An Overlay Network for Forwarding Symbolically Addressed Geocast Messages

Frank Dürr, Kurt Rothermel
Institute of Parallel and Distributed Systems, University of Stuttgart
Universitätsstraße 38, 70569 Stuttgart, Germany
phone: +49-711-7816-431 fax: +49-711-7816-424
{frank.duerr,kurt.rothermel}@informatik.uni-stuttgart.de

Abstract— **Geocast, which allows for forwarding messages to hosts residing at specified geographic areas, is a promising communication paradigm with a wide range of applications. Geocast target areas can be specified either by geometric figures or symbolic addresses, such as `/usa/fl/miami/market-street`.**

In this paper, we present a novel geocast routing protocol for symbolically addressed messages. Compared to geocast protocols based on geometric information, our protocol can operate on simple symbolic location models, and message forwarding does not require costly geometric operations. The proposed protocol is based on an overlay network that is mapped to an IP-based network infrastructure. The overlay network is structured in a hierarchical fashion, to ensure a scalable global geocast service supporting also large target areas. Although our protocol does not rely on a layer 3 multicast protocol, we also show how to improve the performance of message forwarding by integrating a light-weight layer 3 multicast protocol. Our evaluations of the protocol underline the scalability of our approach and show good routing quality leading to short message paths.

I. INTRODUCTION

With geocast, messages can be sent to all mobile and stationary hosts currently located in a geographic target area. The availability of wireless communication technologies and various indoor and outdoor positioning systems enable a wide spectrum of promising geocast applications like disaster warning, location-based information services, or location-based advertising. In these different applications, the size of the target areas may vary greatly, like from a state, town, or street to a floor or room of a building.

Geocast messages can be addressed either by geometric figures like polygons, or by symbolic names like city names, room numbers, etc. A simple example of a symbolic address is `/usa/fl/miami`, which denotes the city of Miami in the state of Florida in the USA. Although geometric figures are very flexible and can describe arbitrary areas, symbolic addressing also has several advantages over geometric addressing. With symbolic addressing, target areas can be specified by addresses similar to those people are using in everyday life. Therefore, symbolic addresses are more intuitive to use than geometric ones. Also people and applications tend to address messages to real-world locations, such as rooms or buildings, rather than to abstract spatial areas in a geometric coordinate system. Consequently, symbolic addressing is an important alternative to geometric addressing.

Even if symbolic addresses are used at the geocast service interface, routing can still be based on geometric addressing if symbolic addresses are mapped onto geometric ones. However, to be able to perform this mapping, we need a complex location model including geometric descriptions of *every* symbolically addressable location. For example, to address areas within a building, a three-dimensional geometric model including geometric representations of every room, floor, etc. would be required. Moreover, if geocast routing is based on geometric information, forwarding decisions require the comparison of geometric figures, which may be rather costly operations, even if approximated geometries as proposed in [1] are used. If geocast routing is based on symbolic addresses instead, the modeling effort is reduced to a minimum as location models can be represented by hierarchical name spaces without any need for geometric representation. Such name spaces can be mapped onto binary addresses that require only simple prefix matching operations for message forwarding rather than costly geometric operations.

In this paper, we propose a novel geocast routing protocol purely based on symbolic addresses. The proposed protocol provides a global geocast service for any size of target area. Since this protocol is implemented in an overlay network being mapped onto an IP-based underlay network, it can be deployed gradually and without modifying existing IP network infrastructures. However, we also show how this overlay approach can be combined with a light-weight layer 3 multicast protocol to increase efficiency. Our evaluations show that load is distributed evenly among the overlay routers. Moreover, the protocol achieves short message paths.

The remainder of this paper is structured as follows. In Sec. II, we present related approaches. Then, we introduce our system model in Sec. III. Afterwards, we introduce our symbolic location model and addressing scheme used to address geocast messages in Sec. IV. In Sec. V we present our overlay network architecture. The message forwarding algorithm is described in Sec. VI. In Sec. VII, we present the evaluation of our approach, before the paper is concluded with a short summary and outlook on future work in Sec. VIII.

II. RELATED WORK

The geocast approaches that we consider to be closest to our work are the hierarchical geocast routing algorithms described

Published in Proceedings of the 15th International Conference on Computer Communications and Networks (ICCCN '06), pages 426-434, Arlington, VA, October 2006.

http://dx.doi.org/10.1109/ICCCN.2006.286314

in [2], [3], and our approach proposed in [4]. These algorithms forward geocast messages along a hierarchy of dedicated geocast routers that are responsible for forwarding messages to geographic areas. Forwarding decisions are made based on comparisons of symbolic or geometric target and service areas. In contrast to these approaches, we map target areas to binary addresses rather than using the geographic description to make forwarding decisions. These addresses are much smaller than the original target area descriptions resulting in less message overhead. Moreover, binary addresses can be compared very efficiently by routers using simple prefix matching algorithms. Furthermore, although our approach does not rely on a layer 3 multicast protocol, we show how to integrate a light-weight multicast protocol to increase efficiency. Additionally, we propose a heuristics to improve hierarchical routing by adding shortcuts to the routing hierarchy.

