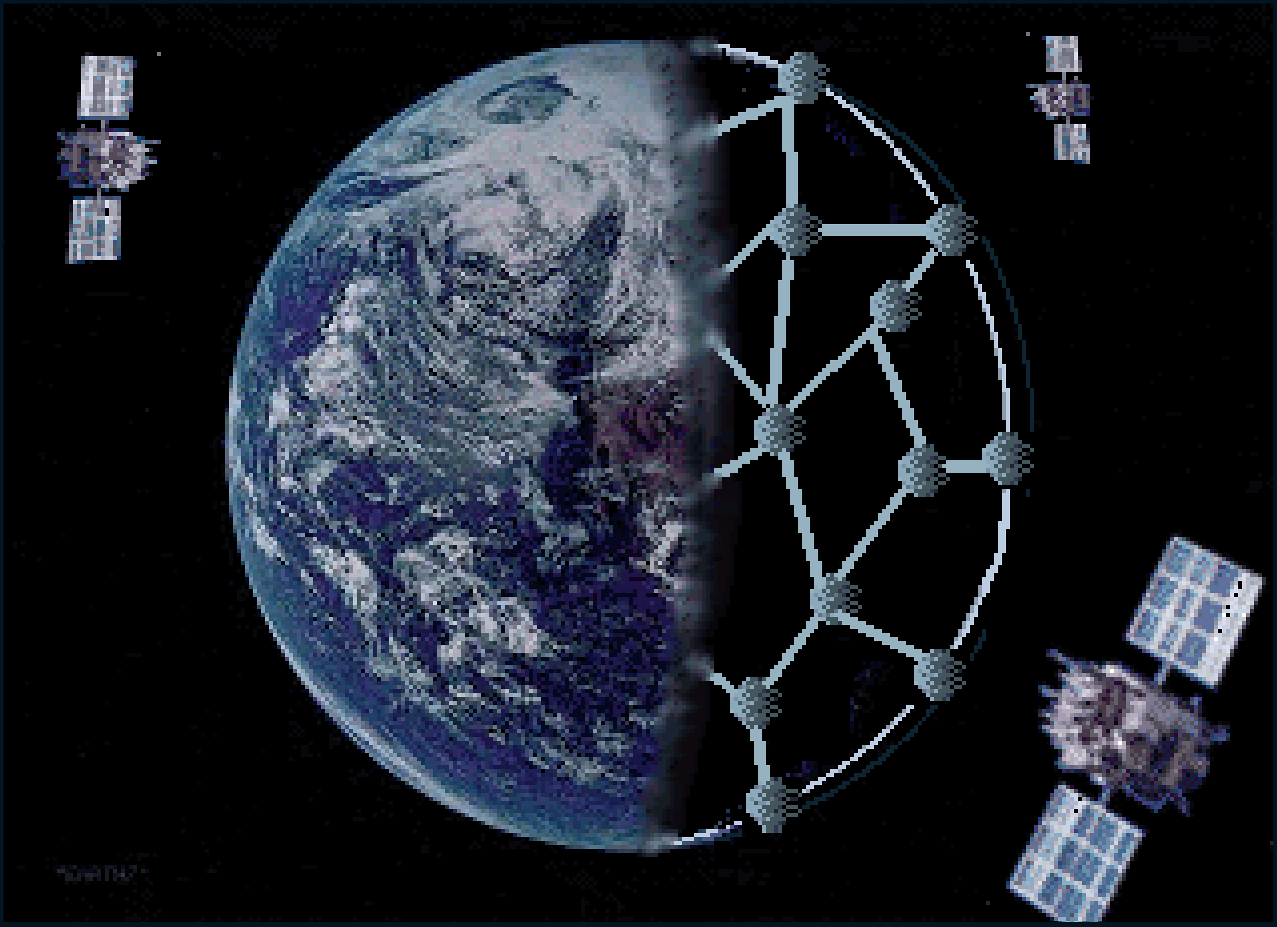
Tomasz Imielinski and Julio C. Navas

GPS-Based Geographic Addressing, Routing, and Resource Discovery

The Global Positioning System can be used to give every terminal a geographic address for multicasting to and from recipients within specified geographical areas.



