A Formal Model of Obfuscation and Negotiation for Location Privacy

Matt Duckham[1] and Lars Kulik[2]

[1] Department of Geomatics,
University of Melbourne, Victoria, 3010, Australia
mduckham@unimelb.edu.au

[2] Department of Computer Science and Software Engineering,
University of Melbourne, Victoria, 3010, Australia
lkulik@cs.mu.oz.au

Abstract. Obfuscation concerns the practice of deliberately degrading the quality of information in some way, so as to protect the privacy of the individual to whom that information refers. In this paper, we argue that obfuscation is an important technique for protecting an individual's location privacy within a pervasive computing environment. The paper sets out a formal framework within which obfuscated location-based services are defined. This framework provides a computationally efficient mechanism for balancing an individual's need for high-quality information services against that individual's need for location privacy. Negotiation is used to ensure that a location-based service provider receives only the information it needs to know in order to provide a service of satisfactory quality. The results of this work have implications for numerous applications of mobile and location-aware systems, as they provide a new theoretical foundation for addressing the privacy concerns that are acknowledged to be retarding the widespread acceptance and use of location-based services.

1 Introduction

Privacy is internationally recognized as a fundamental human right [9]. Location-aware pervasive computing environments provide the ability to automatically sense, communicate, and process information about a person's location, with a high degree of spatial and temporal precision and accuracy. Location is an especially sensitive type of personal information, and so safeguarding an individual's location privacy has become an key issue for pervasive computing research.

This paper addresses the issue of protecting sensitive information about an individual user's location, at the same time as providing useful location-based services to that user. Our approach focuses on negotiating a balance in the levels of privacy and utility for a location-based service. Thus, in our model an individual may deliberately degrade the quality of information about his or her location in order to protect his or her privacy, a process called *obfuscation*. However, the quality of the location-based service an individual receives is directly linked to

H.W. Gellersen et al. (Eds.): PERVASIVE 2005, LNCS 3468, pp. 152–170, 2005.

the quality of information which that individual is willing to reveal about his or her location.

The primary contributions of this paper are the development of:

- a formal model of obfuscation and location privacy;
- an algorithm for efficient computation of a simple obfuscated location-based service; and
- a procedure for achieving a satisfactory balance of location privacy and location-based service quality through negotiation between an individual and a service provider.

Following a brief motivational example, section 2 situates this research within the context of the pervasive computing literature on location privacy. Section 3 introduces obfuscation, with particular reference to existing research related to this topic. Section 4 provides a precise description of the scenario addressed by this paper and the assumptions embedded within this scenario. Section 5 sets out the formal model of obfuscation in location-based services, and develops the mechanisms for obfuscating a simple location-based service. Section 6 concludes the paper with a summary of the results and an overview of current and future related research.

1.1 Motivational Example

In order to motivate this paper, consider the scenario illustrated in Figure 1.1.

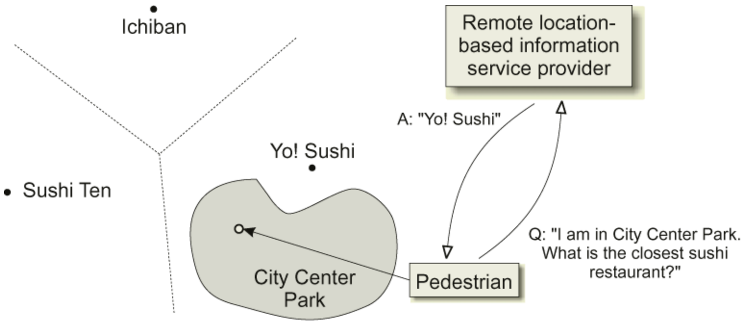
Ichiban
Sushi Ten
Yo! Sushi
City Center Park
Remote location-based information service provider
A: "Yo! Sushi"
Q: "I am in City Center Park. What is the closest sushi restaurant?"
Pedestrian

Fig. 1. Idealized example of an obfuscated location-based information service

A pedestrian wishes to access information about the address of the closest sushi restaurant, via a remote location-based service provider. Although there are three nearby restaurants, our hungry pedestrian would like to protect her privacy by providing only an approximate location to the information service provider. For example, the pedestrian can obfuscate her exact location by revealing only that she is in the "City Center Park." In this case, the service provider should still be able to correctly reply with the address of "Yo! Sushi," using basic spatial analysis algorithms (i.e., by constructing the Voronoi diagram for the sushi restaurants).

This example makes several simplifying assumptions, including a homogeneous Cartesian space with distance measured according to the Euclidean metric, and that the park does not overlap two proximal polygons. Nevertheless, this simple scenario does provide an intuitive example of a situation where it is possible to receive *high-quality information services using low-quality positional information.* It is this intuition that motivates this research.

2 Background

2.1 Location Privacy

Within a pervasive computing environment, failure to protect location privacy has been associated with negative effects across at least three distinct areas [12,27,18].

1. *Location-based "spam"*: Location could be used by unscrupulous businesses to bombard an individual with unsolicited marketing for products or services related to that individual's location.
2. *Personal wellbeing and safety*: Location is inextricably linked to personal safety. Unrestricted access to information about an individual's location could potentially lead to harmful encounters, for example stalking or physical attacks.
3. *Intrusive inferences*: Location constrains our access to spatiotemporal resources, like meetings, medical facilities, our homes, or even crime scenes. Therefore, location can be used to infer other personal information about an individual, such as that individual's political views, state of health, or personal preferences.

Concern about the effects of new technology on an individual's right to privacy is not new, having already surfaced, for example, with respect to GIS [23], the Internet [1], and collaborative user interfaces [16]. In most countries, there already exist legal frameworks that govern the fair use of digital information about an individual, including location (see [19] for a concise overview of the history and current status of privacy legislation and fair information practices internationally).