Similar to our approach, the multicast-based geocast routing protocols proposed in [5] map locations onto binary addresses. In particular, each location is assigned an individual multicast address, and an IP multicast protocol, such as Multicast Open Shortest Path First (MOSPF [6]) or the Distance Vector Multicast Routing Protocol (DVMRP [7]), is used to forward messages to the corresponding location. Since multiple a priori unknown senders may send messages to a given location, either a source-based tree or shared tree protocol can be applied, which both have their limitations. A source-based tree approach may cause significant overhead if messages are sent sporadically—like event notifications—to locations. In the worst case, each message may cause a new tree to be established in the network. While shared trees alleviate this problem, they are not optimal for all senders and add extra complexity for managing rendezvous points. In contrast to that, our geocast approach uses an overlay network for forwarding messages to locations rather than relying on the existence of an IP multicast infrastructure. Of course, this level of independence does not come for free. However, we will propose an optimization of our approach which sends the first messages of a sender over the overlay and then switches to a Source-Specific Multicast (SSM [8]) protocol. With this optimization, an individual SSM channel is set up for a sender forwarding a bulk of messages to a location. Due to its source specific nature SSM avoids the problems associated with shared trees, and the combination of our overlay network and SSM reduces the overhead to set up source-based trees.

In [9], a directory stores the areas covered by subnetworks, so that it can be queried for the addresses of the subnetworks overlapping the target area of a geocast message. This approach only scales up to a small number of subnetworks per target area if messages are sent to the subnetworks via unicast. Using multicast instead leads to an approach similar to the multicast-based approach mentioned above.

It is important to stress that our approach relies on an infrastructure. In contrast to that, geocast approaches for mobile ad hoc networks like the ones described in [10] are based on fundamentally different assumptions. In contrast to these approaches, which typically operate in limited geographic domains, our approach is tailored to the implementation of a *global* geocast service, which is only feasible using a routing infrastructure.

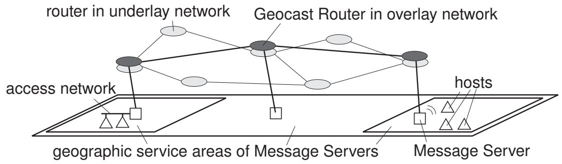

Fig. 1. System model

III. SYSTEM MODEL

The three components of our architecture are hosts, message servers, and routers (cf. Fig. 1):

Geocast Hosts, which can be mobile or fixed, are the recipients and senders of geocast messages. If a message is sent to a given target area, the message has to be delivered to those hosts that currently reside in the target area. We assume that hosts are able to determine their geographic position. For stationary hosts, this position can be configured statically; mobile devices need some positioning system to determine their position.

Geocast Message Servers (GMS) are responsible for distributing geocast messages to hosts within a certain access network. A GMS has a geographic *service area* whose size is equal to the area covered by the access network. Such a service area can be as small as a single floor of a building on which a wireless LAN is installed or as large as a radio cell covering a whole city district. Since the same geographic area may be covered by different access networks, the service areas of GMSs may overlap. Every party that wants to be able to distribute geocast messages to the hosts in their access network must set up a GMS with an appropriate service area and register this GMS with the Geocast Routers of the overlay network as described later in Sec. VI-A.

Geocast Routers (GR) are responsible for forwarding geocast messages from the sender to the GMSs whose service areas overlap with the target area of the message, so that these GMSs can further distribute the message within their access networks. GRs are arranged in an overlay network and exchange messages using the UDP service offered by the underlying IP-based Internet infrastructure. This paper is focused on a best-effort geocast service. Therefore, we do not deal with lost messages and other reliability aspects like message reordering.

In this paper, we focus on the forwarding of messages between access networks through an overlay network of GRs. The efficient local distribution of messages within an access network by GMSs is beyond the scope of this paper. A GMS may simply broadcast a received message within its access network and hosts filter messages according to their current location. A more efficient approach could utilize multicast to address only hosts in certain geographic parts of an access network.

IV. SYMBOLIC ADDRESSING SCHEME

Our approach is based on a hierarchical symbolic location model similar to the ones described in [11] and [12]. The location hierarchy consists of a set of locations L and is structured according to the spatial containment relationship. For two locations l_1 and l_2 it holds $l_1 < l_2$, if l_2 spatially contains l_1. l_1 is called a *descendant* of location l_2, and l_2 is an *ancestor* of l_1. A direct descendant of location l is called a *child location* and a direct ancestor a *parent location*. $\mathrm{ancestors}(l)$, $\mathrm{children}(l)$ and $\mathrm{parent}(l)$ denote the set of ancestor locations, child locations, and the parent location, respectively. Each symbolic location has a unique *symbolic name*, which is constructed by concatenating the (relative) names on the path from the root of the hierarchy to the location. For instance, New York City in New York State in the USA has the symbolic name `/us/ny/newyork`. Target areas may be any location in L identified by its symbolic name. Furthermore, locations of the location model are used to define host positions, and service areas of GRs and GMSs.

We map each location l onto a binary *location address*. These addresses are shorter than symbolic names leading to less overhead in the message and they can be compared very efficiently. This also allows us to use different naming schemes where the same location can have different names that are mapped onto the same address. Locations are mapped such that the address of location l_2 is a prefix of the address of l_1 iff $l_1 \leq l_2$. That means, the prefix property of location addresses reflects the spatial containment relationship between locations. A benefit of using an overlay network is that our location address space is not restricted by the length limitation of IP addresses. Therefore, we can assign a unique address to each location even if a huge, global location model is used. We assume that location addresses are assigned manually, possibly coordinated by national and international organizations similar to the International Corporation for Assigned Names and Numbers (ICANN). Additionally, we assume that the mappings from symbolic location names to location addresses are stored in a directory similar to today's Domain Name System (DNS) or some directory based on distributed hash tables like Overlook [13].