GPS cards will soon be included in cars manufactured in the U.S. and Europe and possibly in every other form of mobile computer as well. A user's location will be another piece of information—as common as the date is today—getting input from the GPS when outdoors and from other location-providing devices when indoors. The availability of location information will have a broad effect on both application-level and network-level software. Possible new services and functions include geographic messaging, advertising, and resource discovery.

Geographic messaging is the ability to send a message selectively to specific geographic subareas defined by latitude and longitude—for example, sending an emergency message to everyone in a specific area, such as a building, train station, or highway. The ability to send a message to a distinct geographical area would make it possible to perform geographically targeted advertising through the Internet. For example, a business might want to advertise a given service only to clients within a certain geographic range, say, within two miles of the company's store. Conversely, users could use geographic messaging to locate services or resources within a geographical region, such as in their direct proximity. One can imagine a "Who is around?" service that would locate and identify the people present in a given geographic area. Assuming that terminals are also equipped with cameras, users could point their terminals in a specific direction and get annotation (links) on and to the objects displayed by camera viewers; the whole external world could be viewed as one large Web page, so a building, for example, might include a link explaining its business function. Links could also be attached to mobile objects appearing on the camera viewer.

To support such applications, location has to be a first-class citizen in networking protocols, like the Internet Protocol (IP) and Asynchronous Transfer Mode (ATM) and those in the application layer. Routing protocols for geographic messages should therefore be developed to allow routing to a specific area defined by a polygon of geographic coordinates. Location should also be a parameter in Web access protocols to deliver pages on servers within a given distance from the user. Distance-based Web bookmarks could help define the relevance of Web pages by using distance as an extra criterion when accessing material on the Web.

Our main objective here is to show how new services and new network functions could emerge as a consequence of location being universally available to mobile terminals. But how do the protocols have to be rewritten to support location-aware services, such as geographic messaging, geographic service discovery, and geographic service advertising? Geographic routing is a key requirement, and the exact routing mechanisms to make it happen are critical. We also look into geocasting, or broadcasting to geographical areas defined as arbitrary polygons, as well as the intersection of geocasting and multicasting.

Linking an IP address with a geographic location has been of interest to network researchers for quite some time. The first attempt to design a system that routes packets according to their geographic destination, and the work most like ours, was dubbed "Cartesian routing" by Gregory Finn in 1987 [5]. Xerox's PARC research laboratory also pioneered location-dependent services [10].

The recently proposed redesign of IP and the advent of the GPS [11,12] has given new impetus for this work. In the proposed redesign of IP [2], IP address type space was specifically allocated for geographic addresses [3, 9] that would be assigned to subnets and hosts based on geographic criteria. However, the sender of a "geographic message" would be unicasting messages only to hosts with geographic IP addresses. Our methods seek to provide the more general ability of sending a message to all recipients within a geographical area, regardless of whether or not the hosts have geographical addresses.

Addressing Model

2D geographic positioning offers latitude and longitude information as a 2D vector `< latitude, longitude >`, where longitude ranges from −180° (west) to 180° (east) and latitude ranges from −90°