Regulation, whether legal or voluntary, will inevitably form the baseline for location privacy in any pervasive system. However, there are at least two reasons why regulation needs to be supported by other strategies for protecting location privacy in pervasive computing. First, new regulation often lags behind new technology and ideas. Second, regulation applies "across the board," making it is difficult to guarantee privacy protection for individuals without stifling technology and innovation.

2.2 Privacy Policies

Pervasive computing research into privacy protection is addressing the definition of policy and trust mechanisms for prescribing certain uses of location information. A variety of policy-driven approaches to location privacy in pervasive systems are reviewed in [10]. Much of this research draws on established or developing privacy policy frameworks, such as the W3C Platform for Privacy Preferences (P3P; e.g., [29, 20, 22]), the IETF draft geographic location privacy charter (GeoPriv; e.g., [24]), and personal digital rights management (PDRM, e.g., [13]).

Privacy policies are necessary in order to implement unambiguous computer-readable mechanisms for location privacy protection, and are also undoubtedly a long-term part of the landscape of pervasive computing technology. However, like regulation, privacy policies cannot offer a complete solution since such policies usually incur considerable information infrastructure overheads and are vulnerable to inadvertent or malicious disclosure of private information [12].

2.3 Anonymity and Pseudonymity

Anonymity concerns the dissociation of information about an individual, such as location, from that individual's actual identity. A distinction is often drawn between "true" anonymity, where an individual is indistinguishable from all other individuals in a set, and *pseudonymity*, where an individual maintains a persistent identity (a pseudonym) that cannot be linked to their actual identity [25]. A variety of research has addressed the problem of maintaining anonymity (e.g., [3,11,8]) and pseudonymity (e.g., [26]) within the context of location-based services.

However, anonymity and pseudonymity are, again, not a complete answer to privacy concerns in pervasive computing because:

- Anonymity presents a barrier to authentication and personalization, which are important for a range of applications [19, 15].
- Pseudonymity and anonymity are vulnerable to data mining, since identity can often be inferred from location [7, 3].

Thus, in this paper we argue that obfuscation is complementary to existing privacy protection strategies (regulation, privacy policies, anonymity, and pseudonymity) and demands further investigation.

3 Obfuscation

In this paper, obfuscation is defined as:

> the means of deliberately degrading the quality of information about an individual's location in order to protect that individual's location privacy.

3.1 Related Research

Obfuscation, or closely related ideas, have already been suggested in the literature. Hong et al. Within the Confab system, [15] use landmarks and other significant locations to refer to instead of coordinate-based geographic information. The effect of this "reverse gazetteer" is to provide less information about a user's exact location. Similarly, Snekkenes [29] suggests adjusting the precision of a individual's location as part of a mechanism for policy-based protection of location privacy.

Gruteser and Grunwald use a quadtree to explore the effects of adapting the spatial precision of information about about an individual's location according to the number of other individuals within the same quadrant, termed "spatial cloaking" [11]. Individuals are defined as *k-anonymous* if their location information is sufficiently imprecise in order to make them indistinguishable from at least $k-1$ other individuals. The authors also explore the orthogonal process of reducing the frequency of temporal information, "temporal cloaking." Obfuscation, as presented in this paper, is broadly comparable to the spatial cloaking in [11]. Despite this similarity, there are several distinct differences in our approach when compared with the work presented in [11]. These differences will be noted as they arise. At this point, it will suffice to highlight that the overall aim of [11] was to guarantee an individual's k-anonymity. By contrast, in the model and algorithm presented in this paper, we aim to enable an individual's actual identity to be revealed (thereby facilitating authentication and personalization [19]) at the same time as maintaining that individual's location privacy.

3.2 Imperfection in Spatial Information

There exist clear parallels between uncertainty in spatial information and obfuscation in location privacy. The former focuses on imperfection as the result of the inherent limitations of the measurement and representation of spatial information; the latter focuses on imperfection as the result of the deliberate degradation of spatial information quality.

Three distinct types of imperfection in spatial information are commonly identified in the literature: *inaccuracy*, *imprecision*, and *vagueness*. Inaccuracy concerns a lack of correspondence between information and reality; imprecision concerns a lack of specificity in information; vagueness concerns the existence of boundary cases in information [31, 6, 32]. So, for example, the statement "Melbourne is in New South Wales" is inaccurate (Melbourne is in fact in Victoria). The statement "Melbourne is in Australia" is at the same time more accurate (it accords with reality), but less precise (it provides less detail about where Melbourne actually is). Thus, precision and accuracy are orthogonal, although these terms are often confused or conflated in the literature (e.g., [29]). Finally, the statement "Melbourne is in south-eastern Australia" is vague, since there exist boundary locations that are neither definitely in south-eastern Australia nor definitely not in south-eastern Australia.

Potentially, any or all of these three types of imperfection could be used to obfuscate an individual's location. For example, for "Agent X" located (accurately and precisely) at the corner of Flinders Street and Elizabeth Street in Melbourne city center, we might provide the following obfuscations of that agent's location:

1. "Agent X is located at the corner of Flinders Street and Spring Street" (inaccurate, since the statement does not accord with the actual state of affairs);
2. "Agent X is located on Flinders Street" (imprecise, since the statement does not specify at which of the many locations on Flinders Street the agent is located); and
3. "Agent X is located near Flinders Street Station" (vague, since there exist boundary locations that are neither definitely near nor definitely not near Flinders Street Station).