V. OVERLAY NETWORK ARCHITECTURE

In this section, we describe the architecture and construction of the overlay network. First, we define the nodes of the overlay and then we describe the establishment of links between nodes.

A. Overlay Network Nodes

GRs constitute the nodes of the overlay network. Each location l is associated with one *designated Geocast Router*, say r. l is called the *service area* of r, denoted by $\mathrm{s}(r) = l$.

A designated GR of a location is set up by configuring the name of its service area manually. The GR then determines the associated location address by querying the directory mentioned above.

A designated GR with service area l is also designated GR of each location $l' < l$ if no other GR has been set up for some location l'' with $l' \leq l'' < l$. For instance, a GR that is designated GR of a city is also designated router of a city district of this city as long as no city district GR has been set up.

B. Overlay Network Links

Next, we describe how the links between GRs in the overlay network are established. Since the overlay can consist of a huge number of GRs, it is not feasible that a GR knows all other GRs. Moreover, establishing links to many other GRs would lead to large forwarding tables. Therefore, we have to create links in the overlay carefully to allow for short message paths on the one hand and small forwarding tables on the other hand.

The overlay network basically resembles the location hierarchy. In detail, GR r with service area $\mathrm{s}(r)$ has a *minimum topological knowledge* of the overlay network that includes links to the following set of GRs:

TK1) The designated GR of $\mathrm{parent}(\mathrm{s}(r))$. In Fig. 2, r_5 has a link to r_4.

TK2) Designated GRs of locations in $\mathrm{children}(\mathrm{s}(r))$. In Fig. 2, r_5 has a link to r_6.

With this minimum topological knowledge, messages are forwarded along the hierarchy of GRs. However, with more topological knowledge of further GRs, messages can take *shortcuts* to distant GRs, i.e., messages can be sent directly to non-parent and non-child GRs via the underlay network. This reduces the load of by-passed GRs and leads to shorter message paths. In order to keep routing tables small and reduce the amount of data that has to be exchanged between GRs to set up routing tables, we only include shortcuts to certain GRs. Like [14], we assume that the relevance of location-specific information decreases as distance increases. Therefore, we assume that most senders are located geographically close to the addressed target area. Based on this assumption, our heuristics adds more shortcuts to nearby areas than to distant areas. In detail, the *extended topological knowledge* of GR r with service area $\mathrm{s}(r)$ includes the following GRs:

TK3) Designated GRs of locations in $\mathrm{ancestors}(\mathrm{s}(r))$ and of child locations of r's ancestor locations (locations in $\{l \in L \mid \exists l' \in L : l' \in \mathrm{ancestors}(\mathrm{s}(r)) \wedge l \in \mathrm{children}(l')\}$). For instance, a city GR of city c knows the earth GR, the country GRs, and the state GRs in which c is located, as well as all city GRs of the state in which c is located, all state GRs of the country in which c is located, etc. In Fig. 2, r_5 has shortcuts to r_1, r_2, and r_3, but not to r_7, r_8, and r_9.

This leads to a forwarding table size of $O(dc)$, where d is the depth of the location hierarchy and c is the maximum number of child locations. For a balanced hierarchical location model with n locations, d grows with $O(\log n)$. If we consider typical examples of location hierarchies, d is rarely greater than 10. For instance, a global location hierarchy consisting of the earth, countries, states, cities, streets, buildings, floors, and rooms would lead to $d = 8$. Giving an estimate for c

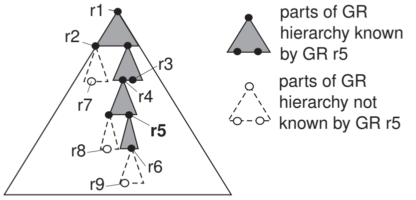

Fig. 2. Topological knowledge of overlay router r_5

is more difficult. For instance, the number of states within a country is rarely much greater than 50, whereas the number of streets per city can exceed 1000 for large cities. However, big cities will very likely be partitioned into districts, which automatically reduces the number of child locations of the city location. To reduce the number of forwarding table entries, we can also insert artificial locations into the location model to decrease the number of child locations. Therefore, we expect a GR to have at most a few 1000 forwarding table entries.

The links of the overlay network are established top down. For example, first the links from the earth router to the country GRs are established, then links from country GRs to state GRs, etc. In order to join the overlay network, a GR, say r_{new}, must know some other bootstrap GR, $r_{\text{bootstrap}}$, that is already part of the overlay network. This GR can either be configured manually (e.g., the earth or country GRs should be well known), or a mechanism like expanding ring search can be used in the underlay network to find this GR. Consider for instance the New York City GR (r_{new}) that wants to join the overlay network. The following steps are executed to integrate r_{new} into the overlay network:

