Distributed Computing manuscript No.
(will be inserted by the editor)

Rida A. Bazzi · Goran Konjevod

On the Establishment of Distinct Identities in Overlay Networks

Received: date / Accepted: date

Abstract We study ways to restrict or prevent the damage that can be caused in a peer-to-peer network by corrupt entities creating multiple pseudonyms. We show that it is possible to *remotely* issue certificates that can be used to test the distinctness of identities. Our certification protocols are based on geometric techniques that establish location information in a fault-tolerant and distributed fashion. They do not rely on a centralized certifying authority or infrastructure that has direct knowledge of entities in the system, and work in Euclidean or spherical geometry of arbitrary dimension. They tolerate corrupt entities, including corrupt certifiers, collusion by either certification applicants or certifiers, and either a broadcast or point-to-point message model.

Keywords sybil attack · identity verification · overlay networks · peer-to-peer systems · distance geometry

1 Introduction

In a large scale peer-to-peer overlay network, physical entities that reside on different physical nodes communicate with each other using pseudonyms or logical identities. In the absence of direct physical knowledge of a remote entity or a certification by a central authority that a particular identity resides in a particular node, an entity can appear in the system under different names or *counterfeit* identities. Counterfeit identities are problematic in a peer-to-peer system because they can prevent entities from performing a remote operation, such as saving a file, multiple times to increase availability. An entity might select different identities to perform an operation, but these identities might all reside on the same corrupt entity, resulting in a loss of redundancy. Counterfeit identities can also prevent the formation of reliable reputation-based recommendation systems. An entity that can create counterfeit identities can also create identities with fake reputations, thus making reputations meaningless. Douceur [10] calls the forging of multiple identities a *Sybil attack*.

In this paper we study ways to restrict or prevent the damage that can result from corrupt entities performing Sybil attacks. We are interested in mechanisms to restrict the damage due to the creation of pseudonyms, while not relying on a centralized certifying authority or infrastructure with direct knowledge of entities in the system. While standard authentication techniques work well to prevent impersonation of existing identities, they do not address the issues arising from proliferation of pseudonyms. The first work that studies counterfeit identities that we are aware of is the paper by Douceur [10]. He argues that under the strictest requirements, that is, in a fully distributed system without a central authority and in which entities communicate by broadcasting messages, the only means to limit the generation of multiple identities is by exploiting the fact that resources of individual entities are bounded.[1] Douceur argues that, by requiring entities to dedicate significant portions of their resources to establish their identities, one could, at least theoretically, limit the number of identities that are forged by a corrupt entity. The three types of resources

The second author was supported in part by the NSF Grant CCR-0209138

Rida A. Bazzi
Computer Science and Engineering Department
Arizona State University
Tempe, AZ 85287
Tel.: 1-480-965-2796
E-mail: bazzi@asu.edu

Goran Konjevod
Computer Science and Engineering Department
Arizona State University
Tempe, AZ 85287
Tel.: 1-480-965-2783
E-mail: goran@asu.edu

[1] The absence of a central authority precludes the use of IP addresses to identify entities because they rely on the authority of the Internet Corporation for Assigned Names and Numbers (ICANN). Furthermore, in real systems IP addresses can be spoofed and a host might be provided dynamic IP addresses by its ISP.

he considers are: computation, communication and storage.

Our main observation is that the damage caused by Sybil attacks in Douceur's model is not only due to the fact that corrupt entities can forge multiple identities. The damage caused by Sybil attacks is also due to the fact that, in the absence of additional information, for any two identities one of which resides on a corrupt entity with unbounded resources, one cannot conduct a test to determine that their entities are distinct. So, our goal need not necessarily be to test that any two particular identities reside on distinct entities, but rather to test that amongst a group of identities, a large enough subset of them resides on a set of distinct entities. We call the problem of determining the number of distinct entities on which a group of identities reside the *group distinctness* problem. When determining the exact number of distinct entities is not possible, we determine a lower bound on that number. We call a test to solve the group distinctness problem a *group-distinctness test*. Realizing such a test would allow the remote execution of remote operations and therefore circumvent the harm done by Sybil attacks. An example illustrates this point. Assume that one can divide identities into two separate groups such that any two identities chosen from different groups are distinct, but two identities from the same group are not necessarily distinct (we will abuse terminology and say that the identities are distinct when they reside on distinct entities). For concreteness, also assume that there is only one corrupt entity in the system. Under these assumptions, if an entity asks n entities in one group to perform an operation and another n entities in the other group to perform the same operation, it can be guaranteed that at least n distinct entities performed the operation even though it cannot tell which ones they are. If the operation consists of saving a file, the entity can be assured that there are enough correct replicas of the file in the system. The goal of this paper is to show group-distinctness tests are possible and to explore conditions under which such tests can be made accurate.

We develop our work by exploiting an ingredient that was eliminated by the strong assumptions of Douceur, namely that entities have locations. Most of the real distributed systems we can imagine are in some way embedded in space with geometric properties. Moreover, entities in the system have (at any moment in time) their own physical locations and no two entities share the exact same location at any moment (our model will allow entities to share locations if their locations cannot be distinguished by remote entities). In general, we can assume that the underlying space (whose points include all the participants in the protocol) has a geometric structure of standard d-dimensional Euclidean space $\mathbb{R}^d$ or sphere $\mathbb{S}^d$. For sound or radio communication these assumptions are quite realistic (even though accuracy of measurement is always an issue) and they have already been exploited for secure location verification [18], while in the case of Internet-based overlay networks they are justified by recent work on estimating network distances [19] (we further discuss these assumptions as they relate to the Internet in Section 2) . We distill our assumptions into the following:

1. the actual distances between pairs of entities at least approximately satisfy two of the three metric properties: symmetry and triangle inequality; and
2. the transfer of a message back and forth between two identities takes time that is lower-bounded by a (non-decreasing) function of the distance between the two entities on which they reside.

We do not assume that logical identities are always honest, and we place no bounds on the computational resources of corrupt entities. However, since each entity is located at a point in a geometric space, its communication with the rest of the identities in the system is restricted by the geometry of the space. In particular, a simple assumption of finite message propagation implies our second assumption above: the time in which a message is transmitted from point x to point y gives an upper bound on the distance $d(x, y)$ between the two points. Note that our model allows multiple entities to reside at the same point in the geometric space.

1.1 An example

To illustrate how physical locations can be used to provide a test of distinctness, consider two correct entities A and B at a distance d from each other. Assume that there is only one corrupt entity C in the system, that C has unbounded resources, and that C is within a radius of $d/2$ from A. Under these conditions, C can forge an unbounded number of identities, but none of these forged identities can pretend to be at a distance less than $d/2$ from B. In fact, for each identity c of C, one can request from A and B an upper bound on their distance to c. This bound can be obtained by having A and B broadcast probe messages to c and measure the time it takes to receive a reply from c. Since the distance from C to A is less than $d/2$ and the distance from A to B is d, it follows from the triangle inequality that the roundtrip time of probes sent from B to C (under any of its pseudonyms) will always indicate a distance that is larger than $d/2$ and therefore none of C's identities can prove that they are within radius $d/2$ from B. Using this test of distinctness, an entity can require that a remote operation be executed by n identities that can prove that they are within a radius of $d/2$ from A and another n identities that can prove that they are within a radius of $d/2$ from B and therefore be guaranteed that enough correct entities executed the operation. One can use the distance between A and B and their distance from c as a certificate of identity of c as follows. For a group of identities, whose certificates are all computed with respect to the same A and B and such that m of the certificates have

distances from A that are less than $d/2$, n certificates have distances from B that are less than $d/2$, and l certificates that have distances from A and B that are at least $d/2$, then at least $\min\{m, n, l\}$ of the identities are distinct. Note that for such a group of certificates one can give a lower bound on the number of distinct entities in the groups while in many cases it is not possible to determine the exact number of distinct identities in the group. Our goal is to get as high a lower bound as possible for such a group-distinctness test.

We should emphasize that A and B in the example above are not the same as a *centralized* certifying authority (we discuss this point further in Section 10). In fact, A and B's knowledge of C or its forged identities is obtained solely through *remote interaction* with C's various identities and with each other and the assumption that they are both honest (which we will not require in general), whereas a centralized certifying authority requires some form of direct knowledge of C. Also, note that A and B need not know each other's location, they only need to know the distance between them and that they are both correct.

1.2 Paper Outline

The goal of this paper is to study various scenarios under which entities such as A and B in the example above can be used to significantly restrict the types of counterfeit identities by corrupt entities and therefore eliminate the harm caused by Sybil attacks. We show that one can construct certificates that are much more powerful than the one suggested above and that can be used under stronger adversarial conditions. The rest of the paper is organized as follows. Section 2 defines our system model. Additional discussion of the model, mostly dealing with issues that need to be resolved in a practical application of our work, is presented in Section 10. Section 3 explains how our model can be applied to a wireless network. Section 4 introduces group-distinctness tests and geometric certificates. Section 5 summarizes our results and contributions and discuss related work. Section 6 collects in one place the notation and terminology of the paper. Finally, in Section 7 we begin the technical development of our results.

2 System Model

We consider a system consisting of a set $\mathcal{B}$ of n *beacons* and a set of $\mathcal{A}$ of *applicants*. The set $\mathcal{A} \cup \mathcal{B}$ is the set of *participants*. We assume the participants are points in either the standard d-dimensional Euclidean space $\mathbb{R}^d$ or the d-dimensional unit sphere $\mathbb{S}^d$. In making statements that hold for both $\mathbb{R}^d$ and $\mathbb{S}^d$, we refer to the space as X. We denote by ρ the metric in X, that is, if $x, y \in X$, then $\rho(x, y)$ is the distance between x and y. We assume that participants are not mobile and that their locations are fixed.

Our model is not tied to any particular underlying system. It can be applied to overlay as well as wireless or other networks as long as the basic assumptions are satisfied. A particular system might satisfy the assumptions only for a subset of its points. For instance, a system might satisfy the metric assumptions only for sets of points that are a minimum distance θ of each other. We discuss how our assumptions map to a particular system in Section 3.

In what follows we list our remaining assumptions about various aspects of the system, namely: communication, synchrony, and failures.

2.1 Communication

Beacons communicate with each other and with applicants by exchanging messages. We assume that beacons communicate always using broadcast messages. This is in keeping with the beacons' a priori ignorance of the locations of applicants. In what follows we detail our assumptions about communication between beacons and applicants.