In this paper, as in [11], we only consider the use of imprecision to degrade location information quality. Future work will also address the use of inaccuracy and vagueness in obfuscation. Deliberately introducing inaccuracy into location information as part of an obfuscation system raises some difficult questions, since it essentially requires the obfuscation system to *lie* about an individual's location. Vague spatial terms, like "near" and "left," are commonly used by humans to communicate information about geographic space[1] and have been a particular focus of qualitative spatial reasoning research over the past decade (e.g., [34, 33]).

In summary, in this paper we consider the obfuscation of location information, using imprecision, in order to protect an individual's location privacy. With respect to regulation, privacy policies, anonymity, and pseudonymity, this approach has the advantages that obfuscation:

- is flexible enough to be tailored to specific user requirements and contexts;
- obviates the need for high levels of legal and policy infrastructure;
- enables an individual's identity to be revealed, facilitating authentication and personalization; and
- combats data mining by not revealing an individual's precise location.

4 Scenario

In presenting the obfuscation algorithms and formal model (section 5) we make several assumptions about the high-level architecture and usage scenario for the obfuscation system, detailed in this section.

4.1 Location Sensing

A location-aware system is a type of context-aware system that uses information about a mobile individual's current location to provide more relevant information

[1] Although Hong et al. [15] do use the term "near" in the Confab system as part of their location privacy protection system, the term is used in the paper specifically to refer to anything within 100m of the actual location (i.e., as an imprecise term rather than as a vague linguistic term).

to that individual. There exist a wide variety of techniques for automatically sensing an individual's location, including GPS, infrared proximity sensors, and wireless RF network triangulation. The different techniques are surveyed and classified in [14]. Using one or some combination of these techniques we assume that a user's current location, in the form of georeferenced x and y coordinates, is known with a high level of precision and accuracy. We deliberately leave unstated the details of how this location information is generated, in order to focus solely on the situation where reasonably precise and accurate location is known.

4.2 Architecture

We assume that a user wants to access information relevant to his or her current location from a remote location-based service provider via a wireless communications network. Further, we assume that the location-based service provider (LBSP) has no information about the user's location, other than the information that the user chooses to reveal to the LBSP. These assumptions taken together can at first sight seem somewhat strong, and so warrant further examination.

Infrastructure. The technical characteristics of many positioning systems demand that information about an individual's location must necessarily be revealed to the second-party who maintains of the positioning system infrastructure, such as the cell phone company (see [27] for a classification of positioning systems according to their privacy characteristics). This private information will undoubtedly need to be protected in some way, usually by legal or regulatory means. Irrespective of the positioning system, an individual may often wish to access a location-based service from some third party. It is exclusively this situation that we are addressing in this paper. The desire to access location-based services from third party service providers is not unusual in today's architectures, and we anticipate this situation will become increasingly common as pervasive location-aware systems become more commonplace.

Our obfuscation architecture allows an individual to connect *directly* with a third-party location-based service provider, without the need to use a broker or other intermediary (such as cell phone company or the "location server" used in [11]). As a consequence, our architecture is lightweight and distributed: it is compatible with mobile ad-hoc networks as well as more conventional client-server networks. Although the architecture presumes an individual acts as a client for the location-based service provider, these are simply roles that different agents fulfill. It is conceivable, for example, that the location-based service provider itself is in turn a mobile user who also acts as a client for other location-based services.

Communication. One way to protect sensitive location information is simply not to communicate this information to any third party service provider (cf. [21]). However, our scenario presumes that a user must communicate information about his or her location to a third party in order to receive useful location-based services [27]. The alternative, where users carry all the information they might need around with them in their mobile devices, is not viable for most location-based services for several reasons, including:

- Mobile devices typically possess limited processing and storage capacity, making it inefficient to perform complex calculations on voluminous spatial data directly on the mobile device.
- Despite advances in positioning systems, spatial data sets remain expensive to collect and collate. The companies who collect this data would usually be reluctant to make their valuable data sets available in their entirety to mobile users.
- Maintaining copies of the same data sets across multiple mobile devices inevitably leads to data integrity and currency issues.

Further, we assume that the location-based service provider does not derive any additional positioning information from the communication itself. For example, we assume that the location-based service provider cannot infer additional information about users' locations via their devices' mobile IP addresses (a separate issue that is already being addressed by the networking community, e.g., [4]).

4.3 Summary

Our architecture is deliberately simplified, in order to abstract away from the technical details of location sensing and communication in mobile networks. But we believe the architecture is not over-simplistic. In summary, the key assumptions made by are architecture are:

- A client device uses some combination of conventional location-sensing techniques to provide precise and accurate information about the client's location.
- That client device is able to communicate directly with a third-party location-based service provider (TPLBSP) via a wireless network to obtain some information service based on the client's current location.
- The information that the client chooses to reveal about his or her location constitutes the only source of information available to the TPLBSP about that client's location.

5 Obfuscation Model

In the previous section, we defined the scenario and architecture of our proposed obfuscation system. This section aims to provide a concise formal model of obfuscation and location privacy.

5.1 Geographic Space

As a first step, we adopt a discrete model of geographic space as a graph, $G = (V, E)$. Thus, geographic locations are modeled as a set of vertices, V, with connectivity or adjacency between pairs of locations modeled as a set of edges E, where $E \subseteq V \times V$. Graphs provide a highly flexible model of geographic space. For example, a graph is a natural choice for modeling a road network, with vertices as distinct locations on the network, edges indicating the presence of a direct road link between pairs of locations. Alternatively, vertices can be used to represent regions, with edges representing adjacency or neighborhood between pairs of locations. A function $w : E \to \mathbb{R}$ is used to associate a weight with each edge. This weight can be used to model proximity between vertices, including Euclidean distance or travel time.