1) r_{new} sends a “parent discovery” message containing its own UDP address and service area address $\text{addr}(\text{s}(r_{\text{new}}))$ to $r_{\text{bootstrap}}$ (Fig. 3 ❶).
2) The GRs in the overlay network forward this message as described in detail in Sec. VI to the GR, r_p, whose service area address $\text{addr}(\text{s}(r_{\text{p}}))$ is the longest prefix of $\text{addr}(\text{s}(r_{\text{new}}))$ of all GRs that are already part of the overlay network (❷). Consequently, r_p is the GR with the smallest service area that contains r_{new}’s service area. In the example, this is the New York state GR, whose service area address is a longer prefix of the New York City location address than the addresses of the USA or earth location.
3) When r_p receives a “parent discovery” message, it sends all entries of its forwarding table and its own UDP address and service area address to r_{new} (❸).
4) r_{new} copies these entries to its own forwarding table. Note that according to the definition of extended topological knowledge, r_{new}’s forwarding table is a superset of the table of its parent router. Therefore, r_{new} can establish shortcuts to routers in TK3 (see above) without contacting these routers itself by copying r_p’s table. Additionally, r_{new} adds an entry $[\text{addr}(\text{s}(r_p)) \mapsto$ UDP addr. of $r_p]$ to establish a new child-to-parent link.
5) r_p adds an entry $[\text{addr}(\text{s}(r_{\text{new}})) \mapsto$ UDP addr. of $r_{\text{new}}]$ to its forwarding table to establish the new parent-to-child link.

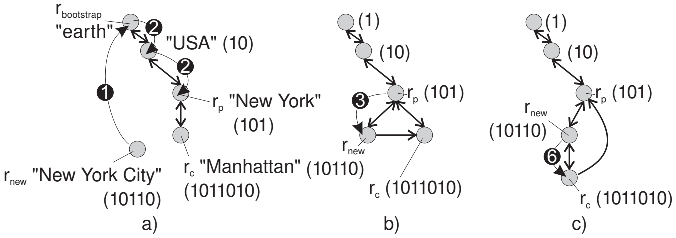

Fig. 3. Steps of overlay network construction. Only parent-child links and all links between r_p, r_c, and r_{new} are shown for clarity. Location addresses of service areas are given in parenthesis.

After the steps 1–5, r_{new} is integrated into the overlay network as shown in Fig. 3b, but there might be designated GRs of child locations of $\text{s}(r_{\text{new}})$ that miss a child-to-parent link to r_{new}. In the example, this situation arises if the Manhattan GR has joined the overlay network *before* the New York City GR (see GR r_c in Fig. 3b). r_{new} can detect this situation by checking the forwarding table received from r_{p}. Every entry $[\text{addr}(\text{s}(r_c)) \mapsto$ UDP addr. of $r_c]$, where $\text{addr}(\text{s}(r_{\text{new}}))$ is a prefix of $\text{addr}(\text{s}(r_c))$ identifies a GR that is designated GR of a child location of $\text{s}(r_{\text{new}})$. In the example the Manhattan GR is such a GR. To establish the missing link, r_{new} sends a “new parent” message to r_c (❻) that contains its service area address and UDP address. On receiving this message, r_c inserts an entry $[\text{addr}(\text{s}(r_{\text{new}})) \mapsto$ UDP addr. of $r_{\text{new}}]$ into its forwarding table thereby establishing the missing link. Note that r_{new} already has an entry $[\text{addr}(\text{s}(r_c)) \mapsto$ UDP addr. of $r_c]$ since this entry is part of the copied forwarding table of r_p. Finally, r_p discards all entries $[\text{addr}(\text{s}(r_c)) \mapsto$ UDP addr. of $r_c]$, where $\text{addr}(\text{s}(r_{\text{new}}))$ is a prefix of $\text{addr}(\text{s}(r_c))$ except for the entry $[\text{addr}(\text{s}(r_{\text{new}})) \mapsto$ UDP addr. of $r_{\text{new}}]$ in order to keep its forwarding table small. Remember that a GR only needs links to designated GRs of child locations but not to GRs of grandchild locations. In the example, the New York state GR discards the link to the Manhattan GR since the New York City location contains the Manhattan location. Figure 3c shows the final overlay network of the example.

To keep forwarding tables up-to-date, GRs send forwarding table updates to child GRs and they propagate changes down the GR hierarchy. Periodic heartbeat messages between child and parent GRs are used to detected failed GRs. If a parent router notices a failed child router, it simply removes the child router from its routing table an sends a routing table update to all other children. If a child GR notices a failed parent router, it re-registers at an ancestor router that becomes its new parent router using the bootstrap mechanism described above. Additionally, the child router sends an update message to its children to tell them about the failed router and the now extended service area of its new parent router.

VI. MESSAGE FORWARDING

Our approach uses three phases to forward messages from the sending host to all hosts in the target area, t, of the message. In phase 1, the message is forwarded to the designated GR, r_t, of the target area. For instance, a message to New York City is forwarded to the New York City router. In this phase, the message may be sent via shortcuts to by-pass routers of the hierarchy. In the second phase, the message is distributed among all routers in the target area by forwarding it down the router hierarchy starting at r_t. That means, the New York City router forwards the message to all borough routers like the Manhattan router. Then, the borough routers forward the message to city district routers, etc. During the second phase, the routers within the target area also forward the message to the GMSs whose access networks cover parts of the target area. In the third phase, these GMSs finally forward the message to the hosts in their access networks that are located in the target area.

