Monitoring region relations in limited granularity geosensor networks

Matt Duckham[1], Lin-Jie Guan[1], Sanjiang Li[2], Jochen Renz[3]

1. University of Melbourne, Victoria 3010, Australia
mduckham@unimelb.edu.au

2. University of Technology Sydney, NSW 2007, Australia
Sanjiang.Li@uts.edu.au

3. Australian National University, Canberra, ACT 0200, Australia
jochen.renz@anu.edu.au

1. Introduction

The primary contribution of this paper is to propose and investigate a new efficient algorithm for determining the topological relationship between two regions monitored by a geosensor network (a wireless networks of miniaturized sensor-enabled computers monitoring phenomena in geographic space Nittel et al., 2004). The unique resource constraints imposed by technologies like geosensor networks mean that traditional, centralized approaches to spatial computation are inefficient and not scalable. In particular, technologies like geosensor networks typically impose high costs on communicating data between nodes (e.g., due to limited node energy), compared with relatively low costs of processing data at the node itself (so-called *in network* processing) (cf. Estrin et al., 2000).

Consequently, our algorithm is *decentralized*: nodes operate without any centralized control using only knowledge of their local environment and spatially nearby neighbors, with no system component ever possessing global knowledge of the network state.

2. Decentralized topological region queries

Algorithm 1 presents a version of our algorithm, adopting the distributed algorithm presentation style of Santoro (2007). In brief, this approach defines for every node a number of *states* (in upper case letters). In each state, nodes can respond to different *events* (in italics), specifically receiving a message (*Receiving* keyword), responding to alarms triggered by some condition (*When* keyword) or spontaneous events external to the system (e.g., algorithm initialization, *Spontaneously* keyword). The responses to each event are specified as atomic *actions*, which are procedures that the node can complete without interruption (i.e., they require no interaction with other nodes).

2.1 Preliminaries

Following previous work, we assume:

- a geosensor network, modeled as a graph $G = (V, E)$. Our algorithm places no restrictions on whether the graph is planar, although the results of the algorithm may vary depending on the network structure (see Sadeq and Duckham, 2008).
- node sensors capable of determining whether a node detects a region A or B, modeled as a function *sense* $: V \to \mathcal{P}(\{A, B\})$. These regions may be derived from thresholding of sensor measurements of continuous fields (e.g., A equals "temperature above 20°C," B equals "soil moisture below 15%"), (Worboys and Duckham, 2006). After Egenhofer and Fransoza (1991), regions are assumed to be homeomorphic to a disk.
- a routing algorithm, capable of routing information to a "sink" node, modeled as a function *next* $: V \to V \cup \{\varnothing\}$ (e.g., *next*$(v) = v'$ indicates v' is the next node from v along a route to the sink, where $\{v, v'\} \in E$; *next*$(v'') = \varnothing$ indicates node v'' is the sink) (cf. Karp and Kung, 2000; Bose et al., 1999; Goldin et al., 2004).

These assumptions are listed as restrictions to algorithm 1 in line 1. However, individual nodes only have *local* knowledge of these functions (i.e., a given node v will have access to its own sensed data *sense*(v), but not to that of any other node *sense*(v')). To enforce and highlight the local knowledge to a node, we use the overdot notation $\mathring{sense}$ (termed "local" or "my" *sense*) inside the algorithm to refer to the current node's knowledge of that function (i.e., for an arbitrary node $\circ \in V$ clear from the context, $\mathring{sense}$ is equivalent to *sense*$(\circ)$).

2.2 Algorithm

The algorithm proceeds in five main steps:

1. Nodes initialize from the INIT state by broadcasting to network neighbors a `ping` message specifying the regions they can sense locally (i.e., either $\varnothing$, A, B, or A and B) (lines 6–7).
2. Nodes that receive an `ping` message can deduce whether they are at the boundary of region A and/or B, by comparing their local sensed value with those of their one-hop neighbors, subsequently transitioning into state BNDY (lines 9–12).
3. When nodes in the BNDY state have received `ping` messages from all their network neighbors, they compute their local boundary intersections as a three bit binary number, $b \in \mathbb{B}^3$. The first (smallest) bit indicates whether that node detects an intersection between the boundary of B and the interior of A (i.e, $A^\circ \cap \partial B \neq \varnothing$); the second, whether it detects an intersection between the boundary of A and the interior of B (i.e., $\partial A \cap B^\circ \neq \varnothing$); the third (largest) bit indicates whether it detects an intersection between the boundary of A and the boundary of B (i.e., $\partial A \cap \partial B \neq \varnothing$). Boundary nodes that have a non-zero boundary intersection number forward this information as a `rprt` message to the sink (lines 13–22), finally transitioning to a DONE state. Thus, the information sent by nodes corresponds to