In previous work on location-based services, geographic space is often modeled as the Cartesian plane (e.g., [11]). Adopting a graph-based model of geographic space offers several additional capabilities, primarily:

- Graphs can be used to model constraints to movement, such as a road network, and non-metric spaces, such as time-travel spaces.
- Graphs can be embedded within the Cartesian plane (e.g., the vector model used in many GIS and spatial databases) or higher dimensional Euclidean space (e.g, modeling the location of users in all three spatial dimensions).
- Graphs are computationally efficient discrete structures.

5.2 Obfuscation in a Graph

An individual's location in geographic space is represented as a vertex $l \in V$. An obfuscation of an individual's location l is represented as a set O, such that $l \in O$ and $O \subseteq V$. For every element $o \in O$, we say that o is *indiscernible* from l. The set O provides an imprecise representation of an individual's location. At any moment in time, the individual must be located at exactly one of the vertices in O, but which one remains unspecified.

Consequently, for an obfuscation O of an individual's location, one measure of the level of privacy (imprecision) is the cardinality of the set O, $|O|$ (the number of locations that are indiscernible from the individual's actual location). The larger this set, the less information is being revealed about the individual's actual location, and the greater the individual's privacy.

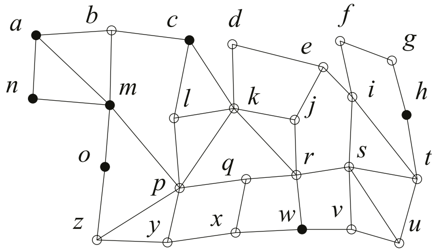

Fig. 2. Graph $G = (V, E)$ with example obfuscation $O = \{a, c, h, m, n, o, w\}$ (black vertices)

Figure 2 shows an example graph. An individual, located at one of the vertices in the graph, might represent his or her location using the obfuscation $O = \{a, c, h, m, n, o, w\}$, thereby not specifying at which of the vertices in O that individual is actually located. Note that the obfuscation O does not need to be in any sense contiguous (e.g., elements of O do not need to be connected in G). For ease of depiction, the graph in Figure 2 is planar, although planarity is also not a requirement of our model.

5.3 Proximity Location-Based Services

A simple but important subclass of location-based services concerns querying a spatial database in order to find information about features of interest that are nearest to an individual's location, which we refer to as *proximity queries*. Examples of such queries include, "With reference to my current location,

- "what is the address of the closest sushi restaurant?"
- "where is the nearest hospital?"
- "where is the nearest person waiting for a taxi?" (for a taxicab driver)

All these queries have the general form of finding the most proximal feature to a user's current location, usually returning some information related to that feature, for example the address, phone number, or price list. Data mining multiple such proximity queries could provide the basis for inferring the location (and so identity) of an anonymous or pseudonymous individual. Further, the final query might additionally demand authentication, where an individual is required to verify his or her true identity before being allowed to access privileged information (in this case, the location of a paying passenger). Obfuscation offers the ability to support authenticated location-based services, where a user cannot remain anonymous but still wishes to retain location privacy (see section 2.3).

In these queries, proximity may be quantified using a variety of different proximity measures, including Euclidean distance, network distance, or travel-time. In the graph, different proximity measures are represented using different weighting functions. Proximity queries are straightforward to answer if we know precisely where an individual is located. Even if an individual's location is not precisely known (i.e., under obfuscation), basic spatial analysis operations are enough to answer proximity queries in cases where Euclidean distance in a homogeneous Cartesian space is used (see section 1.1). The algorithm and negotiation process presented in the following sections provide a mechanism for responding to proximity queries in a weighted graph when an individual's location is not precisely known.

The discrete graph-based model of geographic space is general enough to be able to represent proximity queries in continuous Cartesian space. Thus, we can choose to represent the motivational example in section 1.1 as a weighted graph (see Figure 3). The locations of the three sushi restaurants and the pedestrian would be vertices of the graph. Edges between the pedestrian's location and the restaurants would assume as weights the Euclidean distance between these points

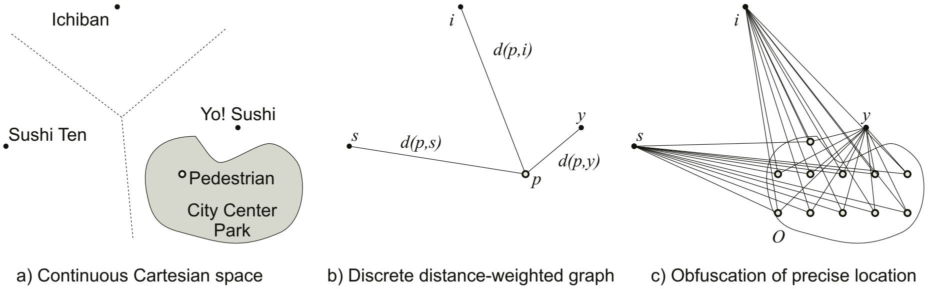
Ichiban
Sushi Ten
Yo! Sushi
Pedestrian
City Center Park
a) Continuous Cartesian space
i
d(p,i)
s
d(p,s)
y
d(p,y)
p
b) Discrete distance-weighted graph
i
y
s
O
c) Obfuscation of precise location

Fig. 3. Representation of continuous Cartesian space using discrete distance-weighted graphs

(Figure 3b). By discretizing the "City center park" region as a set of points, we could further represent the obfuscated scenario, where the pedestrian's precise location is unknown (Figure 3c).

5.4 Negotiation and Quality of Service

We are now in a position to present an algorithm for negotiation between a mobile client and a TPLBSP. The negotiation process aims to provide the best quality of service possible, whilst revealing as little information as possible about the client's location.