2.1.1 Broadcast and Point-to-Point

We distinguish two models for the messages transmitted by the participants: *broadcast* and *point-to-point*. We assume that our broadcast (or multicast) primitive is an indivisible operation that cannot be split into multiple point-to-point operations. In other words, a beacon will be able to tell if a message it receives is a broadcast message and an applicant cannot make a point-to-point message look like a broadcast message. A number of our results assume that applicants must communicate using broadcast; these results do not hold if the broadcast operation cannot be distinguished from multiple point-to-point operations. By restricting applicants to broadcast communication, we are able to tolerate more adversarial conditions. It is interesting to note, though, that restricting communication to broadcast makes it harder for faulty applicants to benefit from colluding, whereas in Douceur's argument, broadcast makes it harder to prevent Sybil attacks. The broadcast model is meaningful in general overlay networks in which a *message flooding* primitive is available for communication. Certificates can then be established using flooding exclusively. The broadcast model is also meaningful if the applicants cannot modify the underlying communication interface and a broadcast or multicast primitive is available for communication. Finally, the broadcast model is natural for wireless networks in which nodes do not have directional antennas [15].

2.2 Bounded-range Broadcast

We also consider a model in which beacons can bound the range of a broadcast. In such a model, a broadcast will only reach applicants that are within a given radius from the broadcaster. We distinguish two variants on this model. In the basic model, a recipient of a broadcast message with bounded range can determine the range of the broadcast (note that the distance between a recipient and a broadcaster might be smaller than the range of the broadcast). In another model, we assume than an applicant that receives a bounded-range broadcast message can only tell that the range of the broadcast is at least equal to the distance between itself and the broadcaster. In other words, an applicant cannot tell the ultimate reach of a signal it receives. These models of communication are motivated by wireless sensor networks. While the main thrust of our work emphasizes overlay networks, our techniques are also applicable to other settings in which communication delay has geometric properties. As we noted in the introduction, wireless communication systems present such a setting and our techniques are applicable to such systems. In such systems, there exists the additional capability of limiting the range of communication by reducing the transmission power. This corresponds to our idealized model in which messages are received by all applicants within a given radius from a broadcaster and no applicants beyond that radius. In practice, the range does not have the shape of a step function, but rather the signal strength tapers off exponentially with distance. The main model we are interested in is one in which applicants are not able to use measurements of the power of a signal to determine the ultimate range of the signal (we still assume though that the broadcasting beacon's location is known to all applicants, which only makes it harder for beacons to tolerate faulty applicants).

This model can also be meaningful in computer networks if corrupt applicants do not have access to the routing functionality. In that case, a bounded-range broadcast can be implemented with a time-to-live (TTL) field in the message header. That field gets decremented as the message is forwarded from one node to the other in order to reflect the elapsed time since the message was sent. This idea is similar to that of packet leashes [13] which have been proposed to handle wormhole attacks.

2.3 Failures

Some applicants and some beacons might be faulty (or corrupt). When we consider corrupt beacons, we assume that no more than f of them are corrupt and the remaining are correct (honest). A corrupt beacon can report fake distances to any participant, and these distances can be smaller or larger than the actual distance to the participant. An applicant can also be corrupt and it can delay its responses for probes from the beacons therefore making it appear farther away than it really is. We consider cases in which applicants might collude and cases in which corrupt applicants do not collude. To strengthen our adversarial model, and unless otherwise noted, we assume that corrupt applicants and beacons know the locations of all beacons and that correct beacons know about each other's locations only what can be inferred from the time it takes to receive replies from probes. We assume that the distance between two correct entities is a non-decreasing function of the roundtrip message delay between them (we discuss this and our other assumptions in detail in Section 10). We use μ for the distance as *measured* by exchanging messages between points. Thus $\mu(A, B)$ is the distance A can deduce from the roundtrip time of a message transmitted from A to B and back. For correct participants A and B, we assume that $\mu(A, B) = \rho(A, B)$ that is, the distance can be accurately measured by observing the roundtrip delay. Note that in the presence of faulty participants μ is not necessarily symmetric. For a participant A in the system, we denote by $x(A)$ the location of A in the underlying geometric space.

Throughout, we assume that corrupt entities have unbounded computation power. Our protocols implicitly assume that an entity cannot anticipate probe messages and send replies before the receipt of the actual probes. This assumption can be easily enforced by using the standard technique in which each probe message includes a randomly generated string that only the sender knows and that must also be included in the reply. This way, an entity would have only a very small probability of successfully being able to reply before receiving a probe.

2.4 Synchrony and Reliability

We assume that the system is asynchronous and that message transmission is not reliable, but that there are periods of time during which message transmission is synchronous and reliable. We assume that for large enough time intervals, the system will enter a synchronous period. While these assumptions do not really simplify online communication between peers in an overlay network, because peers cannot wait for the periods of synchrony, they are necessary for establishing geometric certificates. The idea is to have participants probe each other for a somewhat long period of time in order to get an accurate measure of distance. In fact, we expect that the measurements during periods of synchrony (low congestion periods) accurately reflect the distances between correct participants. Once certificates are obtained, we do not require any synchrony assumptions for communications between participants. Beacons have local clocks and the rate of drift of these clocks is small enough so that clock drift is negligible during the time it takes to establish a certificate.

3 Specific System Model

We illustrate the applicability of our model by explicitly mapping our assumptions to *wireless* networks. Section 10 further discusses the applicability of our results to overlay networks in the Internet. For this section, we use terminology from [16].

Metric Assumptions In wireless sensor networks there are multiple possible candidates for the distance measure. The Time of Flight (ToF) or Time of Arrival (TOA) is the equivalent of our roundtrip delay and it readily satisfies the metric assumptions. Accurately measuring the time of flight is possible [16] and it requires synchronization between applicants and beacons. An alternative measure used in wireless networks is Time Difference of Arrival (TDOA) which, instead of measuring ToF, measures the difference between the times of arrivals of signals from a given sensor to beacons. The advantage of using TDOA is that it only requires synchronization between beacons, but it still assumes that ToF defines a metric. The Receiver Signal Strength Indicator (RSSI) is another possibility for measuring distance. It is based on a model of signal fading with distance, but it is not as accurate as TDOA.

As long as the overhead imposed by the protocols is not large, the time of flight satisfies metric assumptions.

Communication In wireless networks, both broadcast and point to point communication is possible. Broadcast requires no special hardware support, while point to point communication is possible with the use of directional antennas [15].

Synchronization Synchronization is possible in wireless networks. Synchronization between applicants and beacons is more difficult to achieve, especially if applicants have limited resources. Synchronizations between beacons is easier to achieve.

Failures Our model of failures encompasses all kinds of malfunctions that can occur in a wireless sensor network, including sensor nodes and beacons being taken over by an adversary.

4 Geometric Certificates

4.1 Geometric Certificates

An applicant can request a *geometric certificate* from a set of beacons. When an applicant requests a geometric certificate, the beacons and the applicant execute a protocol that might require the applicant (as well as other beacons) to respond to probe messages. The protocol might also require the applicant to send probe messages to the beacons and report distances to various beacons. The result of these exchanges will be a geometric certificate: a set of distance values between the beacons and the applicants that are signed by the beacons as well as the applicant. We assume that the beacons' public keys are known to applicants. An applicant's public key is provided by the applicant. Its function is to simply tie the applicant's identity to a key. In Section 10 we explain why our assumption about the beacons' public keys does not imply that we need a central certification authority to verify the identities of applicants.

4.2 Group-Distinctness Test

A *group-distinctness test* is a function $D : 2^{\mathcal{C}} \rightarrow \mathbb{N}$ that assigns to a group of certificates a number that is a lower bound on the number of entities that were involved in obtaining these certificates. A trivial lower bound for any group is 1 and our goal is to get as high a lower bound as possible.

A special case of a group-distinctness test is a test in which the group is restricted to two elements. In that case, we define a *2-distinctness test* as a function $D : \mathcal{C} \times \mathcal{C} \rightarrow \{\textbf{true}, \textbf{unknown}\}$ that assigns to a pair (c_1, c_2) of geometric certificates a value in the set $\{\textbf{true}, \textbf{unknown}\}$. If $D(c_1, c_2) = \textbf{true}$, then the entities that obtained these certificates are distinct.

Our approach is conservative. The distinctness tests we propose are sufficient but not necessary to establish distinctness of identities. For example, different machines that reside on the same LAN would likely appear to be at the same location to remote beacons. In fact, such machines would appear to be at the same location in the geometric space. This does not mean that we can only use one machine from a set of machines that appear to be at the same location. As our introductory example shows, to execute a remote operation, one can choose multiple groups of machines such that machines in each group appear to be at the same location. If the number of corrupt entities in the system is smaller than the number of groups of entities that are collocated, then a *group-distinctness* test applied to certificates of entities from these groups would reveal that there are multiple entities in the groups without necessarily identifying the entities that are different and the collection of groups would be guaranteed to contain at least as many entities as in the smallest amongst them.

5 Contributions and summary of results

5.1 Contributions

The main contribution of this work is the identification and introduction of the group-distinctness problem as a problem whose solution can mitigate or eliminate the effects of Sybil attacks and to show that it is possible to

remotely issue certificates that can be used to test the distinctness of identities. To our knowledge this is the first work that shows that remote anonymous certification of identity is possible under adversarial conditions.

For various settings, we exhibit 2-distinctness tests. For some settings we only provide general group-distinctness tests. Typically, we implement 2-distinctness tests by verifying the locations of identities, but one should not confuse the location verification problem with the group-distinctness problem. These two problems are somewhat related, but are fundamentally different. Verifying the location of identities is one way to establish their distinctness, but the converse is not true and distinctness tests are a more general approach to dealing with Sybil attacks because in many instances it is possible to come up with distinctness tests, even when it is not possible to do location verification; the introductory example is a case in point.

The following is a summary of our results. We present geometric certification protocols, which issue compact and easily-checkable certificates to applicants. Some protocols issue certificates that can be used in 2-distinctness tests and others issue certificates that can be used in group distinctness tests for groups of more than 2 elements. Protocols that issue certificate for 2-distinctness tests are basically location verification protocols and they outperform existing location verification protocols presented in the literature (we expand on this in the next section). Given two certified entities, a distinctness test may be performed, and if the two entities' geometric locations are distinguishable from the point of view of the beacons that participated in the certification protocol, the distinctness test will succeed and certify that the two entities are indeed distinct. Protocols that issue certificates to be used in group distinctness tests have no counterpart in the literature. The certification protocols we present work for several different settings. In all cases, we assume the number of beacons is at least $d+1$, where d is the dimension of the space; also, unless otherwise specified, the applicant entity or entities should be in the convex hull of the certifying beacon set; finally, beacons' messages are always broadcast.

