75$.

2) *Hardware dependencies*: None. We note that we tested our modified version of Cadence using two hardware configurations:

- 128-core Linux server with AMD 7551 processors and 362GB of RAM running Linux 6.8.0 (Ubuntu 24.04.2 LTS); and
- 8-core Linux laptop with Intel i7-8550u processors and 16GB of RAM running Linux 6.8 (Ubuntu 24.04.2 LTS)

Cadence is designed to leverage parallelism, and its performance scales with the number of available cores.

3) *Software dependencies*: Our modified version of Cadence requires few software dependencies:

- Linux OS (tested on Ubuntu 24.04.2 LTS exclusively, but Cadence should be compatible with other platforms that support Golang)
- GoLang (version 1.24.4)
- sqlite3 (version 3.45.1)
- Python (version 3.12.3)

4) *Benchmarks*: To evaluate MIRAGE, we consider two human movement datasets:

- **YJMob100K** [30] is an anonymized human mobility dataset that describes the movements of individuals in a city in Japan over 75 days, the last 15 of which occurred during an unspecified emergency. Locations were collected using mobile phone location data.
- **T-Drive** [33] is a collection of taxicab trajectories recorded in Beijing, China, in 2008.

Our simulations consider the first 500 users from each dataset. We include the modified versions of these datasets in our artifact package.

B. Artifact Installation & Configuration

To install and configure our modified version of Cadence, first download the repository at <https://doi.org/10.5281/zenodo.16953762>. Follow the instructions in the `README.md` file to reproduce our simulation results.

C. Major Claims

MIRAGE is a privacy-preserving mobility-based messaging system designed for censorship-resistant communication. To prevent leakage of individual mobility behavior, MIRAGE protects users’ mobility patterns with local differential privacy, ensuring that participation in the network does not reveal an individual’s location history.

Our paper makes the following claims:

- (C1): MIRAGE’s routing outperforms both epidemic-style approaches and random-walk protocols by exploiting historical mobility data collected from the population.
- (C2): MIRAGE provides provable privacy guarantees which ensure that a user’s participation in MIRAGE does not reveal *too much* about their individual mobility patterns; we quantify a user’s privacy loss using differential privacy.

We show the veracity of Claim C2 through formal argument in Section IV of our paper. This artifact appendix focuses

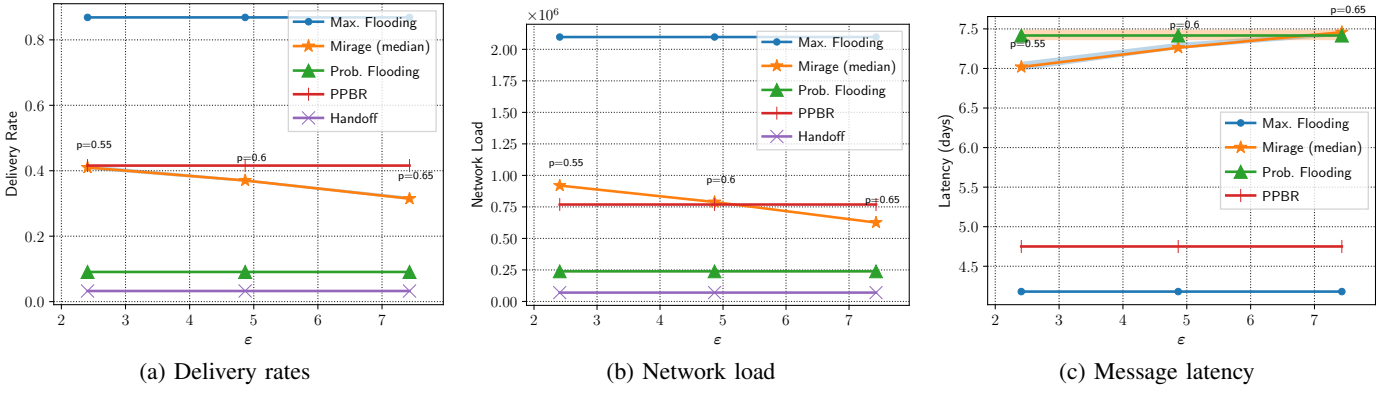


Fig. 7: Performance of MIRAGE and other routing protocols on the YJMob100K dataset, with $k = 4$. Shaded regions show the IQRs for MIRAGE and probabilistic flooding over 10 runs.