Algorithm 1 provides a formal description of the negotiation algorithm from the perspective of the TPLBSP. The algorithm requires as input:

- the weighted graph $G = (V, E)$ representing the geographic environment (presumably stored in the location-based service provider's spatial database);
- an obfuscation $O \subseteq V$ of the client's location $l \in O$; and
- a set Q of query vertices $Q \subseteq V$, from which the client wishes to identify the closest vertex (e.g., Q might represent the locations of all the sushi restaurants in the geographic environment G).

The preliminary lines in Algorithm 1 (1.1–1.3) define the relation δ such that for all $o_1, o_2 \in O$, $o_1 \delta o_2$ iff o_1 and o_2 have the same query vertex $q \in Q$ as their most proximal. Proximity is determined using the shortest path between the two vertices in the graph. Thus, $o_1 \delta o_2$ can be understood to mean "o_1 is most proximal to the same query vertex as o_2." By stipulating that each element has a unique most proximal query vertex (line 1.2) δ becomes an equivalence relation (reflexive, symmetric, and transitive). In fact, an equivalence relation is not a requirement of the obfuscation model, and we could recast Algorithm 1 to permit non-transitive relations (important to allow for the case where two query vertices are equidistant from an element of O). However, making δ an equivalence relation yields a much simpler algorithm, and so is used here purely for reasons of notational convenience and ease of presentation.

Any equivalence relation on a set induces a partition on that set. As a consequence, O/δ is a partition on O. If O/δ has only one element (O), then the

Algorithm 1: Negotiation proximity query with obfuscation

```
Data: The geographic environment represented as a weighted graph $G = (V, E)$,
      the set of query vertices $Q \subseteq V$, and the obfuscated location of the client
      $O \subseteq V$
Result: The pair $\langle q, C\rangle$, where $C \in (0.0, 1.0]$ indicates the confidence that $q \in Q$
        is the nearest target to the client's current location (the quality of
        service)
1.1  Define $d(v_1, v_2)$ to be the distance between $v_1, v_2 \in V$, measured using the
     shortest path through the graph $G$;
1.2  Define the function $min : O \rightarrow Q$ where $min(o) \mapsto q_1$ such that $\forall q_2 \in$
     $Q.d(o, q_1) \leq (o, q_2)$ (assume that $\forall q_1, q_2 \in Q.q_1 \neq q_2 \rightarrow d(o, q_1) \neq d(o, q_2)$);
1.3  Define the relation $\delta \subseteq O \times O$ such that $\forall o_1, o_2 \in O.o_1 \delta o_2$ iff $min(o_1) = min(o_2)$;
1.4  Construct the partition $O/\delta$ ;
1.5  if $O \in O/\delta$ then
1.6      Return the pair $\langle q, 1.0\rangle$ where $q = min(o)$ for an arbitrary $o \in O$;
1.7  else
1.8      if Client agrees to identify for its current location $l$ the equivalence class
         $[l] \in O/\delta$ then
1.9          Return the pair $\langle q, 1.0\rangle$ where $t = min(o)$ for an arbitrary $o \in [l]$;
1.10     else
1.11         if Client agrees to identify a new obfuscation $O' \subset O$ then
1.12             Reiterate algorithm with $O'$ in place of $O$;
1.13         else
1.14             Return the pair $\langle q, C\rangle$ where $C = |[o]|/|O|$ and $q = min(o)$ for some
                 $o \in O$ such that $C$ is maximized;
```

possible client locations in O must all be closest to the same query vertex (line 1.5). In this case, we can simply return the closest query vertex $q \in Q$ to some arbitrary element of O along with the confidence measure 1.0 (highest possible quality of service, total confidence in the result).

If O/δ has more than one element, then the next stage is to begin negotiations with the client. The TPLBSP now communicates with the client. The client knows its true location $l \in O$, and so potentially can identify the equivalence class $[l]$ in O/δ in which it is located. If the client is prepared to reveal this information, then it will again be possible to return the nearest query vertex with total confidence (lines 1.8–1.9). (Note that even if the client chooses to reveal this information, the TPLBSP will still not know the precise location of the client.) However, it is entirely possible that the client may not wish to reveal this much information about its location (especially if the cardinality of the set $|[l]|$ is much smaller than the cardinality of the set O). If so, then the client has two options.

First, if the client does not agree to provide any further information about its location, then the service provider returns the closest query vertex $q \in Q$ to an arbitrary location $o \in O$, along with a measure of the confidence that the returned query vertex is indeed the closest to the client's actual location (line 1.14). In our algorithm, we assume all locations in O are equally likely to be the client's location $l \in O$, and therefore for an arbitrary $o \in O$, the likelihood C that the closest query vertex to o will also be the closest query vertex to l is simply $|[o]|/|O|$ (the ratio of the cardinality of the equivalence class of o to the cardinality of O). The algorithm selects an $o \in O$ such that C is maximized, and so represents a "conservative" algorithm that returns the query vertex associated with the highest confidence level. Alternatively, the algorithm could also be adapted to select the $o \in O$ such that the distance from some $q \in Q$ to o is minimized (an "optimistic" algorithm) or that this distance is maximized (a "pessimistic" algorithm). Indeed, it is not too difficult to devise several more sophisticated ways of dealing with this case. For example, a further alternative would be to return the entire set Q of query vertices and leave the client to choose between them. However, in many cases a commercial location-based service provider might be unwilling to reveal such information, especially where Q represents high-value or high-volume data.