1. **2-distinctness tests**
   (a) Correct participants. The applicant's identity can be established with no restriction on the applicant's location or additional restrictions on the number of beacons.
   (b) Corrupt applicants that do not collude and correct beacons. The applicant's identity can be established if it is in the convex hull of beacons (in $\mathbb{R}^d$), or the set of beacons is "sufficient" (in $\mathbb{S}^d$) without restriction on the applicant's location (for an exact definition, see Section 7.1.2).
   (c) Multiple colluding applicants. The applicants' identities can be established if they are restricted to the broadcast message passing model. In the point-to-point message passing model, the applicant's identities can be established in two dimensions if there are no more than 2 corrupt applicants in collusion. We also consider the case of multiple corrupt applicants in two dimensions.
   (d) Up to f corrupt beacons. We require at least $f + d + 1$ correct beacons in order to identify applicants; this is in addition to the requirements for the correct beacons case (single corrupt applicant or multiple colluding applicants).
2. **Group distinctness tests**
   (a) Multiple colluding applicants in the point to point model. We present a protocol that establishes a lower bound on the number of distinct applicants amongst a group of applicants and in the presence of multiple beacons.
   (b) Multiple colluding applicants in the point-to-point model and with bounded-range broadcasts used by beacons. In a system in which beacons can limit the range of their broadcasts and applicants cannot determine the reach of a message they receive, we show that in $\mathbb{R}^2$, and in the presence of three correct beacons, k faulty entities cannot simulate more than k^2 distinct points.
   (c) Multiple colluding applicants with a grid of beacons in the point-to-point model (no bounded-range broadcast). We present a protocol for the setting where a set of beacons are equally spaced around the perimeter of a square and we give a lower bound on the number of entities corresponding to a group of certificates. The protocol has the desirable feature that the number of corrupt applicants needed to simulate any point inside the square is quadratic in the number of beacons.

5.2 Related work

There is a substantial body of literature that is related to the results in this paper. Giving an overview of related work is difficult and presents many subtle problems, because of the differences in terminology and assumptions made in various works. We do not aim to give an exhaustive overview of related work and we only present work that is most closely related to the results of this paper.

5.2.1 Coordinate-based network distance prediction

Ng and Zhang [19] model the Internet as a geometric space by using measurements of roundtrip delay for ICMP ping messages between a set of known hosts *probes* and several sets of *targets*. They assign the targets to points in a coordinate system, by defining each coordinate of a node as its distance from one of the probes. This *embedding* into a low-dimensional Euclidean space allows them to derive simple lower and upper bounds on distances between targets from their probe-target measurements by using the triangle inequality. Our work is motivated by

these results in considering embeddings of overlay networks in Euclidean space.

5.2.2 Triangulation and embeddings

Kleinberg et al. [14] design algorithms that try to infer a complete distance matrix of a finite set of points, given only the distances from a small number of selected points (*beacons* in their terminology) to every other point. They show that it is possible to reconstruct most of the distances, but also that arbitrary distortion of a certain fraction of all distances is unavoidable. They use some of the powerful recent results on metric embeddings and provide very general algorithms, however their results do not seem to have immediate applications to the Sybil attack problem.

5.2.3 Sybil attack

The Sybil attack was introduced by Douceur [10]. We already discussed that work in the introduction and we further discuss it in Section 10. After Douceur's paper there were a few attempts to deal with the Sybil attack.

Newsome et al. [18] study the Sybil attack in the context of sensor networks. They describe several approaches to Sybil attack prevention, designed to cope with the limited resources of sensor nodes. For example, one of their approaches relies on radio resource testing (assuming no node can listen simultaneously on several frequencies). Another is random key predistribution, where neighboring nodes establish secure links, which is more useful for maintaining a Sybil-attack-resistant infrastructure, than for building one. They describe location verification as an open problem.

Sastry et al. [21] describe a protocol that uses node *location verification* to establish node identities. However, their methods only use single beacons (*verifiers* in their terminology). More precisely, the verifiers in their approach can only test whether a given node is within a given region that surrounds the verifier. They do not describe ways to determine the exact location of a node. Also, they do not consider adversarial conditions such as faulty verifiers or collusion by applicants.

5.2.4 Node replication prevention

Parno et al [20] study protocols for prevention of node replication. In their model, an adversary can take control of a node and any private keys it might have and then attempt to clone it. Surprisingly, their model restricts the clones to follow the original protocol and their work does not tolerate corrupt nodes. More importantly, the work assumes that "the adversary cannot readily create new IDs for nodes", so in effect it does not deal with Sybil attack.

5.2.5 Beacon-based location verification

Independently of our work, Čapkun and Hubaux [7,8] consider the problem of establishing the location of a sensor in a wireless network through the measurement of distances from multiple verifiers. They show how in 2-dimensional (respectively, 3-dimensional) space having upper bounds from 3 (respectively, 4) verifiers suffices to establish the location of a node or detect cheating. This is a special case of our Theorem 1. Most of their work focuses on the case where beacons (verifiers) are correct and honest, and sensors do not collude. They also give an example of a collusion attack where three colluding nodes, positioned so that one is very close to each of three verifiers, can appear to these verifiers as a single node whose position may be anywhere within the triangle spanned by the three colluding nodes. This is the situation we discuss in Section 7.2.2. In Theorem 4 we show that fewer than three colluding nodes cannot achieve the same by giving a protocol to deal with the problem. Further, we show in Section 8.2 that in the bounded-range broadcast model where applicants cannot determine the range of the broadcast, it is possible to tolerate any number of corrupt beacons inside the triangle formed by three beacons in two-dimensional space.

At first, our work might seem to be closely related to theirs. The common idea motivating both papers is the observation that when message roundtrip time is used to estimate the distance of a remote entity, that entity can only cheat by pretending to be farther from the verifier than its real distance, not closer. In fact, this idea appears even earlier, in the report by Waters and Felten [22]. What distinguishes our work is the ability to deal with multiple colluding entities controlled by an adversary. Čapkun and Hubaux do consider collusion by corrupt sensors, and the approaches they propose to solve the problem are (1) tamper-proofness of authentication information within each device, and (2) frequency fingerprinting—both reliable methods for proving uniqueness. However, both of these assumptions really claim that each device is uniquely identifiable by some "black-box" method. In the presence of such an assumption, again, as in the case of Parno et al. [20], Sybil attack prevention becomes trivial. Their work considers only location verification and does not consider the more general problem of group-distinctness testing that we introduce in this paper. Finally, Čapkun and Hubaux do not deal with corrupt verifiers, nor consider the general Sybil attack problem.

Thus, despite certain similarities, our results are much more general, especially as we consider corrupt beacons as well as colluding participants. Furthermore, instead of focusing on specific technologies and making detailed assumptions (that may or may not persist as new technological developments are made), we study the fundamental limitations of localization protocols in a more abstract setting. The one basic assumption that we make

is that the participants lie in a metric or almost-metric space in which communication delays depend on the geometry of the space. We do focus on a few specific classes of metric spaces (Euclidean, spherical geometry) because they seem most relevant to real-life applications.

6 Notation and terminology summary

For easy reference, in Table 1 we summarize most of the notation used in the statements and proofs of Theorems and Lemmas, as well as in some discussions throughout the paper.

7 Geometric certification protocols: 2-distinctness tests

In this section we present our results under various system assumptions. For each set of assumptions, we state our results in the form of a theorem that specifies conditions under which a participant (or group of participants) is incapable of pretending to be in a location other than the real location of the participant or one of the group members. We say that a participant (or a group of participants) simulates a point, if it can make all its communications appear to come from the point.

These results can be readily used to construct geometric certificates for the applicant. In each case, a certificate would consist of the set of measurements that is sufficient to uniquely identify the location of an entity, and a test of distinctness is simply a comparison between the two locations defined by two certificates.

All our results are stated assuming the distance between correct participants is accurately measured using roundtrip delays (as explained in Section 2). These results can be extended to the case in which measurements are not accurate. For that case, the statements of the theorems will change to specify conditions under which an applicant is incapable of pretending to be outside of a well-defined neighborhood of its actual location. In the presence of inaccuracies, a certificate consists of the measurements that establish a neighborhood of the applicant's location, and a test of distinctness is simply the test of disjointness of two such neighborhoods. While we do not describe such protocols here, our results can be generalized to account for small inaccuracies (as outlined in Section 9).

In our model, we assume only that the distances between beacons can be calculated, while the locations of beacons are unknown. Given a distance matrix M_d whose entries are the pairwise distances between points in a geometric space, it is possible to find a set of points expressed in an orthonormal coordinate system and whose distance matrix is identical to M_d [4,9]. If all beacons are correct, these methods can be used to transform a distance matrix representation into a coordinate system representation. In the presence of faulty beacons, the computed distance matrix might not be realizable in a geometric space and a coordinate representation consistent with all the beacons might not be possible. Still, even in the presence of faulty beacons, the distance matrix is realizable if it is restricted to the set of correct beacons. Our goal is then to find a realization that is guaranteed to be consistent with the set of correct beacons. Assuming that the set of correct beacons is in general position (that is, no $(d+1)$-subset is contained in a d-dimensional hyperplane, and no $(d+2)$-subset is contained in a d-sphere), this can be easily achieved by considering either all sets of $d+1$ or all sets of $d+f+1$ beacons (depending on which of the two families is smaller). In the first case, we use each $(d+1)$-set to build a coordinate representation and then check if there are another f beacons consistent with this representation. In the second case, we look for a consistent $(d+f+1)$-set of beacons. In case such a set is found, it must contain at least $d+1$ correct beacons, therefore the coordinate representation defined by this set is consistent with all the correct beacons and every beacon inconsistent with this representation can be discarded as faulty.

It is important to note that, while the procedure described above is expensive—being exponential in the (usually small constant) d, and including a verification of the positive-semidefiniteness of a matrix—it is only performed once for an applicant to establish the certificate. The size of the certificate itself is small, and the test of distinctness efficient.

7.1 Honest beacons with known locations

7.1.1 Trilateration in an honest world

If all participants in the protocol are honest, then the problem is easy. To determine the exact location of a point $x(A)$ in d-dimensional space, it is enough to know all the distances $\rho(x(A), x(B_i))$ between $x(A)$ and $d+1$ other affinely independent points $x(B_1), \ldots, x(B_{d+1})$. With this information, the point $x(A)$ can be reconstructed as follows: let S_i be the sphere of diameter $\rho(x(A), x(B_i))$ around $x(B_i)$. The point $x(A)$ belongs to S_i for every i. A sphere with center $c = (c_1, \ldots, c_d)$ and radius r is the set of all points $x = (x_1, \ldots, x_d)$ that satisfy the equation $\sum_i (x_i - c_i)^2 - r^2 = 0$. Equating the left-hand sides of the equations for S_i and S_j gives a linear equation in x_i, thus the intersection of two spheres belongs to a hyperplane. Since we assume general position, each pair S_1, S_i defines a hyperplane, which we denote by H_i. Since $S_1 \cap S_i \subseteq H_i$, it follows that $x(A) \in \bigcap_i H_i$, and we can determine $x(A)$ by solving a linear system.