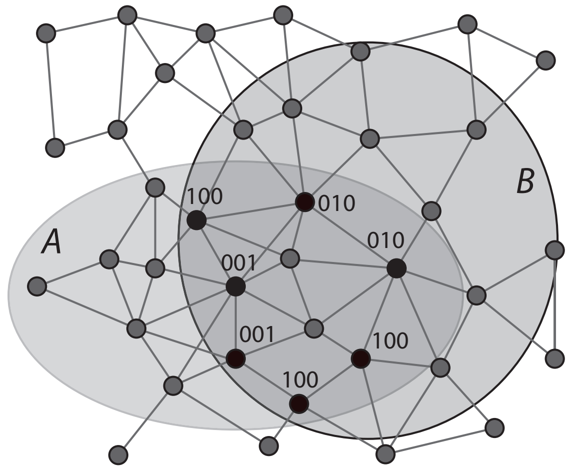

Figure 1: Example of boundary node states

$\bigvee B$	Topological relation (after Egenhofer and Fransoza, 1991)
(no messages received) 000	A, B disjoint
001	A contains B
010	B contains A
100	A, B meet or equal
011*	A, B overlap
101	A covers B
110	B covers A
111	A, B overlap

Table 1: Determining the 3-intersection topological relation between regions (see Algorithm 1, line 25), for set $B \subset \mathbb{B}^3$ of `rprt` message payloads received at sink (001 indicates $A^\circ \cap \partial B \neq \varnothing$; 010 indicates $\partial A \cap B^\circ \neq \varnothing$; 100 indicates $\partial A \cap \partial B \neq \varnothing$) * Note, 011 only possible for sink node awaiting delivery of 100 message.

a 3- (as opposed to 4-) intersection model of regions (see Figure 1). Only boundary nodes detecting these three intersections report to the sink. The implications of this design are discussed further in the analysis section below.

4. Nodes receiving a `rprt` message by default forward this message to the sink (lines 23–27). However, DONE nodes only forward information if this include new knowledge not already forwarded by that node (lines 28–30).

5. In the final step, the sink node uses a logical (bitwise) OR operation to compose all the messages received (line 25). Following directly from Egenhofer and Franzosa's 4-intersection model (Egenhofer and Fransoza, 1991), the topological relations that can be deduced from these messages are given in Table 1. Note that in this case we have only three intersections (intersection of region interiors is omitted) for efficiency reasons, discussed further below.

Algorithm 1 Topological relations between regions A and B

Restrictions: $G = (V, E)$, *sense* : $V \to \mathcal{P}(\{A, B\})$, *next* : $V \to V \cup \{\varnothing\}$
States: $\{$INIT, IDLE, BNDY, DONE$\}$
Transitions: $\{($INIT, IDLE$), ($IDLE, BNDY$), ($BNDY, DONE$)\}$
Initialization: All nodes in state INIT
Local variables: *bndy* : $V \to \mathcal{P}(\{A, B\})$, initialized $\mathring{bndy} \mapsto \varnothing$; *bnum* : $V \to \mathbb{B}^3$, initialized $\mathring{bnum} \mapsto 000$

INIT
Spontaneously
  broadcast (`ping`, $\mathring{sense}$) --*Broadcast locally detected regions to one-hop neighbors*
  become IDLE --*Transition to* IDLE *state*

INIT, IDLE, BNDY
Receiving (`ping`, x)
  if $x \neq \mathring{sense}$ and $\mathring{sense} \neq \varnothing$ **then** --*If node is inside a region, adjacent to a node outside*
    set $\mathring{bndy} \mapsto \mathring{bndy} \cup (\mathring{sense} - x)$ --*Store names of regions which this node bounds*
    become BNDY --*Transition to boundary state*