Second, the client can provide a new obfuscation $O' \subseteq V$ with which to reiterate the algorithm (line 1.12). At this point it becomes critical that the obfuscations O and O' are only imprecise and not inaccurate (i.e., O and O' definitely contain l). Since $l \in O$ and $l \in O'$, we know $l \in O \cap O'$. Consequently, the negotiation process requires that $O' \subseteq O$. Further, there is no purpose in a client reiterating the algorithm with the same obfuscation, $O' = O$, since that will return exactly the same result. Therefore, to continue the negotiation process a client is required to provide a strict subset of O, $O' \subset O$ (line 1.11). This is an important result, for two reasons. Since at every iteration the client is obliged to provide a strict subset of O in order to continue the negotiation process, we know that:

- the negotiation process must terminate (since O is finite); and
- the service provider needs to compute the shortest paths between locations in the graph only once, the first time the query is posed (all future iterations can reuse the same information).

5.5 Efficient Shortest Path Computation

One further aspect of importance is the complexity of the shortest path computation needed to construct the relation δ. A naive approach would be to compute the shortest path from every element of O to every element of Q. In general, single source shortest path algorithms, like Dijkstra's algorithm, have a computational complexity of $\mathrm{O}(n^2)$. As a result, the naive approach would require m iterations of the shortest path algorithm to compute this result, where m is the cardinality of O. As the size of $|O|$ increases (m approaches n), the complexity of this naive algorithm approaches $\mathrm{O}(n^3)$, the complexity of the all pairs shortest path algorithm. Thus, a naive approach significantly increases the computational complexity of this task. (A similar reasoning holds for the related task of computing the shortest path from every element of Q to every element of O.)

However, it is possible to use a single pass of Dijkstra's algorithm to compute all the shortest paths needed for the entire negotiation process in Algorithm 1, as described in Algorithm 2. Algorithm 2 is based on a multisource shortest path algorithm (e.g., [28]). Given a set of source vertices, a multisource shortest path algorithm computes the shortest of all shortest paths from any source vertex to every other vertex in the graph. Algorithm 2 is essentially a multisource shortest path algorithm which finds the shortest of all shortest paths from an element of Q to each element of O. (We assume the graph is not directed, so the shortest path from a to b is the same as the shortest path from b to a, although it would be simple to modify the algorithm to also deal with directed graphs.)

Algorithm 2 accepts the same inputs as Algorithm 1, but returns the equivalence relation δ. The first step is to add a new dummy vertex, s, to the graph G, such that there exists an edge from s to every query vertex $q \in Q$, each with zero weight (line 2.1). This step is included in the standard multisource shortest path algorithm. Figure 4 illustrates this step, showing the addition of the dummy vertex s to a graph with $Q = \{q_1, ..., q_5\}$ and $O = \{o_1, ..., o_6\}$.

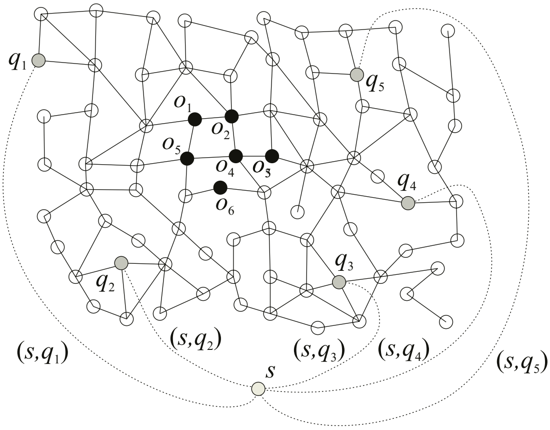

Fig. 4. Addition of dummy vertex s to obfuscated multisource graph algorithm, with $Q = \{q_1, ..., q_5\}$ (gray vertices) and $O = \{o_1, ..., o_6\}$ (black vertices)

Next the algorithm performs a single pass of Dijkstra's algorithm to yield the shortest paths from s to every other vertex in the graph (line 2.2). Finally, we compute the equivalence relation δ from this information (lines 2.3–2.4). Every shortest path begins at s and must have one element of Q as its second vertex. For each element of $o \in O$, this second vertex represents the most proximal element of Q to o. Therefore, all the information needed to compute δ can be obtained using a single pass of the Dijkstra algorithm, with computational complexity $\mathrm{O}((n+1)^2) \approx \mathrm{O}(n^2)$ (the same computational complexity as the conventional case, where no obfuscation is used).

Algorithm 2: Computation of the relation δ

```
Data: The geographic environment represented as a weighted graph $G = (V, E)$
      (with weights $w : E \rightarrow \mathbb{R}$), the set of target locations $Q \subseteq V$, and the
      obfuscated location of the client $O \subseteq V$
Result: The relation $\delta \subseteq O \times O$ such that $\forall o_1, o_2 \in O.o_1 \delta o_2$ iff $min(o_1) =
        min(o_2)$ (see Algorithm 1 line 1.3);
2.1 Construct a new graph $G' = (V', E')$ using a dummy vertex $s$ such that $V' =
    V \cup \{s\}$, $E' = E \cup S$, and $\forall e \in S.w(e) \mapsto 0.0$, where $S = \{s\} \times Q$;
2.2 Using Dijkstra's algorithm, compute the shortest paths from $s$ to all vertices in
    $G'$;
2.3 Define $second : O \rightarrow Q$ such that $second(v_o) \mapsto v_q$, where $v_q$ is the second
    vertex of the shortest path from $s$ to $v_o$ (note that the second vertex must be
    an element in $Q$);
2.4 $\forall o_1, o_2 \in O$ define $\delta = \{(o_1, o_2) \in O \times O | second(o_1) = second(o_2)\}$;
```

5.6 Summary

The obfuscation model and algorithms given above provide a mechanism to balance an individual's location privacy with that individual's need for a high quality location-based service. The more information an individual is prepared to reveal, the higher the quality of information service an individual can be provided with. The importance of such a balance between privacy and utility has already been noted in the literature (e.g., [2]). Further, the aim of this approach is to use only just enough location information to provide a location-based service: the so-called "need to know principle" [17, 29] or "principle of minimal collection" [11].