A. Forwarding to Access Networks in Target Area

Algorithm 1 shows the forwarding algorithm that each GR r executes on receiving a message from another GR or the sending host. To find out the current phase of message forwarding, r first checks a flag in the message header (lines 2 and 13). If this flag signals phase 2, then r forwards the message to all child routers in the target area (line 15).

Otherwise, r determines the next hop router, r_{next}, by searching for the router whose service area address is the longest prefix of the target area address (line 3, S denotes the service areas of all routers in r's topological knowledge). If r finds itself to be the router with the longest prefix address, then r is the designated router of the target area that starts phase 2 (line 4–7). If r finds another router, message forwarding is still in phase 1, and r forwards the message to r_{next} that is closer to the designated router. r_{next} may be a router reached via a shortcut.

Consider for instance a message to Manhattan reaching the Bavaria router r_3 in Fig. 4. The router with the longest prefix address in r_3's architectural knowledge is the USA router that can be reached via a shortcut. When later the Manhattan router, r_7, searches the router with the longest prefix, it finds itself and starts phase 2. The same holds, if no physical Manhattan router exists and the New York City router is therefore also responsible for Manhattan. Note that in this case, the New York City router has to ensure that the child router's service area is actually located in the target area Manhattan (line 15). For instance, the message to Manhattan must not be forwarded to the Staten Island router.

In order to forward messages to GMSs in phase 2 (line 16), we assume that each GMS gms registers with each router r with $\mathrm{s}(r) \leq \mathrm{s}(gms)$ by sending a register message via geocast to the target area $\mathrm{s}(gms)$. For instance, a GMS associated with an access network covering Manhattan registers with the Manhattan router, every street router in Manhattan, etc. In order to prevent a GMS from receiving the same message multiple times in phase 2 from different routers, only router r with $\mathrm{s}(r) = \mathrm{s}(gms) \cap t$ actually delivers a message targeted at t to gms. Note that the intersection $\mathrm{s}(gms) \cap t$ can be calculated efficiently by simply chosing the shorter location address of $\mathrm{s}(gms)$ and t. For instance, a message to New York City is delivered once by the Manhattan router to a GMS covering Manhattan rather than multiple times by street or building routers in Manhattan.

```
   On receiving message $m$ with target area $t$ do
2    if $\mathrm{phase}(m) = 1$ then
       $r_{\text{next}} \leftarrow$ longest_prefix_match($t$, $S$)
4      if $r_{\text{next}} = r$ then
         // $r$ is the designated router of the target area.
6        // $r$ starts phase 2.
         $\mathrm{phase}(m) \leftarrow 2$
8      else
         // phase 1: forwarding to target area
10       forward message to router $r_{\text{next}}$
       fi
12   fi
     if $\mathrm{phase}(m) = 2$ then
14     // phase 2: distribution within target area
       forward $m$ to each child router $r_c$ with $\mathrm{s}(r_c) \leq t$
16     forward $m$ to each GMS $gms$ with $\mathrm{s}(r) = \mathrm{s}(gms) \cap t$
     fi
```

Algorithm 1. Forwarding algorithm executed by GR r

B. Optimized Message Forwarding

The overlay network routing algorithm presented in the previous subsection does not rely on a layer 3 multicast protocol, and thus can be deployed without modification of the existing layer 3 infrastructure. However, it is clear that an overlay network approach can hardly distribute messages as bandwidth-efficient as a layer 3 approach (cf. Sec. VII-B). Therefore, we present now how our overlay network approach can be integrated with a layer 3 multicast protocol to optimize message forwarding.

The basic idea is to start delivering messages to GMSs via the overlay network and then switch to layer 3 multicast. Since switching to a layer 3 multicast tree introduces additional overhead, such an optimization is especially useful if several messages are sent frequently to the same target area rather than only single messages that are sent sporadically. Therefore, we distinguish between two modes of geocast message forwarding that are offered to applications on a host. The *message mode* is tailored to communication patterns where applications sporadically send messages to frequently changing target areas, e.g., the transfer of short textual warning messages to different

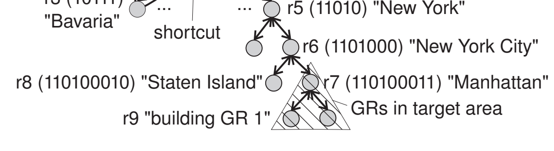

Fig. 4. GR hierarchy (only shortcut links of r_3 are shown)

endangered areas. In this mode, messages are forwarded solely via the overlay network without optimization. The *streaming mode* is for scenarios where applications send streams of messages over a longer period of time to certain target areas, e.g., periodical advertisements including images and product descriptions that are sent from a company's host to potential customers in a shopping mall. The application explicitly requests to send a stream of messages to a target area and also signals when it has no more messages to send. In streaming mode, the protocol establishes a multicast tree that exists as long as the stream exist, i.e., as long as applications intend to send messages from the host to the target area.

A geocast stream can be identified by a (source,target area) pair, where "source" is the sending host. In order to get an efficient layer 3 multicast trees, the utilized layer 3 protocol should create a source-based tree, where the sending host is the root and the GMSs in the target area are the leaves. The class of source-specific multicast (SSM [15]) protocols is well-suited for our requirements. SSM defines so-called channels (S, G), which are identified by unicast source address S and multicast address G. In our context, S is the sending host and G identifies the addressed target area. A SSM protocol like Protocol Independent Multicast (PIM) SSM (a subset of the PIM Sparse Mode protocol [8]) can build shortest path trees (SPT) rooted at S that can be used for efficient message transfer from S to all GMSs in the target area.