7.1.2 Trilateration against cheaters

In the situation where the applicant may cheat by pretending to be at a different location, the protocol should

Table 1 Notation summary

Symbol	Meaning
$\mathcal{B}$	set of beacon entities (verifiers)
$\mathcal{A}$	set of applicant entities
n	(usually) the number of beacons; $n = \|\mathcal{B}\|$
f	upper bound on the number of faulty beacons
d	dimension of underlying geometric space (most proofs)
d	distance between two objects (early discussions)
$\mathbb{R}^d$	d-dimensional Euclidean space
$\mathbb{S}^d$	d-sphere (boundary of unit ball in $\mathbb{R}^{d+1}$)
$\rho(x, y)$	distance between points x and y
$\mu(A, B)$	observed distance between entities A and B
$x(A)$	the location of participant A in the space
B_i for various i	(usually) beacon
A_i for various i	(usually) applicant
$\mathbb{B}(x, r)$	ball around x of radius r
X	(in most proofs) the host metric space, usually Euclidean or spherical
$\mathrm{conv}(S)$	the convex hull of set S
e_i	the standard i-th coordinate vector (in Euclidean space)

compute the applicant's position or detect the cheating. We first discuss the case where a single point attempts to cheat without colluding with other entities.

Consider an applicant at A that attempts to impersonate a point $x' \neq x(A)$. The applicant contacts $d+1$ beacons $B_1, \ldots, B_{d+1}$ and exchanges a message with each of them. Let $\mu_i = \mu(B_i, A)$. If A can successfully impersonate x', then $\mu(B_i, A) = \rho(x(B_i), x')$ for every i. Since $\mu(B_i, A) \geq \rho(x(B_i), x(A))$, it follows that $\rho(x(B_i), x(A)) \leq \rho(x(B_i), x')$ for every i, that is, $x' \in D_1 \cap \cdots \cap D_k$, where $D_i = \mathbb{B}(x(B_i), \mu(B_i, A))$, the ball of radius $\mu(B_i, A)$ around B_i. For a set Z, we use $\mathrm{int}\, Z$ to denote its interior, and $\mathrm{conv} Z$ its convex hull.

Theorem 1 *Let $X = \mathbb{R}^d$. Let B_i be a beacon with $x(B_i) = b_i$ for each $i = 1, \ldots, d+1$. If $\{b_1, \ldots, b_{d+1}\}$ is affinely independent and $x' \in \mathrm{int}\,\mathrm{conv}\{b_1, \ldots, b_{d+1}\}$, then x' cannot be simulated by any other point.*

Proof First, the set S of all points that can simulate x' can be written as $S = \{x \mid \forall i\ \rho(b_i, x) \leq \rho(b_i, x')\}$. If $S \neq \emptyset$, take $x^* \in S$. Since S is an intersection of closed balls, it is convex and so $x_\lambda = \lambda x^* + (1-\lambda)x' \in S$ for all $0 \leq \lambda \leq 1$. Take $\lambda > 0$ small enough that $x_\lambda \in \mathrm{conv}\{b_1, \ldots, b_{d+1}\}$. If we can show that for some i^*, $\rho(b_{i^*}, x_\lambda) > \rho(b_{i^*}, x')$, it will follow that $x_\lambda \notin S$, and the proof by contradiction will be complete. So assume that $\rho(b_i, x_\lambda) \leq \rho(b_i, x')$ for all i. Let H be the hyperplane through $\frac{1}{2}(x' + x_\lambda)$ with normal vector $x_\lambda - x'$. H is exactly the set of points at equal distance to x' and x_λ. This implies that for every i, b_i is on the same side of H as x_λ, in other words, the hyperplane H separates x' from b_i for every i. This contradicts the fact that $x' \in \mathrm{conv}\{b_1, \ldots, b_{d+1}\}$.

If the underlying geometric space X is $\mathbb{R}^d$, then the theorem above gives necessary and sufficient conditions for a set of beacons to be able to detect a cheating point. If x' is not in the convex hull of the beacon set, it may be impossible to detect cheating.

However, on the sphere $\mathbb{S}^d$, the situation is different, and in fact much better. Note first that on a sphere, the distance between a pair of points is measured along a geodesic curve (great circle) that connects the pair. The notion of convexity can then also be defined for the sphere. Given two points $x, y \in \mathbb{S}^d$, we would like to define their convex hull as the set of all points on the shorter segment of the geodesic between x and y. We will refer to the shorter of the two segments of the geodesic between two points as *the segment* between these points, and we will write $[x, y]$.

Note that there is a unique x-y geodesic as long as x and y are not antipodal points. (If x and y are antipodal, then there are infinitely many geodesics between x and y, and in fact every point on the sphere lies on one of them.)

Now consider a set of points $X \subset \mathbb{S}^d$. We say that X is convex, if

1. X is contained in some half-sphere, and
2. For every $a, b \in X$, the segment $[a, b]$ is a subset of X.

It is easy to see that convex sets under this definition satisfy some of the same properties as standard convex sets in Euclidean space.

Lemma 1 *Let $\{X_\alpha \mid \alpha \in I\}$ be a family of convex subsets of $\mathbb{S}^d$. Then the intersection $\cap_{\alpha \in I} X_\alpha$ is also convex.*

Let $x, y \in \mathbb{S}^d$ be such that $\angle x0y \leq \pi/2$, where $\angle x0y$ is the angle formed by x, the origin 0 and y. We use this angle as the measure of distance between x and y, that is, $\rho(x, y) := \angle x0y$.

Let $\mathbb{B}(x, r) = \{y \mid \rho(x, y) \leq r\}$. We call $\mathbb{B}(x, r)$ *the ball around x with radius r.*

Lemma 2 *If r is smaller than a quarter of the length of a great circle, then $\mathbb{B}(x, r)$ is a convex set.*

As a justification for our claim that the situation on the sphere is even better than in the Euclidean space, consider the following theorem.

Theorem 2 *Let $X = \mathbb{S}^{d-1}$. Let $b_1, \ldots, b_{2d}$ (the locations of beacons $B_1, \ldots B_{2d}$) be the points defined by the coordinate unit vectors $+e_1, -e_1, +e_2, -e_2, \ldots, +e_d, -e_d$ in both orientations (that is, if $\mathbb{S}^{d-1}$ is considered as a subset of $\mathbb{R}^d$, the beacons are at the vertices of the polar of the inscribed d-dimensional cube). Then no point $x' \in \mathbb{S}^{d-1}$ can be simulated by any other point.*

Proof First, we can assume without loss of generality that x' is in the interior of the positive orthant.

Indeed, suppose this is not true. Then we first project the whole space to the subspace spanned by coordinate axes in which x' is zero. After this step, all of the components of the projected x' are nonzero. Then we rename the coordinate axes so that x' belongs to (the interior of) the positive orthant. Now x' satisfies our assumptions. We next prove that such an x' cannot be simulated. In order to then generalize this result to unrestricted x', one need only observe that the only effect of the projection is that we restrict ourselves to using even fewer beacons.

We will use only the d beacons located at $+e_1, \ldots, +e_d$, We claim that for any $x^* \neq x'$, there exists an i such that $\rho(b_i, x^*) > \rho(b_i, x')$. For the most interesting case, where the simulating point is also located in the positive orthant, the proof is completely analogous to the proof of Theorem 1 for Euclidean space.

First, the set S of all points that can simulate x' can be written as $S = \{x \mid \forall i\ \rho(b_i, x) \leq \rho(b_i, x')\}$. If $S \neq \emptyset$, take $x^* \in S$. Since S is an intersection of closed balls, it is convex and so contains the whole segment $[x', x]$. Since x' is in the interior of $\mathrm{conv}\{b_1, \ldots, b_d\}$, we can take a point x_λ close enough to x' that $x_\lambda \in \mathrm{conv}\{b_1, \ldots, b_{d+1}\}$. If we can show that for some i^*, $\rho(b_{i^*}, x_\lambda) > \rho(b_{i^*}, x')$, it will follow that $x_\lambda \notin S$, and the proof by contradiction will be complete. So assume that $\rho(b_i, x_\lambda) \leq \rho(b_i, x')$ for all i.

Let H be the hyperplane through the midpoint of the spherical segment $[x' + x_\lambda]$ with normal vector $x_\lambda - x'$. $H \cap \mathbb{S}^{d-1}$ is exactly the set of points at equal distance to x' and x_λ. This implies that for every i, b_i is on the same side of H as x_λ, in other words, the hyperplane H separates x' from b_i for every i. This contradicts the fact that $x' \in \mathrm{conv}\{b_1, \ldots, b_d\}$.

As we saw from the proof of Theorem 2, the boundedness of the sphere makes it possible to use a single "universal" set of beacons of fixed size (depending linearly on the dimension of the space) to distinguish any point from any other. We do not claim that we can always place beacons exactly at the locations used in Theorem 2. However, this theorem gives a sufficient condition to ensure that every point in the space is contained in the convex hull of some set of beacons, and therefore cannot be simulated by any other point. (As long as the number of beacons is finite, this cannot be done in Euclidean space because the applicant can always be far enough to be outside of the convex hull of the beacons). In practice, we may use any appropriate beacon set, but distinguishability may be more difficult to guarantee if the beacons used are not all within a single half-sphere because then we must be very careful about convexity.

7.2 Multiple colluding entities

We have seen that it is impossible for a single applicant at point x^* to impersonate any other point x' in the convex hull of a sufficiently large set of active beacons. However, the proof was based on the fact that the distance from x^* to some beacon would have to be greater than from x' to the same beacon and so x^* couldn't return messages in time. If several entities located at different points collude to jointly impersonate another point, our protocols from Section 7.1.2 don't work anymore. In fact, in this setting there is a significant difference between the broadcast and the point-to-point communication models.

7.2.1 Broadcast messages

In the broadcast model the applicant cannot send a message to a single recipient. Instead, every message sent is broadcast and thus received by every other entity (or at least every entity expecting a message). More precisely, every message sent by an applicant A at time t is received by every beacon B_i at time $t + \rho(x(A), x(B_i))$.

Theorem 3 *Let x' be a point surrounded by an independent set of beacons $\{B_1, \ldots, B_{d+1}\}$, either in the sense of Theorem 1 in $\mathbb{R}^d$ or in the sense of Theorem 2 in $\mathbb{S}^d$. In the broadcast model, x' cannot be simulated by any set $\{A_1, \ldots, A_k\}$ of entities unless $x(A_i) = x'$ for some i.*

In the proof of the theorem, we simply use A to denote the entity of the applicant. The first reason for this is that no beacon can tell which A_i sends the message just by the content of the message since $A_1, \ldots, A_k$ are in collusion. The second reason is that the broadcast model ensures that each message sent by any of the entities $A_1, \ldots, A_k$ is received by all beacons.