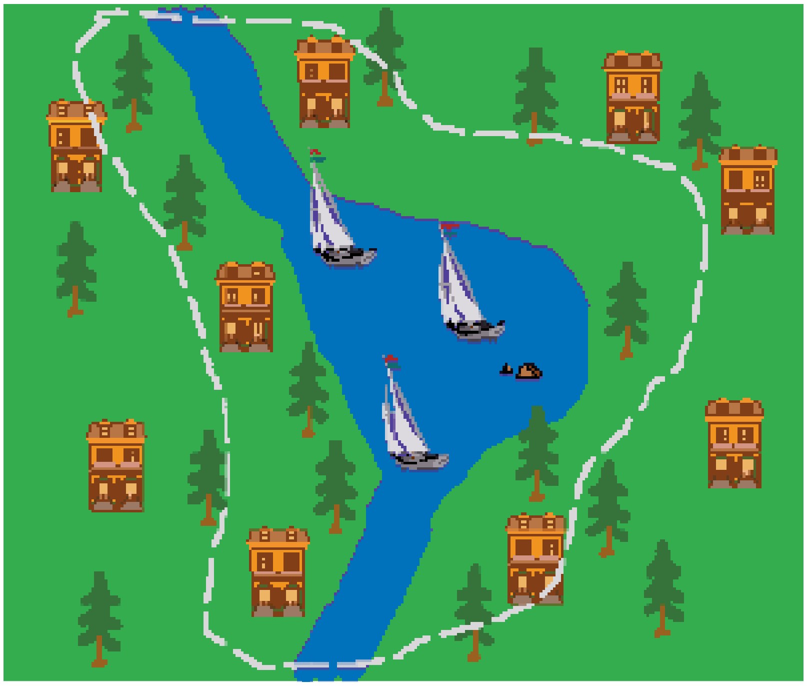


Figure 1. Interacting with a zoomable map interface.

(south) to 90º (north). Thus, < 40.48640, -74.44513 > are the geographic coordinates for New Brunswick, N.J., U.S.A.

Assuming the use of single precision floating-point numbers, 4B of addressing space are needed to store latitude and 4B to store longitude. Thus, a total of only 8B are needed to address the Earth's entire surface with precision down to 0.1 mile. A destination geographic address would be represented by some closed polygon, such as: point; circle(center point, radius); polygon (*point*1, *point*2, ... , *point*n−1, $point_n$, $point_1$), where each vertex of the polygon is represented by geographic coordinates. This notation would be used to send a message to everyone or to a group of people within a specified geographical area defined by the closed polygon.

Consider sending a message to the city hall of Fresno, Calif. We would specify its geographic limits as a series of connected lines forming a closed polygon surrounding city hall. Therefore, the address of Fresno city hall could look like: polygon([36.80,−119.80], [36.85,−119.76], ...)

In this hypothetical Fresno scenario, a user interacts with a zoomable map through a graphical user interface. The address of the message is specified as a polygon on the map. The polygon is then translated into geographic coordinates, and the message is sent to clients located within the bounds of that polygon. Figure 1 shows such a scenario, in which a polygon is drawn around the banks of a river.

Routing Geographically

In trying to deliver a message to any geographical destination, three basic types of solutions seem to work best—the geographic routing method, the geographic-multicast routing method, and the domain name service (DNS) method. We chose these solutions so the necessary geographic routing infrastructure in the Internet would vary from very little to significant. So far, we have implemented geographically aware software routers employing the geographic routing method. Evaluations of these geographic routers were published in [7] and demonstrated to the U.S. Defense Advanced Research Projects Agency (DARPA) and members of the DARPA research community during the DARPA/ITO Global Mobility (GloMo) meetings in 1997 and 1998.

All three of these solutions assume users can determine their own locations. While outdoors, they can use the GPS to determine their locations. When indoors, they have to use a different method; one possible solution is for each room in any building to include a radio beacon embedded in its ceiling. Each beacon would have its own geographic address, which it would broadcast periodically. The geographic address of the mobile hosts would be the same as that of the beacon. Therefore, mobile users would have an associated geographic address, even though they are indoors and their GPS modules are useless.

Geographic routing method. For routing, the GEO (short for geographic) routing method uses the geographic destination area information directly, in a form represented by a closed polygon, and includes it in the header information of a geographic message. Ideally, geographic routing would be implemented as part of the Network Layer (Layer 3) of the Open Systems Interconnect (OSI) protocol stack. However, to facilitate research and testing, we implemented the routing and forwarding logic in GEO as an application-layer software router. The software routers are designed to create a virtual internetwork overlaid onto the current IP network by using multicasting to discover neighbor routers and IP tunnels to transport data packets through areas that do not support geographic routing.

The GEO system includes three main components: GeoRouters, GeoHosts, and GeoNodes (see Figure 2). GeoRouters, or geographic routers, are in charge of moving a geographic message from a sender to a set of receivers. They are essentially IP routers that are geographically aware. Each is charged with performing geographic routing functions for the networks to which it is attached directly. Each GeoRouter keeps track of the geographic area it services (called its service area) by calculating the union of the geographic areas covered by the networks attached to it. A GeoRouter's service area is represented as a single simple closed polygon whose vertices are denoted by geographic coordinates.