In detail, a SPT is built as follows. Sender S starts sending geocast messages to target area t via the overlay network (Fig. 5 ❶). In the header of the message, S sets a "streaming mode bit" that tells the receiving GMSs to switch to a SPT tree. Additionally, S maps t to a multicast address, say G, and includes G in the header. The only requirement for this mapping is that if S sends messages to different target areas *at the same time* then these areas must be mapped to different multicast addresses. Note that other senders can re-use the same multicast address for different target areas. Therefore, even a restricted layer 3 multicast address space is no problem in contrast to the multicast-based geocast approaches described in Sec. II where locations are mapped to globally unique multicast addresses. When GMS, gms, receives the message, it tells its local designated PIM-SSM router to join channel (S, G) (❷). In PIM-SSM, a SPT for channel (S, G) is built by sending join messages from designated routers towards S. PIM-SSM routers traversed by this message become part of the SPT. When the first GMS has joined the channel, S starts sending messages via the overlay network *and* via the SPT (❸). gms now receives the same messages via the overlay and the SPT. When gms starts receiving messages via the SPT, it sends an "ACK" message including sender S and target area t to the overlay GR from which it also receives these messages (❹). The GRs aggregate ACKs of all downstream GMSs and GRs and forward the aggregated ACKs upstream towards the sender. When all GMSs receive the message via the SPT, S finally receives the aggregated ACK for all GMSs in the target area from its GR and stops sending messages via the overlay network. Now, messages are sent solely via the SPT to the target area (❺).

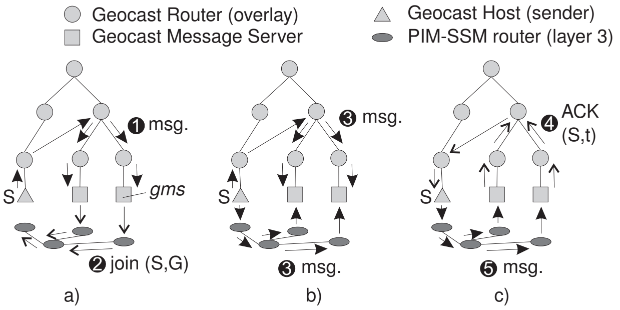

Fig. 5. Optimization using Source-Specific Multicast. a) Sender S sends message to GMSs in target area via overlay. GMSs join shortest path tree (SPT). b) Message is delivered via overlay and SPT. c) S stops using overlay.

VII. EVALUATION

In this section, we evaluate our approach by simulating different scenarios with an implementation of our algorithm for the network simulator ns-2.

A. Simulation Set Up

The underlay network topology for our evaluation consists of a real network topology from Liljenstam, Liu, and Nicol [16]. The whole topology consists of the backbones of 8 major Internet service providers in the USA. The topology models routers and their links. Moreover, routers are mapped to geographic positions (latitude, longitude) and symbolic addresses of the form country/state/city. To get a manageable network topology for our simulations, we have selected 2942 underlay routers for our simulation.

The overlay network consists of country, state, city, and city district GRs. These GRs are placed at underlay network routers. Since an algorithm for automatic placement is beyond the scope of this paper, we use a simple heuristics for GR placement: A designated GR of a certain geographic area is placed at an underlay network router located in this area. For instance, we place the New York City GR at an underlay network router with the geographic position `/US/NY/NEW_YORK`.

GMSs are placed at each underlay network router with exactly one link. We assume that each GMS is responsible for a different access network covering a city district.[1] In the evaluation, we consider the forwarding of messages from senders to GMSs, i.e., to access networks in the target area.

B. Path Quality

In this subsection, we evaluate the path quality achieved by our routing algorithm. As metric we use the *stretch factor*. The stretch factor denotes the factor by which the achieved underlay network path length is longer than the optimal path length in the underlay network. A stretch factor of 2 for instance means that the path is twice as long as the

---

[1] Since the geographic mapping of the underlay network only has city granularity, modeled city districts do not correspond to real city districts.

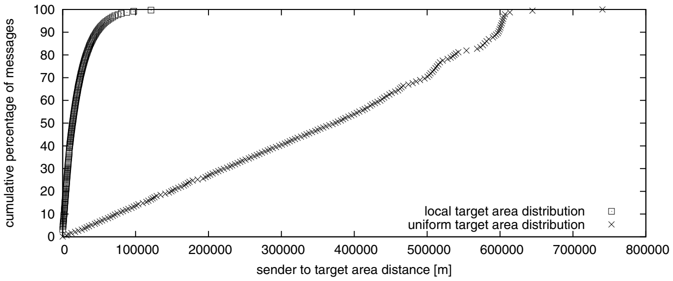

Fig. 6. Cumulative distribution of sender to target area distance.