Proof The protocol begins by beacon B_1 sending a probe at time 0. The applicant responds (broadcasting to all the beacons). Upon receiving the response, each beacon immediately forwards it to B_1. Now B_1 records $\mu(B_1, A)$ and the times at which it receives the responses from the other beacons. Given the distances $\rho(x(B_1), x(B_i))$, the beacon can deduce the time it took the message from A to arrive to each beacon by subtracting $\mu(B_1, A)/2 + \rho(x(B_1), x(B_i))$ from the time t_i at which the response from B_i arrived. Therefore, the beacon B_1 can calculate the distance $\mu(A, B_i)$ for each i. Now by Theorem 1 for $\mathbb{R}^d$ or by Theorem 2 for $\mathbb{S}^d$, the point x' cannot be simulated by any other point.

7.2.2 Point-to-point messages

The case in which an applicant can send a message to any single beacon appears to be substantially more difficult in the case of colluding entities. For example, our protocols from the previous section fail because the simulating entities $A_1, \ldots, A_k$ can reply to the beacons selectively, so that A_i sends a message to B_j only if A_i is closer to B_j than x'. Thus a point x' can be simulated whenever for each beacon B_i there exists an A_j such that $\rho(x(A_j), x(B_i)) \leq \rho(x', x(B_i))$. Intuitively, the communication to each beacon may be controlled by a single adversary. We present a protocol that forces the adversary to do just this, namely, a protocol that can only be cheated with the number of entities equal or greater than the number of beacons, and located in small bounded regions, one per beacon.

The protocol consists of two rounds. In the first round, the beacons exchange the initial messages with the applicant and each other, accumulating enough information to compute x' from the values of $\mu(B_i, A)$, $i = 1, \ldots, d+1$, and to compute their own locations in some fixed frame of reference. In the second round, the beacons synchronize clocks and broadcast special messages $M_1, \ldots, M_{d+1}$. For each i, the message M_i is sent by beacon B_i, so that at time t_0 all $d+1$ messages simultaneously reach x'. Each of these messages should be impossible to forge for an entity who has not received it. A simple way to achieve this is for each beacon to send a random message to the applicant. The applicant is then required to immediately combine the $d+1$ messages into a single one, and forward this to each beacon. This message is constructed so that it is difficult to forge it without having received all $d+1$ of the beacons' messages. Finally, the beacons verify that from the forwarded message they can indeed reconstruct the original messages that were sent to the applicant.

The critical observation is that unless the applicant receives all $d+1$ messages, it cannot forge the combination message.

If the protocol is completed and the beacons decide to accept the participant, then it must be true that every beacon receives the combination message on time. Since we assume that if a beacon receives a single message, the message must have been sent by a unique entity (multiple messages cannot combine themselves on the fly into a single message), we may denote by A_i the participant in charge of forwarding the combination message to B_j. From the definition of the protocol, if A_i can forward the required message to B_j on time, it must be true that

$$\begin{aligned} \rho(x(B_k), x') + \rho(x', x(B_j)) \leq \\ \rho(x(B_k), x(A_i)) + \rho(x(A_i), x(B_j)) \end{aligned} \tag{1}$$

for all $k \neq j$.

We claim that if the protocol is completed successfully from the beacons' point of view, then either one of the entities is at the claimed applicant location, or there are at least $d+1$ entities controlled by the adversary.

Before we discuss the protocol in general, we take a look at the two-dimensional case and prove that no two entities can simulate a third point in the convex hull of three beacons. This implies that if the protocol is completed successfully from the beacons' point of view, then either one of the entities is at the claimed applicant location, or there are at least 3 entities controlled by the adversary.

Theorem 4 *Let $X = \mathbb{R}^2$ and let x' be a point in the convex hull of 3 affinely independent beacons B_1, B_2, B_3. Then no 2 entities can simulate x' unless one of them is located at x'.*

Note that, just as in the broadcast model, we do not assume the entities controlled by the adversary are in the convex hull of the beacons.

Proof Denote the two entities under adversary's control by A_1 and A_2 and assume that the protocol is completed successfully. Since A_1 and A_2 successfully simulate x' , it follows that for every j, the beacon B_j receives the combination message by time $t_0 + \rho(x', x(B_j))$. Let A_i be the participant in charge of forwarding the combination message to B_j. From the definition of the protocol, if A_i can forward the required message to B_j on time, equation (1) must be satisfied.

For convenience, for all $j \neq k$ define E_{jk} to be the ellipse with focal points B_j and B_k that passes through x', and for all i, define $E_i = \bigcap_{j \neq i} E_{ij}$.

The equation (1) is satisfied by A_i exactly if A_i can forward the combination message to beacon B_j on time. On the other hand, the set of points that satisfy this equation is exactly the set E_j. In other words, A_i can forward the message to B_j on time, if and only if $A_i \in E_{jk}$ for each $k \neq j$.

Let h_{jk} be the line tangent to the ellipse E_{jk} at the point x'. Note that, since x' is in the convex hull of $\{B_1, B_2, B_3\}$, for any j, k, the beacons B_j and B_k are on the same side of h_{jk}, but B_i is on the other side. Denote by H_{jk} the halfplane bounded by h_{jk} that contains the two points B_j and B_k. The discussion in this paragraph can now be summarized by writing

$$B_i \neq H_{jk}$$

for all $i \neq j, k$.

In order to show that at the adversary needs at least three participants to simulate a point distinct from each of the participants, we will now argue that each beacon requires a distinct participant. This will follow directly if we can show that the set of points that can reach B_i on time and the set of points that can reach B_j on time have only x' in common. Indeed, we will now show that $E_i \cap E_j = \{x'\}$ for all $i \neq j$.

First, instead of E_i we consider the set $H_i = \cap_{j \neq i} H_{ij}$ for each i. Since $E_{ij} \subset H_{ij}$, it is clear that $E_i \subset H_i$ and

it will suffice to show that $H_i \cap H_j = \{x'\}$ for all $i \neq j$. We show this stronger statement by simple enumeration of all possible cases. Let B_1, B_2, B_3, x' be four points in the plane with $x' \in \text{conv}\{B_1, B_2, B_3\}$. Draw any line through x' that has B_1 on one side and B_2 and B_3 on the other. Then draw a line (again through x') that has only B_2 on one side and the other two beacons on the other. Finally, draw a third line that has B_3 on one side and B_1 and B_2 on the other. The properties just stated related to the separation of beacons by these three lines are certainly true of the lines h_{12}, h_{13} and h_{23}. Therefore, we may safely infer properties of the sets H_1, H_2, H_3 from considering the halfspaces bounded by these three lines.

The three lines drawn as specified in the previous paragraph all pass through x'. Hence they divide the plane into six wedges. Label these in clockwise order, starting with the one that contains B_1, by $W_1, W_2, \ldots, W_6$. First, $B_1 \in W_1$ and this is the only beacon in W_1, for otherwise two beacons would not be separated by any of the three lines. More generally, any one wedge W_j may contain at most one beacon. Consider now the wedge W_2. If there exists an i such that $B_i \in W_2$, then B_1 and B_i are together on the same side of two of the three lines. However, this too is impossible by the separation requirements. Therefore, W_2 contains no beacons. Similarly, W_6 contains no beacons and, more generally, for any j, at most one of W_i, W_{i+1} may contain a beacon. From all this, we may conclude that exactly W_1, W_3 and W_5 contain one beacon each. If the three lines were really drawn in the same position as h_{12}, h_{13} and h_{23}, then three of the wedges would be exactly H_1, H_2 and H_3, namely we would have $W_1 = H_1$ for $B_1 \in W_1$, and so on. Now clearly, $W_1 \cap W_3 = W_1 \cap W_5 = W_3 \cap W_5 = \{x'\}$, and thus our proof is complete.

In the conference version of this paper, we claimed the analogous result for arbitrary dimension. While this still appears true, we have not been able to formalize a proof for this general case.

Conjecture 1 Let $X = \mathbb{R}^d$ and let x' be a point in the convex hull of $d+1$ affinely independent beacons B_1, B_2, $\ldots, B_{d+1}$. Then no d entities can simulate x' unless one of them is located at x'.

The reason for the difficulty here seems to be that the statement relies on more than just affine (projective) properties of d-dimensional space, and so polyhedral arguments (such as the one in our proof of the two-dimensional case) will not suffice. On the other hand, some special cases of the conjecture are relatively easy to prove—such as when the beacons are equidistant.

Theorem 5 *Let $X = \mathbb{R}^d$ and let x' be at the center of the regular simplex spanned by $d+1$ beacons B_1, B_2, $\ldots, B_{d+1}$. Then no d entities can simulate x' unless one of them is located at x'.*

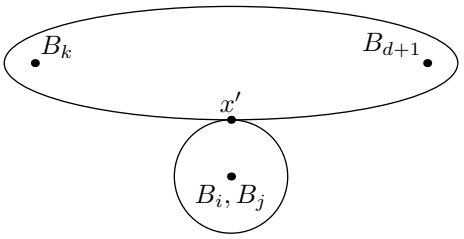

Fig. 1 The projection of two $E_{i,j}$ and $E_{k,d+1}$. is exactly $\{x'\}$.

Proof Let $A_1, \ldots, A_d$ be d entities that can successfully simulate x'. As in the proof of Theorem 4, for all $j \neq k$ define E_{jk} to be the ellipsoid with focal points B_j and B_k that passes through x', and for all i, define $E_i = \cap_{j \neq i} E_{ij}$.

Our goal is to show that $E_i \cap E_j = \{x'\}$ for $i \neq j$. Since E_i must contain an A_j for every i, this will imply that, unless one of the A_i is at x', at least one will be needed for every E_j, and thus at least $d+1$ entities will be necessary to complete the protocol.

By symmetry, it is enough to show that $E_{d+1} \cap E_i = \{x'\}$ for $i \leq d$. Since $d > 3$, we can find $j \neq i$ and $k \neq d+1$ such that $j \neq k$. For such a choice of j and k, consider $E_{i,j}$ and $E_{k,d+1}$. Since $E_i \subseteq E_{i,j}$ and $E_{d+1} \subseteq E_{k,d+1}$, we have $E_i \cap E_{d+1} \subseteq E_{i,j} \cap E_{k,d+1}$. Thus, if we can show that $E_{i,j} \cap E_{k,d+1} = \{x'\}$, the theorem will follow.