The analysis of the algorithms above shows that obfuscation can be highly efficient, in the sense that the computational complexity of answering proximity queries can be the same for the obfuscated case, where an individual does not reveal his or her precise location, as for the conventional case, where an individual's location is known by the TPLBSP. The negotiation process may incur an additional communication overhead, where the service provider requests more information from the client. However, the additional communication is bounded, because at each iteration an individual must provide a more precise specification of his or her location.

An important unanswered question concerns how a client selects the initial obfuscation O to start the negotiation process (section 5.4). There are a number of natural choices for answers to this question. A client could begin by being completely imprecise and giving no information about their location (i.e., $O = V$, the obfuscation is the entire set of vertices in the graph). The efficient shortest path computation ensures this strategy is computationally practical, although it may lead to an extended negotiation process, so increasing bandwidth and communication overheads. Where applicable, an alternative strategy would be to utilize the precision characteristics of the location sensing infrastructure. For example, if a user's location is determined within a cell phone network, the set of all locations within the same cell as the user could be used for O. More alternatives involving further research are discussed in section 6.

Finally, there are several extensions to the algorithm that were not included in the discussion above, but nevertheless could easily be introduced, including:

- extensions to allow obfuscation on a directed graph;
- extensions to fully automate the negotiation process, by setting a minimum privacy threshold for the client, in terms of the size of O;
- extensions to provide a more sophisticated confidence measure, for example a confidence interval specifying the range between the best and worst possible answers.

6 Discussion and Future Work

This paper has argued that obfuscation is an important component of an overall approach to location privacy. Obfuscation provides a framework for the provision of high quality location-based services based on low quality location information. A detailed formal model for obfuscation is presented in the paper, with respect to a simple location-based service. The model includes algorithms able to achieve a balance between location privacy and location-based service utility, using an efficient negotiation process.

In developing our obfuscation architecture in this paper, we have attempted to set out our assumptions in a clear, formal, and methodical way. Planned future research has the objective of relaxing the model assumptions in the following ways.

Spatial Configuration of Obfuscations. In addition to the alternatives in section 5.6, another strategy for initializing the obfuscation O would be for a client to define an arbitrary threshold n, such that O contains n distinct locations, $|O| = n$. However, the spatial distribution of locations in O must be carefully chosen, in order to ensure the client's true location cannot be simply deduced from analysis of O. (For example, the client's true location would be easy to compute from O if it is known that the client always initializes O as the n closest locations to the its actual location.) Current research is investigating the effects of different spatial configurations for O. Following on from this, future research will aim to determine in advance what obfuscation will provide the best balance of location privacy and quality of service to a client.

Invasions of Privacy. For information to be worth protecting, it must also be worth attacking. Ongoing research aims to analyze the techniques a hostile agent might employ to circumvent obfuscation and attempt to discover a individual's exact location, so invading that person's privacy. Beresford and Stajano [3] have already shown how effective heuristics can be as a means of invading location privacy. Our initial approach to this problem is to categorize, and then formalize as an algebra, the different heuristics that a hostile agent might apply to a series of obfuscations (for example, using assumptions about an individual's maximum and/or minimum speed of movement). However, current models of location privacy are limited by their fundamentally static nature (i.e., modeling the movement of an individual as a sequence of static snapshot locations). To overcome this limitation, our longer-term research aims to move toward a truly spatiotemporal model of location privacy, based on approaches developed for moving object databases or process-oriented models of geographic information (e.g., [30]).

User Models for Obfuscation. This paper has focused on the underlying architecture and algorithms for obfuscation. However, research is practically motivated, and current work is implementing and testing the formal model; extending the obfuscation techniques into further types of location-based services (such as wayfinding services [5]); and developing user models and usable interfaces for obfuscation systems. With respect to this last aim, we have argued throughout this paper that the negotiation process should be based on balancing the level of location privacy against the quality of location-based service. We envisage these parameters, level of privacy and quality of service, will form the basis for users to configure an obfuscated location-based service to their personal requirements. Therefore, future work will also develop improved indicators and measures of level of privacy and quality of service that can be easily understood and manipulated by non-expert users. For example, the size of the obfuscation set, $|O|$ is the simple but naive measure of location privacy used in section 5.2. A more sophisticated measure of location privacy would need to reflect the spatial properties of locations in O (for example, the accessibility of the locations, or the number of other individuals at those locations) in order to adequately represent the level of location privacy to an obfuscation system user.

Acknowledgements

Dr Duckham is partially supported by an Early Career Researcher Grant and a Edward Clarence Dyason Fellowship from the University of Melbourne. Dr Kulik is partially supported by an Early Career Researcher Grant from the University of Melbourne. The authors are grateful to the four anonymous reviewers for their thorough and constructive comments.

References

1. M. S. Ackerman, L. F. Crannor, and J. Reagle. Privacy in e-commerce: Examining user scenarios and privacy preferences. In *Proc. 1st ACM conference on Electronic Commerce*, pages 1–8. ACM Press, 1999.
2. M. S. Ackerman, T. Darrell, and D. J. Weitzner. Privacy in context. *Human-Computer Interaction*, 16(2, 3, & 4):167–176, 2001.
3. A.R. Beresford and F. Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