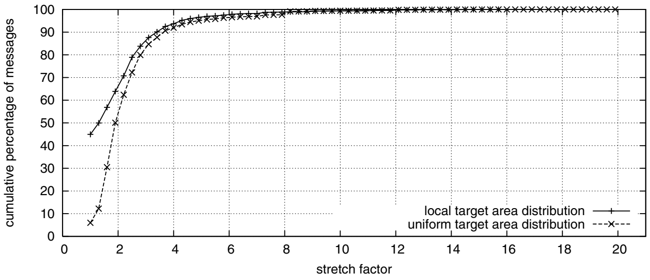

Fig. 7. Cumulative stretch factor distribution: small target areas (city districts)

optimal path, and therefore twice as much message copies (and bandwidth) are required as in the optimal case. The optimum is defined by single source shortest path trees (SPT). The source of a SPT is the sender of the message. Trees are pruned such that they only contain branches leading to GMSs in the target area. The calculated SPTs are similar to source-based trees calculated by layer 3 multicast protocols like MOSPF or PIM-SSM. Therefore, using a SPT as reference gives a realistic comparison to "native" multicast-based geocast approaches operating on layer 3 rather than on an overlay. At the same time, this shows the the gain we can achieve by using the multicast optimization (SSM) presented in Sec. VI-B.

We evaluate two scenarios. In the first scenario, we address small target areas at the size of city districts, i.e., at the leaf level of the GR hierarchy. Each simulation run consists of 10,000 messages. For each message, a city district at which the sender is located is selected randomly. Then a target area is selected. To analyze the influence of our assumption that most senders address nearby areas, we consider two sender to target area distance distributions. For each message, first a sender to target distance according to the distribution function is calculated. Then, the target area is selected from the set of city districts whose distance is closest to the desired distance. For the *local target area distribution* we assume that messages have strong locality properties, i.e., the sender mostly addresses target areas in his vicinity. Figure 6 shows the distribution of sender to target area distances for the local target area distribution. This distribution follows an exponential function that heavily favors short-range messages; about 90% of all messages are sent to target areas that are at most 50 km away from the sender. For the *uniform target area distribution* the target area distance is distributed uniformly, i.e., the sender sends messages to distant target areas with the same probability as to target areas in his vicinity. Figure 6 also shows the uniform target area distribution. The differences of the shown distribution to a perfect uniform distribution are due to the geographic distribution of underlay network routers. Especially for great distances, we cannot always find a target area at a wanted distance.

In the second scenario, large target areas at the size of states are addressed.

Small Target Areas: Figure 7 shows the cumulative stretch factor distribution for the two evaluated target area distance distributions. We get average stretch factors of 1.9 and 2.3 for local and uniform target area distribution, respectively. This difference is due to the fact that GRs mainly know shortcuts to designated GRs of nearby areas. A message to distant target areas is sent via more GRs than messages to nearby areas. Since each hop in the overlay network introduces additional hops in the underlay network, the detour in the underlay network is longer if more GRs are involved in message forwarding.

Moreover, we see that although the number of messages with high stretch factors is small there is still a number of messages with factors of 10 and more. Especially if the optimal path in the underlay network is very short, even a few GR hops in the overlay network lead to a very high stretch factor, although the absolute underlay network path length is still acceptable: We get 13.5 and 18.2 underlay network hops on average for local and uniform target area distribution, respectively. The maximum underlay path lengths are 44 and 46 hops. The optimal SPTs have 7.6 and 8.8 hops on average for local and uniform target area distribution, respectively, and a maximum path length of 19 hops (both, local and uniform target area distribution).

Large Target Areas: In the previous paragraph we considered comparably small target areas at the size of a city district. Since in our scenario exactly one GMS is responsible for a city district, each message had to be forwarded to only one GMS. Now, we consider geocast messages to larger areas, namely states, covered by a large number of GMSs.

Figure 8 shows the stretch factor distributions for state-level target areas. Suprisingly, in contrast to city district target areas, now the uniform target area distribution leads to smaller stretch factors than the local target area distribution. This is due to the fact that the majority of hops now belongs to the second phase when the message is distributed among all GRs in the target area rather than to the first phase, when the message is forwarded from the sender to the designated state GR. That means, messages to a certain state lead to nearly the same stretch factor, no matter where the sender is located. In our scenario, for example, the state Virginia has a stretch factor that is greater than all other factors of the other states in the simulation area. For the local target area distribution, 42% of the messages are sent to Virginia. This leads to the last steep slope of the "local target area" curve shown in Fig. 8.

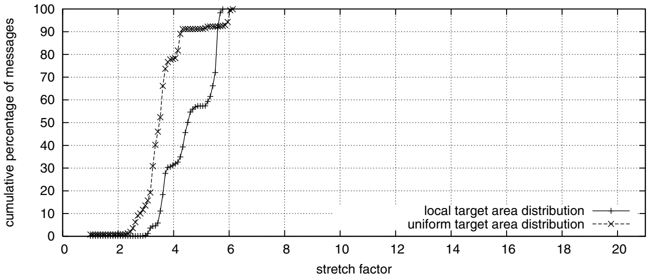

Fig. 8. Cumulative stretch factor distribution: large target areas (states)

	city district GR	city GR	state GR	country GR
local distr.	0.002	0.002	0.030	0.000
uniform distr.	0.002	0.003	0.083	0.000

TABLE I
AVERAGE LOAD OF GEOCAST ROUTERS

For uniformly distributed target areas, only about 8% of the messages are sent to Virginia. Therefore, the last steep slope of the curve “uniform target area distribution” is much smaller.

This evaluation shows that our heuristics performs well although a simple static GR placement strategy was used. Moreover, we see that our optimization using layer 3 multicast can improve the path lengths by factor of 2 and more which justifies to set up a SPT for senders having a long-term communication relationship with certain target areas.