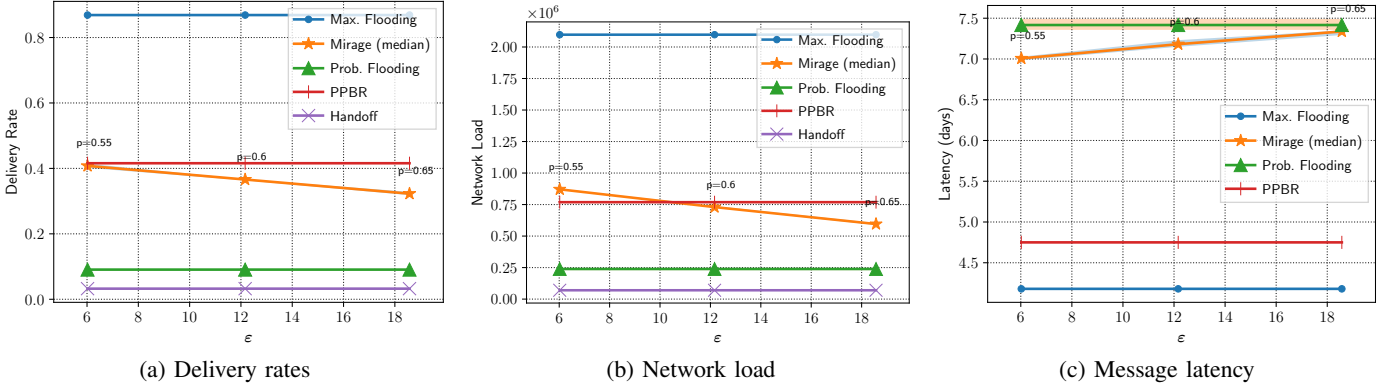


Fig. 8: Performance of MIRAGE and other routing protocols on the YJMob100K dataset, with $k = 6$. Shaded regions show the IQRs for MIRAGE and probabilistic flooding over 10 runs.

on the *performance* of MIRAGE relative to other routing protocols—that is, Claim C1.

D. Evaluation

We use the (modified) Cadence simulator to demonstrate the performance of MIRAGE routing. Our metrics of interest include the delivery rate (the fraction of sent messages that reach their destination); message latency (the time between the sending and reception of a received message); network load (the total number of message transfers); delivery efficiency (delivery rate divided by network load).

a) Overview: We use Cadence to simulate network message passing under five different routing protocols: MIRAGE, PPBR, Probabilistic Flooding, Handoff, and Maximal Flooding. Two networks created from the modified Japan and Tdrive human movement datasets are simulated.

The simulations take approximately ten hours of computation time to complete.

b) Preparation: For detailed experiment instructions, see the README.md document contained within the code repository. In brief, running the simulations and reproducing the results from our paper requires the following steps:

First, navigate to the code repository and set up a Python virtual environment.

```
python3 -m venv venv
```

Next, set up the required software and dependencies using the provided script:

```
python3 scripts/run_setup.py
```

Then, build the executable:

```
python3 scripts/run_build.py
```

Once all required modules are installed, create the necessary databases and import the included datasets using the provided script:

```
python3 scripts/run_import_parallel.py
```

c) Execution: Start the set of simulations by executing the provided script:

```
python3 scripts/run_experiments_parallel.py
```

This will run all trials for all routing algorithms for each network dataset, including all parameter combinations for MIRAGE. This encompasses the simulation results in Section VII of our paper.

Successful completion of experiments will generate two CSV files—japan.csv and tdrive.csv—stored in the results/raw-data directory; these correspond to our two human mobility datasets. The experiments will also generate auxiliary data.