Project the simplex to a 2-dimensional plane spanned by the segment $[B_k, B_{d+1}]$ and the point x'. In this projection, $E_{k,d+1}$ is mapped to an ellipse, and $E_{i,j}$ to a circle, as in Figure 1. In the projection, the intersection of $E_{i,j}$ and $E_{k,d+1}$ is exactly $\{x'\}$. If $E_{i,j} \cap E_{k,d+1}$ contained an open ball, then this ball would project to either an open disk or an open segment in any 2-dimensional projection. Since this is not the case, the intersection $E_{i,j} \cap E_{k,d+1}$ does not contain an open ball. Suppose there was another point $x'' \neq x'$ contained in $E_{i,j} \cap E_{k,d+1}$. Then also the segment $[x', x'']$ must be contained in the intersection. However, since every point on the boundary of an ellipse is an extremal point (cannot be written as a convex combination of other points within the ellipse), any point in the interior of the segment $[x', x'']$ is also in the interior of both $E_{i,j}$ and $E_{k,d+1}$. This contradicts our previous conclusion that $E_{i,j} \cap E_{k,d+1}$ doesn't contain an open ball.

Hence, x' is the only point in $E_{i,j} \cap E_{k,d+1}$, thus also the only point in $E_i \cap E_{d+1}$, which proves the theorem.

7.2.3 Point-to-point messages, the case of many beacons

In the previous section we showed that the adversary needs at least three participants to simulate any point in the convex hull of three beacons. Intuitively, what happens is that every beacon must be cheated by a distinct point in order to avoid detection. This intuition is justified by our next result, that shows that if a larger number n of beacons is available, it is possible to formulate

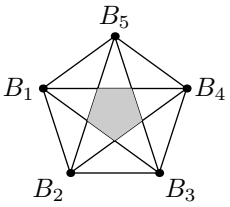

Fig. 2 The points within the inner pentagon satisfy the condition of Theorem 6.

a protocol that cannot be cheated by fewer than n participants.

Theorem 6 *Let $X = \mathbb{R}^2$, and let $\mathcal{B} = \{B_1, \ldots, B_n\}$ be a set of n beacons in convex position (that is, no beacon is in the convex hull of the others). Let x' be a point in the convex hull of these n beacons. If for any pair of beacons B_i, B_j there exists a third beacon B_k with $x' \in \text{conv}\{B_i, B_j, B_k\}$, then no $n-1$ entities can simulate x' unless one of them is located at x'.*

Notice that this condition is generally not satisfied when n is even. Also, in most cases the region in which the applicant point can be decreases as the number of beacons grows. For some examples, consider Figure 2.

Proof First, let us describe the protocol. For this theorem, we use a straightforward generalization of the three-point protocol from Theorem 4. In the first phase of the protocol, the beacons determine the unique claimed location x' of the applicant. In the second phase, each beacon B_i sends a message m_i at a time determined so that all the messages arrive at x' simultaneously. The applicant is then required to combine all the messages received and immediately forward the resulting combination message to all the beacons.

To analyze the protocol, we imagine the beacons as grouped into triples. Each triple contains the applicant point in its convex hull, and each triple is considered as if it were running the three-point version of the protocol. For any of these triples, the combination message contains all the information in the three messages of the triple, so any conclusions derived about the triple from the proof of Theorem 4 hold for the n-beacon protocol as well. In particular, if the beacon B_i receives the correct combination message on time, then the participant who sent it to B_i is contained within E_i, where E_i is the set of all points x such that $d(x, B_i) + d(x, B_j) \leq d(x', B_i) + d(x', B_j)$, for all B_j that are contained in any triple together with B_i. For any i and j, there exists a triple of beacons B_i, B_j, B_k such that $x' \in \text{conv}\{B_i, B_j, B_k\}$. Therefore, $E_i \cap E_j = \{x'\}$. If the protocol is completed, then each beacon receives the combination message on time, and so either a participant in the protocol is located at x' or each beacon receives the message from a different location. In the former case, no cheating is going on and the protocol should succeed; in the latter, at least n participants are required.

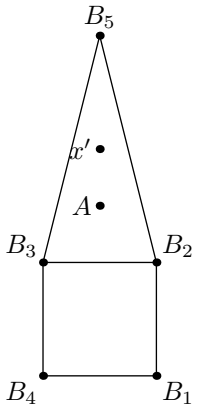

Fig. 3 Beacon B_5 is faulty and, together with A, may create an entity at x'.

7.3 Trilateration with corrupt beacons

In the presence of faults, we cannot rely on the operation of any single beacon to work as specified by the protocols. For example, a corrupt beacon may report an applicant as being further away than the actual distance, violating one of our basic assumptions and creating a situation where $\mu(x(B), x') < \rho(x(B), x')$. Note that an honest applicant may recognize this situation and we assume that a correct applicant will probe a beacon to establish its distance to the beacon and compare it to the distance reported by the beacon to the applicant. If the applicant is correct, then it will accept the value reported by the beacon if it matches its own value. This will have the effect of preventing a beacon from making a correct applicant look like it is closer to the beacon than it really is.

We now show how to tolerate beacon failures.

Theorem 7 *Let x' be the location claimed by the applicant A. Consider a set S of $n \geq d + 1 + 2f$ beacons arranged so that either: (first case) for every $(n-f)$-element subset $S' \subset S$, $x' \in \text{conv}S'$, or (second case), for every $(d+1)$-element subset $S' \subset S$, $x' \in \text{conv}S'$. If at most f of the beacons in S are faulty, then no applicant A can simulate x' unless $x(A) = x'$. A certificate for the applicant can be constructed in time $\binom{n}{f}$ (in the first case) or $\binom{n}{d+1}$ (in the second case).*

The convex hull condition may appear quite restrictive, but if n is large compared to f, then the intersection of convex hulls of all the $(n-f)$-subsets will still be considerably large. A condition such as this is needed because of examples such as the one in Fig 3. Here, the top beacon (B_5) is faulty and it may collude with the applicant. Since the applicant is not in the convex hull of the set of correct beacons, they may be fooled by the collusion of the applicant and the faulty beacon into believing the applicant is farther away from the square than it really is.

Note that a set of $d + 1 + 2f$ beacons contains at most f faulty, and therefore at least $d + 1 + f$ correct beacons. The f faulty beacons may, together with at most d correct ones, determine an incorrect value for $x(A)$. However, there are more correct beacons and so the maximum subset of beacons agreeing on a location for x' is correct. Unfortunately, finding a maximum consistent subset of a set of n linear equalities is not only NP-hard, but also hard to approximate within an n^ϵ factor for some $\epsilon > 0$ [2].

Proof We check either all sets of $d+1$ or all sets of $n-f$ beacons (depending on which of the two conditions in the theorem is satisfied, or if both are, which of the two families is smaller).

In the first case, for every $(d+1)$-set of beacons, we find a candidate point for x' (by solving a linear equality system as in Section 7.1.1). Then for each beacon B, we check if $\mu(B, A) = \rho(x(B), x')$. If there are at least $n - f$ such beacons (and x' is in their convex hull), then $x(A) = x'$ and the certificate is issued.

In the second case, if each beacon in a set of $n - f$ beacons is consistent with x' (and x' is in their convex hull), then this set of beacons defines a unique point, which must be the location of the applicant, because a $(d+1)$-subset of this set consists of correct beacons, and so also defines the actual location of A.

Theorem 8 *Let $d = 2$. Let x' be the location claimed by the applicant A. Consider a set of $n \geq d+1+2f$ beacons surrounding x', such that for every $(n-f)$-element subset $S' \subset S$, $x' \in \text{conv}S'$.*

If at most f of the beacons in S are faulty, then no set A of at most d entities can simulate x' in the point-to-point model unless $x(A_i) = x'$ for some applicant A_i.

Proof We first determine a unique claimed location for the applicant, by running the protocol from (case 1 of) the proof of Theorem 7.

In the second phase, we run the relay protocol from Theorem 4. After the messages are all received, we examine the $(n - f)$-subsets of beacons and their conclusions about the location of the applicant. Consider a set of $n - f$ beacons. If all are correct, then they will detect an adversary trying to simulate x' with fewer than $d+1$ entities. If all are correct, they will also issue a certificate to an honest applicant. Thus, we need only argue that an $(n-f)$-subset containing some faulty beacons cannot agree on an incorrect location for the applicant. But this is true, since in the first phase we determined the unique location claimed by the applicant.

8 Certification protocols: groups-distinctness tests

8.1 Group distinctness: general results

In this section, we consider a general group-distinctness test using a protocol in which applicants are required to relay messages between pairs of beacons. We show how to find a lower bound on the number of applicants needed to simulate a number of observed relay times by reducing the problem to a set cover problem.

In the relay approach, beacons do not attempt to determine the locations of the applicants directly, rather they determine the time it takes for a message sent by one beacon to be received by the applicant and then relayed to another beacon. For a pair of beacons, the time to relay a message defines an ellipse whose foci are the two beacons and whose diameter is equal to the relay time. An applicant that is on or inside the ellipse can simulate the relay time observed by the two foci by introducing appropriate delay. Applicants outside the ellipse cannot simulate the observed relay time.

For the distinctness test, we assume we have as input a set of relay times between applicants and pairs of beacons, and the goal is to determine the minimum number of applicants that could have produced this set of relay times.

Given two ellipses that have a nonempty intersection, there could be an applicant in their intersection that can simulate the observed relay times between their foci. We will take a conservative approach and assume that for a set of ellipses that have a common non-empty intersection there is a point in the intersection that can simulate the relay times for all the ellipses in the set. We call this assumption the *conservative assumption*. By abuse of terminology, we identify such a point with the intersection itself, and say that an intersection of a set of ellipses can simulate the relay times for the ellipses in the set.

Given a set of ellipses, we say that an intersection of a subset (of ellipses) is *minimal* if the intersection is not empty and does not have a proper subset that is a nonempty intersection of ellipses (equivalently, no proper superset of that set of ellipses has a nonempty intersection). For example, if an ellipse does not intersect any other, then the whole ellipse is itself a minimal intersection. We will show that our problem reduces then to determining a set of minimal intersections that are enough to simulate the observed relay times. In fact, assume that a number of applicants can simulate the observed relay times and that some applicants are not in minimal intersections. For every applicant that is not in a minimal intersection, the applicant must be in some ellipse and possibly in the intersection of a number of ellipses. Let S be the smallest intersection of ellipses to which the applicant belongs. Since S is not minimal, we can replace the applicant with another applicant in the minimal inter-

section contained in S (such a minimal intersection exists because the number of ellipses and therefore of their intersections is finite). The replacement does not affect the ability of the set of applicants to simulate or achieve the observed relay times because, by our conservative assumption, some applicant in the minimal intersection in S can simulate any relay time for all ellipses to which S belongs, which is at least as much as can be simulated by an applicant in S that is not in the minimal intersection.

So, our problem reduces to finding a set of minimal intersections that can simulate all the observed relay times.

This problem is an instance of *set cover* [12]. To see this, define a set $X = \{x_1, \ldots, x_n\}$, each of whose elements corresponds to a unique ellipse, and for each minimal intersection of ellipses from X, define a set $\{x_{i_1}, \ldots, x_{i_t}\}$ containing exactly the elements corresponding to the ellipses that form the intersection. Denote these sets by $S_1, \ldots, S_m$. Now our problem is exactly that of finding a minimum-size family of sets $\{S_{i_1}, \ldots, S_{i_c}\}$ such that the union of all the sets in this family contains every element of X.