4. R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation Onion router. In *Proc. 13th USENIX Security Symposium*, 2004.
5. M. Duckham, L. Kulik, and M. F. Worboys. Imprecise navigation. *Geoinformatica*, 7(2):79–94, 2003.
6. M. Duckham, K. Mason, J. Stell, and M. Worboys. A formal approach to imperfection in geographic information. *Computers, Environment and Urban Systems*, 25:89–103, 2001.
7. S. Duri, M. Gruteser, X. Liu, P. Moskowitz, R. Perez, M. Singh, and J-M. Tang. Framework for security and privacy in automotive telematics. In *Proc. 2nd International Workshop on Mobile Commerce*, pages 25–32. ACM Press, 2002.
8. F. Espinoza, P. Persson, A. Sandin, H. Nyström, E. Cacciatore, and M. Bylund. GeoNotes: Social and navigational aspects of location-based information systems. In G. D. Abowd, B. Brumitt, and S. Shafer, editors, *Ubicomp 2001: Ubiquitous Computing*, volume 2201 of *Lecture Notes in Computer Science*, pages 2–17. Springer, 2001.
9. General Assembly of the United Nations. Universal declaration of human rights. United Nations Resolution 217 A (III), December 1948.
10. W. W. Görlach, A. Terpstra and A. Heinemann. Survey on location privacy in pervasive computing. In *Proc. First Workshop on Security and Privacy at the Conference on Pervasive Computing (SPPC)*, 2004.
11. M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proc. MobiSys '03*, pages 31–42, 2003.
12. M. Gruteser and D. Grunwald. A methodological assessment of location privacy risks in wireless hotspot networks. In D. Hutter, G. Müller, and W. Stephan, editors, *Security in Pervasive Computing*, volume 2802 of *Lecture Notes in Computer Science*, pages 10–24. Springer, 2004.
13. C. A. Gunter, M. J. May, and S. G. Stubblebine. A formal privacy systems and its application to location-based services. In *Proc. Workshop on Privacy Enhancing Technologies*, Toronto, Canada, 2004.
14. J. Hightower and G. Boriello. Location systems for ubiquitous computing. *IEEE Computer*, 34(8):57–66, 2001.
15. J. I. Hong and J. A. Landay. An architecture for privacy-sensitive ubiquitous computing. In *Proc. 2nd International Conference on Mobile Systems, Applications, and Services*, pages 177–189. ACM Press, 2004.
16. S. E. Hudson and I. Smith. Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems. In *Proc. ACM conference on Computer Supported Cooperative Work*, pages 248–257. ACM Press, 1996.
17. D. Hutter, W. Stephan, and M. Ullmann. Security and privacy in pervasive computing: State of the art and future directions. In D. Hutter, G. Müller, and W. Stephan, editors, *Security in Pervasive Computing*, volume 2802 of *Lecture Notes in Computer Science*, pages 284–289. Springer, 2004.
18. E. Kaasinen. User needs for location-aware mobile services. *Personal and Ubiquitous Computing*, 2003.
19. M. Langheinrich. Privacy by design—principles of privacy-aware ubiquitous systems. In G. D. Abowd, B. Brumitt, and S. Shafer, editors, *Ubicomp 2001: Ubiquitous Computing*, volume 2201 of *Lecture Notes in Computer Science*, pages 273–291. Springer, 2001.
20. M. Langheinrich. A privacy awareness system for ubiquitous computing environments. In G. Borriello and L. E. Holmquist, editors, *UbiComp 2002: Ubiquitous Computing*, volume 2498 of *Lecture Notes in Computer Science*, pages 237–245. Springer, 2002.

21. N. Marmasse and C. Schmandt. Location-aware information delivery with comMotion. In *Proceedings 2nd International Symposium on Handheld and Ubiquitous Computing (HUC)*, pages 157–171, Bristol, UK, 2000.
22. G. Myles, A. Friday, and N. Davies. Preserving privacy in environments with location-based applications. *Pervasive Computing*, 2(1):56–64, 2003.
23. H. J. Onsrud, J. Johnson, and X. Lopez. Protecting personal privacy in using geographic information systems. *Photogrammetric Engineering and Remote Sensing*, 60(9):1083–1095, 1994.
24. J. Peterson. A presence-based GEOPRIV location object format. `http://www.ietf.org/internet-drafts/draft-ietf-geopriv-pidf-lo-03.txt`, September 2004.
25. A. Pfitzmann and M. Köhntopp. Anonymity, unobservability, and pseudonymity—a proposal for terminology. In H. Federrath, editor, *Designing Privacy Enhancing Technologies*, volume 2009 of *Lecture Notes in Computer Science*, pages 1–9. Springer, 2001.
26. T. Rodden, A. Friday, H. Muller, and A. Dix. A lightweight approach to managing privacy in location-based services. Technical Report Equator-02-058, University of Nottingham, Lancaster University, University of Bristol, 2002.
27. B. Schilit, J. Hong, and M. Gruteser. Wireless location privacy protection. *IEEE Computer*, 36(12):135–137, 2003.
28. R. Sedgewick. *Algorithms in Java, Part 5: Graph Algorithms.* Addison Wesley, 3rd edition, 2003.
29. E. Snekkenes. Concepts for personal location privacy policies. In *Proc. 3rd ACM conference on Electronic Commerce*, pages 48–57. ACM Press, 2001.
30. M. F. Worboys. Event-oriented aproaches to geographic phenomena. *International Journal of Geographic Information Science*, 2005. In press.
31. M. F. Worboys and E. Clementini. Integration of imperfect spatial information. *Journal of Visual Languages and Computing*, 12:61–80, 2001.
32. M. F. Worboys and M. Duckham. *GIS: A Computing Perspective.* CRC Press, Boca Raton, FL, 2nd edition, 2004.
33. M. F. Worboys, M. Duckham, and L. Kulik. Commonsense notions of proximity and direction in an environmental space. *Spatial cognition and computation*, 2004. Accepted.
34. M.F. Worboys. Nearness relations in environmental space. *International Journal of Geographical Information Science*, 15(7):633–651, 2001.