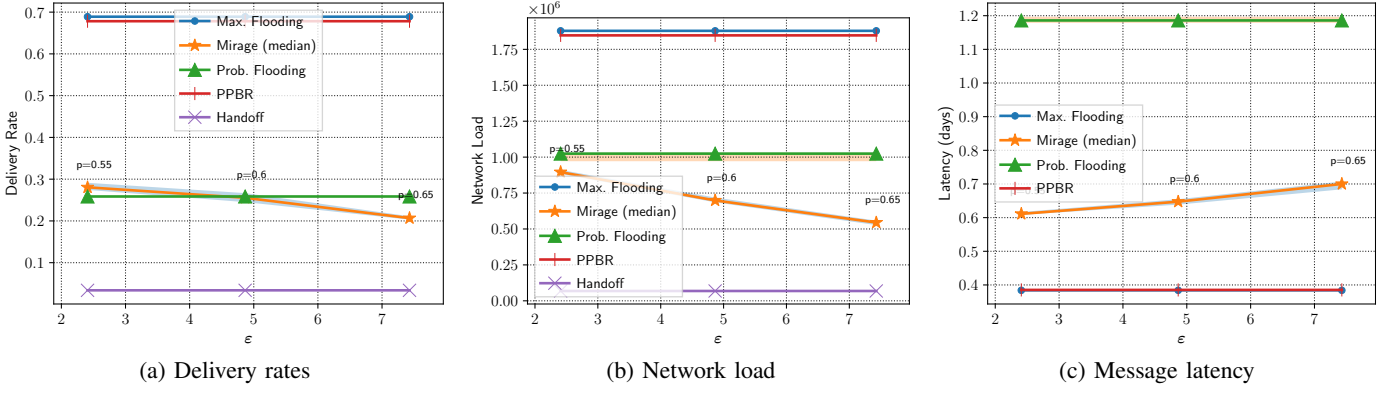


Fig. 9: Performance of MIRAGE and other routing protocols on the T-Drive dataset, with $k = 4$. Shaded regions show the IQRs for MIRAGE and probabilistic flooding over 10 runs.

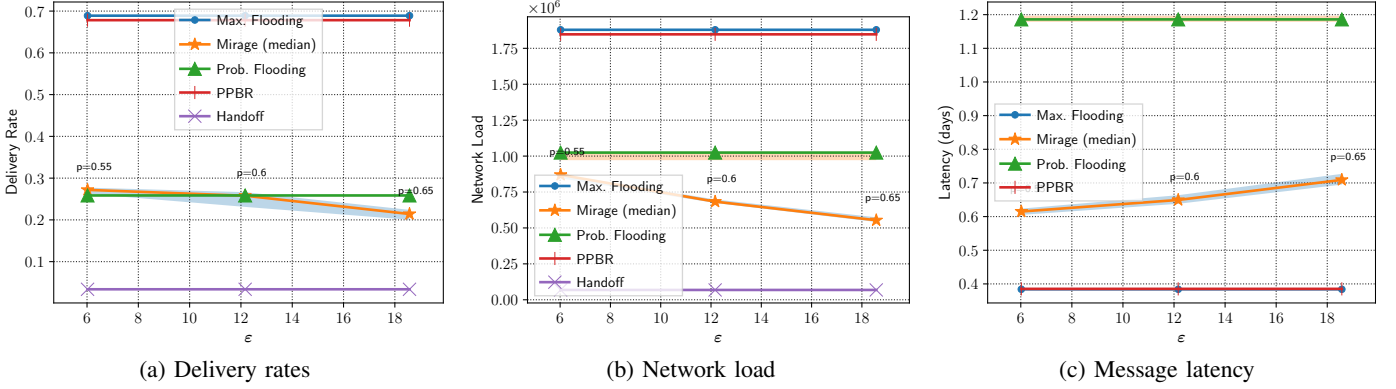


Fig. 10: Performance of MIRAGE and other routing protocols on the T-Drive dataset, with $k = 6$. Shaded regions show the IQRs for MIRAGE and probabilistic flooding over 10 runs.

d) Reproducing Plots: To convert the experiment result data files into the figures that appear in the paper, run the provided plotting script:

```
python3 scripts/run_plots.py
```

This will generate all plots and store them in the `results/plots` directory. The file `README.md` provides the mapping between each plot and its description.