In general, set cover is NP-hard, and even hard to approximate to a factor better than $\log n$ [11]. However, our problem is restricted by the fact that ellipses are not arbitrary subsets of the plane, and also by the fact that we only consider minimal intersections. In fact, if we had to consider arbitrary intersections of ellipses, we might have to create exponentially many sets. Since ellipses are convex and we only consider minimal intersections, the number of sets is bounded by $O(n^2)$, where n is the number of ellipses in the instance. Furthermore, since all the sets have a special geometric structure (they are "pseudo-disks" in terminology of [17]), the results of Matoušek et al. [17] imply that the set system $\{S_1, \ldots, S_m\}$ allows ϵ-nets of size only $O(1/\epsilon)$. For such set systems, Brönnimann and Goodrich [6] give a constant-ratio approximation algorithm for the set cover problem (for a discussion of these results, see also the survey by Bern and Eppstein [3] and the Thesis of Brönnimann [5]).

8.2 Group-distinctness with point-to-point messages and bounded-range broadcast

In this section we consider a system in which applicants can communicate with point-to-point messages and beacons can communicate with bounded-range broadcast and applicants cannot determine the range of a broadcast message they receive. We show that in $\mathbb{R}^2$, and in the presence of three correct beacons, k faulty entities cannot simulate more than k^2 distinct points. This result can be used in the way suggested in the introductory example (Section 1.1) to mitigate the damage of Sybil attacks. If there are no more than k faulty entities in the system, then from a set of applicants that *appear* to be at m different locations, at least $m - k^2$ applicants must reside on correct entities.

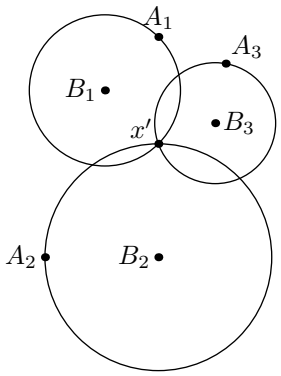

Fig. 4 Three applicants can potentially simulate x'.

To achieve our result, we modify the second phase of the protocol for the point-to-point case. The second phase is replaced with m phases, where m is a security parameter (the protocol will fail with probability inversely proportional to an exponential in m). We describe one of the m phases. First, each beacon sends a message that reaches x' with probability 1/2 and does not reach x', but reaches all points that are closer to the beacon than x', with probability 1/2. As in the protocol for the point-to-point case, all messages that reach x' arrive at the same time, say t_0.

Consider the message sent by the beacon B_i. If there is no applicant whose distance from B_i is equal to the distance between x' and B_i, it follows that no applicant *knows* whether the message of B_i reaches x_i. It also follows that the colluding applicants, as a group, do not know which messages x' must combine and forward to all beacons. So, regardless of the times at which the applicants receive the messages sent by the beacons, any combination of messages they forward might be the wrong combination with probability $1/2^3$. By repeating the phase m times, the probability of forwarding the correct messages m times is $1/8^m$. This is only true if there are no applicants on the three circles centered at the beacons and passing through x'. If there are applicants on these three circles, then those applicants would know if a particular message reaches x' because they are at the same distance (separately) from the beacons. This situation is illustrated in Figure 4.

So, in order for a location to be simulated by faulty entities, there must be three entities, each of which resides on a circle going through the simulated point and centered at the beacons. The question we are interested in becomes the following: given a set of entities, how many points can be simulated by them? We give a loose upper bound on this number. Let k be the total number of corrupt entities in the system. A point x' can be simulated by these k entities if there are three entities A_1, A_2, and A_3 such that $\rho(x(A_i), x(B_i)) = \rho(x', x(B_i))$. In other words, x' is at the intersection of three circles centered around the beacons and each containing one of the k entities. The set of points that can be simulated by

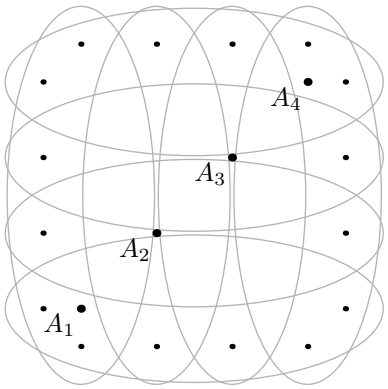

Fig. 5 Beacons around the perimeter of a square and k applicants that can appear to be anywhere.

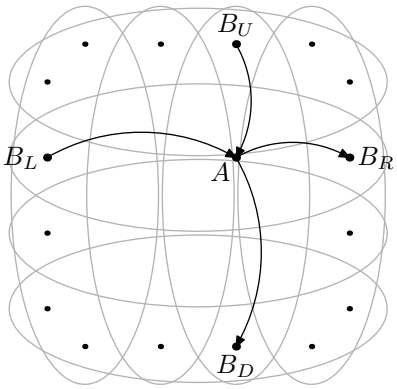

Fig. 6 Two-dimensional protocol: A must forward the combination of messages from B_U and B_L to both B_D and B_R.

the k entities are points that are at the intersection of such three circles. This set is a subset of the set of points that belong to the intersection of the sets of circles centered at B_1 and each containing an entity in the system and set the of the circles centered around B_2 and each containing an entity in the system. Those circles have at most $2k^2$ points in common (any two circles with different centers intersect at most in two points) and at most k^2 of these points are inside the triangle formed by the beacons.

In the foregoing discussion we assume that the bounded range broadcast can be made to reach only points whose distance to a beacon is strictly less than a given value. A more practical assumption would require that a message reaches no point that is more than $\epsilon + R$ from a beacon for some ϵ and some R. This will modify the result to the following: k entities cannot simulate points in more than k^2 disjoint small neighborhoods centered around the k^2 points defined by the intersection of circles, and where the area of a neighborhood is in $O(\epsilon^2)$.

8.3 Group-distinctness with a grid of beacons

In this section, we show how the result of Section 8.1 can be applied in a specific setting. We show how applicants can be severely limited in the number of identities they can simulate. We consider a system in which beacons are evenly spaced on the perimeter of a square as shown in Figure 5. There are $4k$ beacons, k beacons on each edge.

Without loss of generality, let k be the length of an edge of the square (in other words, beacons that are adjacent are at a distance 1 apart). Let B_{L_i}, B_{R_i}, B_{U_i} and B_{D_i} be the i-th beacons on the left edge, right edge, upper edge, and lower (down) edge respectively. We use a coordinate system with center at the top left corner of the circle with the coordinates of the bottom right corner being (k, k) and the horizontal axis being the x axis. For two beacons B_{L_i} and B_{R_i}, we can define a horizontal ellipse E_{H_i} with foci B_{L_i} and B_{R_i} and with diameter just under $\sqrt{k^2+1}$. Similarly we define a vertical ellipse E_{V_i} with foci B_{U_i} and B_{D_i} and with diameter just under $\sqrt{k^2+1}$. Every point inside the square belongs to at least one, but no more than two horizontal ellipses and at least one, but no more than two vertical ellipses. In fact, a horizontal band of width 1 around the line segment joining B_{L_i} and B_{R_i} is completely contained in E_{H_i}. A similar statement is true for vertical ellipses.

If we only consider vertical and horizontal ellipses, then k applicants are sufficient to *cover* all the ellipses. For example, place an applicant at (i, i) for each i. This is illustrated in Figure 5. The situation can be improved drastically, though, if we modify the protocol. In the modified protocol, after the apparent location of an applicant is determined, say (x, y), the applicant receives two nonces (random messages) from $B_{U_{\lfloor x/l \rfloor}}$ and $B_{L_{\lfloor y/l \rfloor}}$. These nonces arrive at the apparent location at the same time as in the protocol of Section 8.1 and the applicant is required to combine then and forward the combined message to $B_{D_{\lfloor x/l \rfloor}}$ and $B_{R_{\lfloor x/l \rfloor}}$ (Figure 6). In order for an applicant to successfully forward the messages on time, it has to be in the intersection of $E_{H_{\lfloor y/l \rfloor}}$ and $E_{V_{\lfloor x/l \rfloor}}$. It follows that to cover all the ellipses, we need $(k/2)^2$ applicants. Indeed, only adjacent ellipses have non empty intersections and we need one applicant to cover the intersection of two adjacent horizontal ellipses and two vertical adjacent ellipses. Since there are $k/2$ disjoint pairs of adjacent horizontal ellipses and an equal number of disjoint pairs of vertical adjacent ellipses, we need at least $(k/2)^2$ applicants to cover all of them.

With this result, we can define the following group distinctness test when the maximum number of faulty applicants is f and the total number of applicants is $m > f$. For every applicant, determine an intersection of a vertical ellipse and a horizontal ellipse in which the applicant is located. Each intersection would have a group of applicants, possibly empty. Let $m(f)$ be the number of applicants in the f largest groups (assuming there are at least f groups). Then, there must be at least $m - (m(f) - f)$ distinct applicants amongst the m applicants.

This result is interesting because it confines corrupt applicants to simulating points in a small region. It is a significant improvement over the introductory example, where we needed one beacon per corrupt applicant. In this example, the number of corrupt applicants needed to cover the whole space of interest is quadratic in the number of beacons.

9 Inaccuracies

In this section we very briefly describe a generalization of the problem that allows inaccuracies in measured distances. We only consider the simplest case, where measured distance may vary by a small constant fraction (known in advance) from the actual distance. This assumption may not be as restrictive as it appears, especially in view of our suggestion in Section 2.4 that each distance be measured more than once over a period of time to ensure accuracy (for example, if the measurements deviate from the true value according to a reasonable probability distribution, the smallest of the multiple measurements will generally tend to the true value fairly quickly).

9.1 Inaccurate distance measurements

To account for the inaccuracies in measuring the distance, we consider the variant of the problem where the measured distance $\mu(B_i, A)$ is allowed to exceed the actual reported distance by a small multiplicative factor: as long as there exists a point x' that satisfies $\rho(x(B_i), x') \leq \mu(B_i, A) \leq (1+\epsilon)\rho(x(B_i), x')$ for each i, the protocol should not reject the applicant.

The problem of validating an applicant becomes equivalent to identifying an intersection of thin (because we assume ϵ to be small relative to the measured distances) spherical shells (one around each of the beacons).