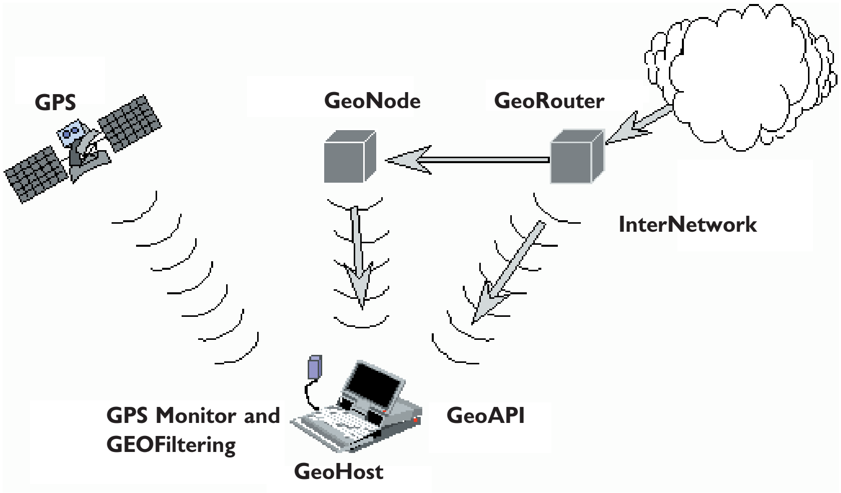


Figure 2. Components of the geographic routing system.

Before a geographic router can determine where to forward an incoming packet, it must first have a routing table containing information about the network topology and geography. Several protocols are available today for discovering the network topology and automatically configuring a routing table; they can be adapted to distribute a router's geographic location information, in addition to the other information already being passed along. For the purposes of geographic messaging, we extended the popular Routing Information Protocol (RIP) to include geographic location information. Using this protocol, which we call GeoRIP, a router has a routing table entry for each destination in the network. Each entry contains information on a destination's geographic location, its IP address, the shortest number of intermediate routers between the current router and the destination, and the preferred neighbor router to use as the next step on the path to the destination.

When forwarding an incoming packet, a router uses the routing table information to determine where the final destinations for the message reside and which neighbor routers have to be sent a copy of the packet. First, the geographic router uses the information in the routing table to search for and discover where to send the packet. The router then creates a list of the neighbor routers on the shortest paths to the destinations, and a copy of the message is sent to each neighbor router on the list. When a geographic message has been forwarded all the way from the sender to all the receivers, the routers will have created a shortest-path routing tree with the root at the sender and the leaves at all the receivers.

In order to reduce forwarding costs, the router keeps a cache of the next-hop destinations of the most recent geographic message packets. When a router receives a geographic message packet, it uses the incoming packet's sender IP address and destination polygon together as a key into the cache. If this is not the first packet to arrive for this destination and if the timer on the cache entry has not yet expired, the cache returns a list of all of the neighbor routers to which copies of the packet must be sent.

GeoHost software, which has to be installed on all computer hosts, consists of an application programming interface (API) and a location-monitoring process. The API can be used to create programs for sending and receiving geographic messages. The location-monitoring process continually updates the host computer's knowledge of its location by interfacing with GPS devices (if available) and by determining the address of the local geographic router.

A GeoNode is a buffer for messages whose lifetimes are due to expire. The GeoNode's main function is storing incoming geographic messages with lifetimes greater than zero for the duration of their lifetimes and periodically multicasting them on all the subnets or wireless cells to which they are attached. Each subnet and each wireless cell would have at most one GeoNode; it could also lack a GeoNode, but the geographic messages would lack lifetime expirations. The sender of the message would specify the lifetime of a geographic message; specifying message lifetimes might be necessary, because mobile receivers of geographic messages might arrive at the message destination some time after the geographic message first arrives.

Moreover, because several geographic messages would probably reside in a GeoNode at one time, the multicasting of the various messages would have to be scheduled. The scheduling algorithm would have to take into account the size of the message, its priority, and the speed of the subnet's transport medium. The GeoNode stores the message locally and assigns a multicast group to it. It periodically multicasts the message