The plots are named based on the convention `METRIC_K-VALUE_DATASET.pdf`, where `METRIC` is one of:

- `dr`: the delivery rate
- `lat`: the message delivery latency (in days)
- `nl`: the network load
- `custom` (message efficiency): a custom metric, defined as the delivery rate divided by the average load; this metric is not currently used in our paper
- `custom2` (delivery efficiency): a custom metric, defined as the delivery rate divided by the network load.
- `al`: the average load (average number of messages in the network per time frame); this metric is not currently used in our paper

The results from the simulation experiment confirm Claim C1. For certain combinations of p and k values, MIRAGE outperforms PPBR and probabilistic flooding in terms of load

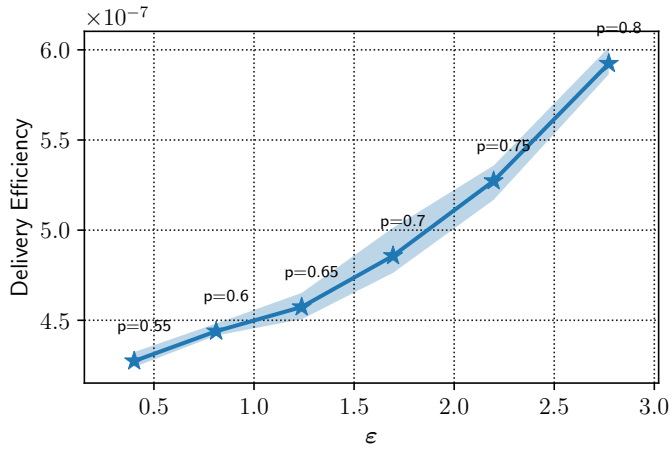
and delivery rate. On the modified Japan network, MIRAGE generated less network load than PPBR for $p = 0.65$. MIRAGE yielded higher delivery rate than PPBR for $p = 0.55$ while beating probabilistic flooding for all values of $p \in 0.55, 0.6, 0.65$. On the modified TDrive network, MIRAGE generated less network load than PPBR and probabilistic flooding for all values of p . MIRAGE yielded higher delivery rate than probabilistic flooding for $p = 0.55$. Since MIRAGE, PPBR, and probabilistic flooding protocols are probabilistic, specific results may vary by experiment.

E. Customization

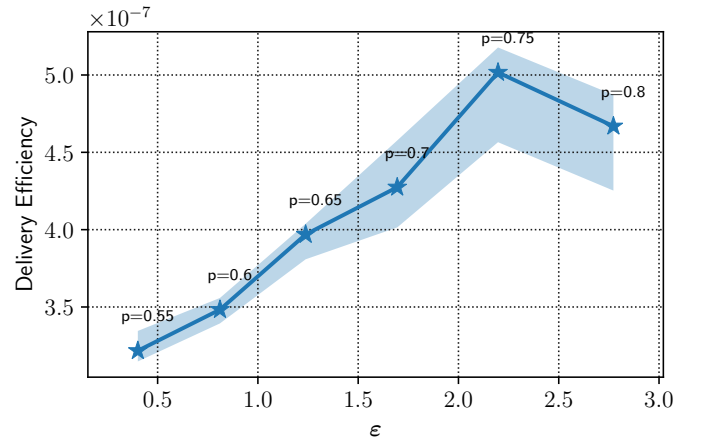
To run experiments for other values of p and k , modify `scripts/parameters.py`. Other configuration parameters can be modified via config files in `cmd/cadence/configs/NDSS` and `pkg/logic/logic_configs/NDSS`.

F. Notes

The top-level `README.md` file in the artifact repository contains more detailed, step-by-step instructions for reproducing the results of our paper.



(a) YJMob100K dataset



(b) T-Drive dataset

Fig. 11: Delivery efficiency for MIRAGE for various settings of p , with $k = 2$. Shaded regions show the IQRs over 10 runs.