The goal of all our protocols is to distinguish between different applicants. Therefore a natural measure of how badly a protocol fails might be the smallest distance between points that cannot be reliably distinguished using the protocol. Now imagine a region E such that no two points in E can be reliably distinguished. Since E is an intersection of shells around beacons and the thickness of each shell is small compared to its radius, we may think of E as bounded by almost straight planar surfaces. Consider as an example the two dimensional case, and focus first on just two of the beacons, B_1 and B_2. The straight line segments from their locations to the applicant's location meet at an angle, say θ. If θ is close to the right angle, then the indistinguishability region is close to a square (actually, two disjoint squares, because two circles around B_1 and B_2 intersect in two points). If θ is very small, the indistinguishability region looks more like a thin parallelogram, and in such a case it can happen that two points a large distance apart cannot be distinguished. When the third beacon is included, it may still be the case that the indistinguishability region has a large diameter.

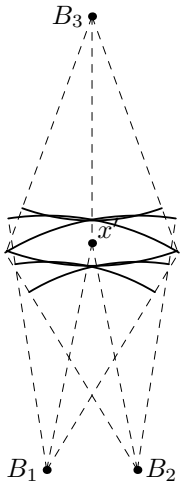

Fig. 7 The small angle case.

For this case, if the shell boundaries are replaced by straight lines, we see that the distance between the two points farthest apart in the indistinguishability region is at most $D = 2d\epsilon\frac{\cos(\theta/2)}{\sin\theta}$, where d is the distance between B_1 and x'. In other words, the diameter of the indistinguishability region may increase proportionally to $1/\sin\theta$.

9.2 Inaccurate clocks in beacons

Some of our protocols depend on quite accurate clocks. For example, in the proof of Theorem 4 we describe a protocol that requires beacons to send messages independently, but at precisely timed moments, in order to prevent several colluding points from simulating a nonexistent applicant. Clock inaccuracy in such a protocol translates directly into increased inaccuracy of distance measurement (because the beacon that sends the message is not the same as the beacon that measures the arrival time). A very similar problem is solved in the existing Global Positioning System (GPS) [1] using redundant information. A GPS receiver reads the timestamps in signals sent by several satellites to measure its distance from each, and given a table of ephemerides deduces its own geographical location. The satellite signals travel at the speed of light, and the Earth is small enough that even slightly inaccurate clocks may lead to inaccurate measurements. (Highly accurate clocks are built into the satellites, but are too expensive for mass-produced GPS receivers.) Therefore, instead of four signals (which would be enough to identify the receiver's location), five or more are used to allow for correction of clock drift.

10 Discussion of model

10.1 Adversary Model

The adversary model is particularly important for the accuracy of measured distances. In fact, a corrupt entity that is used to route messages between non-corrupt beacons can artificially increase the distance between them as well as between corrupt entities and beacons. Later, the calculated distance could be shortened which violates a main assumption of our model. We do not have a fully satisfactory solution to this problem, but we have two approaches to deal with it. The first approach applies to peer-to-peer systems that exhibit locality characteristics. In such systems, the distance between nodes is proportional to the actual network distance between the nodes. If the overlay network exhibits locality characteristics, we can calculate the network-distances between beacons directly without going through the overlay network and therefore without risking that the routing is compromised (assuming the routing on the underlying network cannot be easily compromised). These distances will be smaller than the distances on the overlay network, but one could then use solutions that tolerate inaccuracies in the measured distances. The second approach makes limiting assumptions on the disruption power of the adversary. If we assume that any two nodes are connected by a path that does not go through a corrupt node, then we can use multiple paths to calculate the distance between two nodes. The shortest among the calculated distances would be chosen as the distance between two nodes.

Another potential difficulty can be cause by corrupt nodes trying to flood the network with message in order to prevent accurate measurements of distances. Dealing with such denial of service attacks is beyond the scope of this paper.

10.2 Certificates

It is important to realize that the set of beacons in our model is not the same as a central certifying authority. In fact, all we need to assume about the beacons is that they are distinct and that a certain number of them are correct. In principle, a given entity can establish the distinctness of an initial set of beacons by using some of the resource-consuming challenge-response described in [10] without requiring any certifying authority. The assumption that a certain proportion of beacons chosen at random is correct is a system assumption for we cannot expect a system with an arbitrary number of faulty entities to be able to function.

Once an initial set of beacons is established, our results show that they can be used *remotely* to establish the distinctness of identities created by entities with unbounded computing power. This shows that the following lemma from [10] (his notation is different from ours, but should be clear from the context) does not hold once the geometric properties of communication are considered.

Lemma 3 *Lemma 4 [10] If the correct entities in set C do not coordinate time intervals during which they accept identities, and if local entity l accepts any identity vouched for by q accepted identities, then even a minimally capable faulty entity f can present $g = \lfloor |C|/q \rfloor$ distinct identities to l.*

This lemma basically says that accepted identities cannot be used to accept further entities. We showed that, if we take the geometric properties of communication into account, we can use accepted identities to accept additional entities. In fact, in one of our results we showed that a set of $d+1+2f$, at most f of which are faulty, can prevent one faulty entity e_{faulty} in their convex hull from presenting distinct identities even if e_{faulty} has unbounded resources. This result is achieved without assuming a central authority.

In practice, the beacons can be certified by a central certifying authority to bootstrap the system. Once a set of beacons is certified, it can be used to provide certificates remotely. In that case, an applicant that wants to obtain a certificate from the set of beacons would identify beacons that have valid public certificates obtained from the central authority. Then, the applicant can initiate a geometric certificate request which will result in the beacons probing the applicant as explained in the various protocols we presented. These probes will be started by multiple beacons to obtain the distances as required by the protocols. At the end of the probing period, the beacons will present the applicant with pieces of the geometric certificate (distances from beacon to applicant or location of applicant as calculated by a beacon) that the applicant can put together to obtain the geometric certificate.

10.3 Limitation of distinctness tests

Some of the distinctness tests we presented assume that the entity under consideration is in the convex hull of the beacons in the system. If an entity is outside the convex hull of the beacons, then some of the theorems we prove do not hold. It is reasonable to question whether the convex hull condition is only of theoretical interest and if anything can be done if an entity is not in the convex hull of the beacons. The answer to the first part of the question would depend on the actual network under consideration. We have answered the second part when we presented group-distinctness tests for various scenarios and under different system assumptions. We believe that more work is needed in this direction to further generalize the results.

10.4 Accuracy of measured distances

Ng and Zhang [19] show that on the Internet the roundtrip delays can be used to measure distances between entities if enough measurements are taken and the minimum amongst the measured delay is used as the distance measure. These measurements were done using ICMP ping messages. In our model, communication is done in an overlay network that does not necessarily exhibit the same delay characteristics as those of the Internet. Nonetheless, we can expect that in periods of low congestion, the distances will reflect the underlying network distances. In our work, establishing a geometric certificate can be done over a period of time and multiple measurements can be taken and the smallest times be included in the certificates. If the participants belong to common congestion zones (which can be correlated with time zones), then we can expect that the minimal delay measured by participants will exhibit metric characteristics. Nevertheless, studying delay characteristics in Internet-based overlay networks is a subject that needs further study and our work is based on the assumption that these characteristics are similar to those of the Internet.

11 Conclusion

We have shown that it is possible to exploit the geometric properties of message transmission delay in order to reduce the effects of Sybil attacks. We believe that a lot more work is still needed to make this work of more practical value. In particular, we believe that extensions of the protocols in Section 8 can have a good chance of leading to solutions that can be used in practice.

Acknowledgements We would like to thank Roger Wattenhofer for pointing out the work of Čapkun and Hubaux [7].

References

1. Agarwal, N., Basch, J., Beckmann, P., Bharti, P., Bloebaum, S., Casadei, S., Chou, A., Enge, P., Fong, W., Hathi, N., Mann, W., Sahai, A., Stone, J., Tsitsiklis, J., Roy, B.V.: Algorithms for GPS operation indoors and downtown. GPS Solutions **6**, 149–160 (2002)
2. Amaldi, E., Kann, V.: The complexity and approximability of finding maximum feasible subsets of linear relations. Theoretical Computer Science **147**, 181–210 (1995)
3. Bern, M., Eppstein, D.: Approximation algorithms for geometric problems. In: Approximation algorithms for NP-hard problems, pp. 396–345. PWS (1997)
4. Blumenthal, L.: Theory and applications of distance geometry. Clarendon Press (1953)
5. Brönnimann, H.: Derandomization of geometric algorithms. Ph.D. thesis, Princeton University (1995)
6. Brönnimann, H., Goodrich, M.: Almost optimal set covers in finite VC dimension. In: Proceedings of the 10th annual ACM Symposium on Computational Geometry, pp. 292–302 (1994)
7. Čapkun, S., Hubaux, J.P.: Secure positioning of wireless devices with applications to sensor networks. In: Proceedings of INFOCOM (2005)
8. Čapkun, S., Hubaux, J.P.: Secure positioning in wireless networks. IEEE J. on Selected Areas in Communications **24**(2), 221–232 (2006)
9. Deza, M., Laurent, M.: Geometry of cuts and metrics. Springer (1997)
10. Douceur, J.: The Sybil attack. In: Proc. of IPTPS, pp. 251–260 (2002)
11. Feige, U.: A threshold of $\ln n$ for approximating set cover. J. ACM **45**, 634–652 (1998)
12. Hochbaum, D.S.: Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. In: Approximation algorithms for NP-hard problems, pp. 94–143. PWS (1997)
13. Hu, Y.C., Perrig, A., Johnson, D.B.: Packet leashes: a defense against wormhole attacks in wireless networks. In: Proceedings of INFOCOM (2003)
14. Kleinberg, J., Slivkins, A., Wexler, T.: Triangulation and embedding using small sets of beacons. In: Proc. of the IEEE FOCS, pp. 444–453 (2004)
15. Lazos, L., Poovendran, R.: SeRLoc: secure range-independent localization for wireless networks. In: Proceedings of WISE 2004 (2004)
16. Luo, J., Shukla, H.V., Hubaux, J.P.: Non-interactive location surveying for sensor networks with mobility-differentiated ToA
17. Matoušek, J., Seidel, R., Welzl, E.: How to net a lot with little: small ϵ-nets for disks and halfspaces. In: Proceedings of the 6th annual ACM Symposium on Computational Geometry, pp. 16–22 (1990)
18. Newsome, J., Shi, E., Song, D., Perrig, A.: The Sybil attack in sensor networks: analysis and defenses. In: Proc. of IPSN (2004)
19. Ng, T., Zhang, H.: Predicting Internet network distance with coordinates-based approaches. In: Proc. of INFOCOM (2002)
20. Parno, B., Perrig, A., Gligor, V.: Distributed detection of node replication attacks in sensor networks. In: Proceedings of the IEEE Symposium on security and privacy, pp. 49–63 (2005)
21. Sastry, N., Shankar, U., Wagner, D.: Secure verification of location claims. In: Proc. of ACM WiSe (2003)
22. Waters, B.R., Felten, E.W.: Secure, private proofs of location. Tech. Rep. TR-667-03, Princeton (2003)