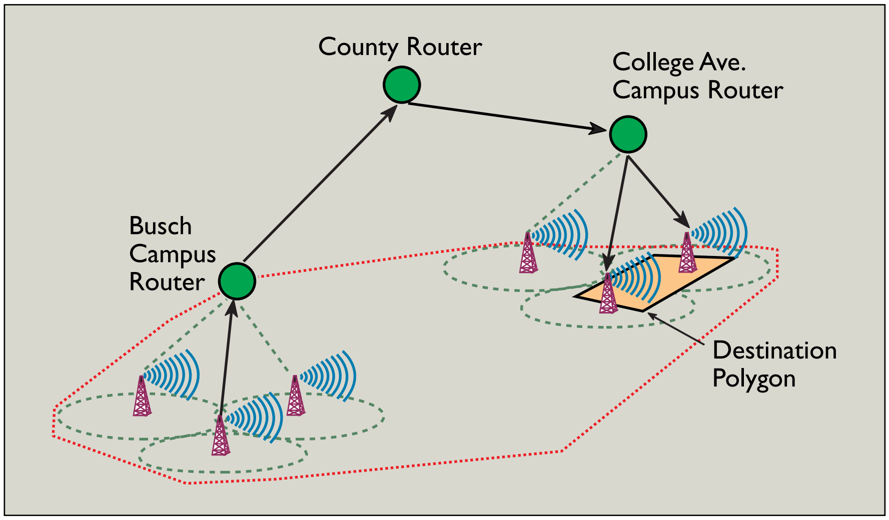


Figure 3. Geometric routing.

schedule to a well-known group address and multicasts each message to its assigned group. The GeoHosts receive the message schedule and determine whether the host computer is located inside the message's destination area. Software clients that want to receive a geographic message would then tune in to the appropriate multicast group to receive it.

In Figure 3, a user on Rutgers University's Busch Campus wants to send a message to the destination polygon around the Rutgers College Ave. Campus. The message is first passed to the Busch Campus router. Using the information in its routing table, the router determines that it does not service the target area, but it also realizes that the College Ave. router services the destination area. So it forwards the message to the county router, because the county router is the next router on the shortest path to the destination. Using the same algorithm, the county router decides to forward the message to the College Ave. router. The College Ave. router then transmits the message to all the wireless cells intersecting the destination area.

Geographic-multicast routing method. The geographic-multicast routing method leverages the power of multicasting to transport geographic messages to their destinations. We use two terms—"atoms" and "partitions"—to describe its operation. Atoms are the smallest geographical areas with geographic-multicast addresses. Partitions are larger geographical areas that also have geographic addresses. A state, county, or town might constitute a partition. Partitions and atoms are arranged in a hierarchical fashion. Each partition contains either a whole number of atoms or a whole number of smaller partitions. The sizes and shapes of the atoms and partitions are determined by the density of subnets and wireless cells in a particular geographic area.

Each partition and atom would have a geographic-multicast address for use by routers. By "geographic-multicast address," we mean each partition and atom would be mapped to a multicast address. The multicast group address would be chosen so it could be calculated using the geographic position of the atom or partition. With the large address space available through IPv.6, the multicast address itself could be encoded using longitude and latitude, simplifying the calculation of the appropriate group address for an atom or partition. Every GeoNode has to join the multicast groups for the atoms and partitions intersecting its geographic range. Thus, a GeoNode has to know not only its own range but also information about the partitions intersecting its range. The key idea here is to approximate the destination polygon with the smallest partition or atom that contains it and use the multicast address corresponding to that partition or atom as the address of that message. Since the partition or atom being used is only an approximation of the destination polygon, some GeoNodes outside the destination polygon erroneously receive the geographic messages.

In order to counter the erroneous receipt of messages, the original destination polygon is inserted into the multicast packet body. The GeoNodes then use the destination polygon to determine whether they should have actually received the message; if not, the message is ignored.

Multicast group information has to be propagated carefully. Because of the large number of atoms and partitions and the resulting large number of multicast groups, we will modify the Protocol Independent Multicast Sparse Mode (PIM-SM) [4], which is slated by the Internet Engineering Task Force to be the future standard multicast protocol. PIM-SM is meant to be used in wide-area networks, networks in which bandwidth is poor, and multicast groups with few or widely scattered members. PIM-SM assumes that not everyone wants to receive the multicast packets and relies on explicit join messages from group members. As a result, PIM-SM has the advantage of having to send multicast packets only to where they have been requested and not having to broadcast the initial packets, as the current multicast protocol does. PIM-SM is similar to core-based multicast trees [1] in that it uses

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.