C. GeoRouter Load

Another goal of our overlay network architecture is the distribution of load among GRs. Especially the load of high-level GRs like country or state GRs is critical for the scalability. Since it is difficult to predict the number of geocast messages sent in a certain period of time, we express GR load as the fraction of all messages sent. For instance, if 200,000 messages are sent in total and a GR forwards 100 of these messages, its load is $\frac{100}{200000}$.

Table I shows the average load of GRs in our scenario. The load of city district GRs is very low because load is distributed well among the great number of district GRs. For the local target area distribution, also the load of city and state GRs is relatively low because most messages by-pass these GRs by using shortcuts. Note that the country GR does not forward any message, because messages are only sent to target areas within the USA in our scenario, and every GR in the USA has shortcuts to every state GR in the USA. Especially for state GRs load grows significantly if target areas are distributed uniformly. This underlines the fact that our approach is tailored to scenarios with medium to strong message locality.

VIII. SUMMARY AND FUTURE WORK

We presented a geocast protocol for the efficient distribution of symbolically addressed geocast messages. This protocol operates on a symbolic location model that can be set up with low modeling effort compared to a geometric model, and it does not require complex geometric operations to forward messages. By using an overlay network, the protocol can be implemented without costly modification of the existing IP router infrastructure. Although the routing protocol does not rely on a globally available layer 3 multicast protocol, we also showed how layer 3 multicast can be utilized to improve efficiency. By means of simulation we showed that under the assumption of message locality, our approach leads to short message paths and low overlay network router load and thus high scalability.

In future work, we are going to investigate how to further improve routing in the overlay network. Our forwarding algorithm can benefit from additional overlay network links to create shortcuts to frequently addressed “hot spot” locations. The presented heuristics favors shortcuts to nearby target areas. Thus, it is less effective for distant hot spots. Therefore, we are going to investigate further heuristics that automatically set up shortcuts based on the observation of usage patterns.

REFERENCES

[1] J. C. Navas and T. Imielinski, “On reducing the computational cost of geographic routing,” Rutgers University, Department of Computer Science, Tech. Rep. DCS-TR-408, Jan. 2000.
[2] Julio C. Navas and T. Imielinski, “Geocast – geographic addressing and routing,” in *Proceedings of the Third Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom ’97)*, Budapest, Hungary, Sept. 1997, pp. 66–76.
[3] J. Roth, “Semantic geocast using a self-organizing infrastructure,” in *Innovative Internet Community Systems (I2CS)*. Leipzig, Germany: Springer, June 2003, pp. 216–228, LNCS 2877.
[4] F. Dürr, C. Becker, and K. Rothermel, “Efficient forwarding of symbolically addressed geocast messages,” in *Proceedings of the 14th International Conference on Computer Communications and Networks (ICCCN 2005)*, San Diego, CA, USA, Oct. 2005, pp. 77–83.
[5] J. C. Navas, “Geographic routing in a datagram internetwork,” Ph.D. dissertation, Rutgers University, Department of Computer Science, May 2001.
[6] J. Moy, “Multicast extensions to OSPF,” IETF, RFC 1584, Mar. 1994.
[7] D. Waitzman, C. Partridge, and S. Deering, “Distance vector multicast routing protocol,” IETF, RFC 1075, Nov. 1998.
[8] B. Fenner, M. Handley, H. Holbrook, and I. Kouvelas, “Protocol independent multicast–sparse mode (PIM-SM): Protocol specification (revised),” IETF, Internet-Draft, Oct. 2004.
[9] T. Imielinski and J. C. Navas, “GPS-based geographic addressing, routing, and resource discovery,” *Communications of the ACM*, vol. 42, no. 4, pp. 86–92, 1999.
[10] C. Maihöfer, “A survey on geocast routing protocols,” *IEEE Communications Surveys and Tutorials*, vol. 6, no. 2, 2004.
[11] B. Brumitt and S. Shafer, “Topological world modeling using semantic spaces,” in *Proceedings of the Workshop on Location Modeling for Ubiquitous Computing, UbiComp 2001*, Sept. 2001, pp. 55–62.
[12] F. Dürr and K. Rothermel, “On a location model for fine-grained geocast,” in *Proceedings of the Fifth International Conference on Ubiquitous Computing (UbiComp 2003)*, Seattle, WA, Oct. 2003, pp. 18–35.
[13] M. Theimer and M. B. Jones, “Overlook: Scalable name service on an overlay network,” in *Proceeedings of the 22nd International Conference on Distributed Computing Systems (ICDCS’02)*, Vienna, Austria, July 2002, pp. 52–64.
[14] B. Xu, A. Ouksel, and O. Wolfson, “Opportunistic resource exchange in inter-vehicle ad-hoc networks,” in *Proceedings of the IEEE International Conference on Mobile Data Management (MDM 2004)*, Berkeley, CA, Jan. 2004, pp. 4–12.
[15] H. Holbrook and B. Cain, “Source-specific multicast for IP,” IETF, Internet-Draft, Sept. 2004.
[16] M. Liljenstam, J. Liu, and D. M. Nicol, “Development of an internet backbone topology for large-scale network simulations,” in *Proceedings of the 2003 Winter Simulation Conference*, New Orleans, LA, Dec. 2003.