BNDY
When `ping` message received from all one-hop neighbors
  if $\mathring{bndy} = \{A, B\}$ **then** --*If node at boundary of both A and B*
    set $\mathring{bnum} \mapsto 100$ --*Store boundary intersection number bitwise OR 100*
  if $\mathring{bndy} = \{A\}$ and $B \in \mathring{sense}$ **then** --*If node at boundary A only, and senses B*
    set $\mathring{bnum} \mapsto 010$ --*Store boundary intersection number bitwise OR 010*
  if $\mathring{bndy} = \{B\}$ and $A \in \mathring{sense}$ **then** --*If node at boundary B only, and senses A*
    set $\mathring{bnum} \mapsto 001$ --*Store boundary intersection number bitwise OR 001*
  if $\mathring{i} \neq 000$ **then** --*If node has non-zero boundary intersection state*
    send (`rprt`, $\mathring{bnum}$) to $\mathring{next}$ --*Forward `rprt` message to sink*
  become DONE

IDLE, BNDY
Receiving (`rprt`, b)
  if $\mathring{next} = \varnothing$ **then** --*If sink reached*
    Deduce topological relation between A and B --*See Table 1*
  else
    send (`rprt`, b) to $\mathring{next}$ --*Forward `rprt` message to sink*

DONE
Receiving (`rprt`, b)
  if $b \vee \mathring{bnum} \neq \mathring{bnum}$ **then** --*Check using bitwise OR ($\vee$) if b contains information not in $\mathring{bnum}$*
    send (`rprt`, b) to $\mathring{next}$ --*Only forward if `rprt` message contains new information*

3. Analysis

As communication is the most resource-intensive operation in a sensor network, computational efficiency is typically measured in terms of communication complexity. The communication complexity of the `ping` message is $\Omega(|V|)$ messages sent (every node sends exactly one message). However, to initialize any sensor network must broadcast handshake or "hello" messages simply to join the network. Thus, we argue that cost of the `ping` messages can be amortized by the cost of network initialization.

The `rprt` messages are initiated at the boundaries only, following a linear path to the sink, using data aggregation wherever possible (see Algorithm 1, line 30). Because linear features, like boundaries, contain $O(\log |V|)$ nodes, the expected communication complexity of the `rprt` messages is $O(\log |V|)$. Note that Algorithm 1 might easily be adapted slightly to use all four intersections (including interior-interior intersections). However, by the same logic, the communication complexity of such an algorithm would increase to $O(|V|)$ (since the interiors of regions would be involved in generating `rprt` messages, not just boundaries). Thus, the increased efficiency of Algorithm 1 comes at the cost of the algorithm being unable to discern apart meet or equal relations (see Table 1).

4. Discussion and conclusions

Algorithm 1 is able to efficiently respond to queries about region topology, using no centralized control. Adopting a 3- as opposed to 4-intersection approach leads to greatly improved computational efficiency ($O(\log |V|)$ as opposed to $O(|V|)$ communication complexity), at the cost of slightly less topological detail about the regions (meet and equals cannot be discerned apart). Potentially, other efficient approaches and algorithm might be used to discern apart these two relations; in other cases applications may not require that level of detail. When compared with other related approaches, this algorithm has the substantial advantage that it requires only qualitative neighborhood information about a node's location (i.e., in contrast to Guan and Duckham, 2009, the algorithm operates in a non-planar graph, and requires no coordinate location information for nodes).

In addition to the level of topological detail, this type of approach does raise important questions about the limited *spatial* granularity of any geosensor network. Even though this spatial granularity may be of comparable levels of detail to the finest granularity conventional spatial data capture sources (like remote sensing, e.g., with sensor nodes potentially decimeters apart), there still exist important questions about what the network *can't* sense. For example, it is always possible that even though two regions appear to be touching given the level of granularity of the geosensor network, in actuality these regions are overlapping (the network just happens to have no node in the small area of interior overlap). Thus, current work is investigating not only what can be detected by the geosensor network, but what we can infer about the actual state of the world based on this limited granularity information (see Table 2).

In addition to the level of topological detail, further work that is currently under way to extend this initial work includes:

- Generalizing the algorithm from regions homeomorphic to a disk, to regions more broadly, including regions with holes or disconnected regions;

Relation detected by the geosensor network	Relations that may in actuality hold
disjoint	{disjoint, meet, overlap}
overlap	{overlap}
contain/by	{contain/by,cover/by,overlap}
cover/by	{cover/by,overlap}
equal	{cover/by, overlap}

Table 2: Summary of current work on relationship between topological detail at limited and unlimited spatial granularity

- Dealing with uncertain regions, where the boundary may not be crisp;
- Extending the algorithm to deal efficiently with time-varying regions, monitoring topological change over time (cf. Jiang and Worboys, 2008, 2009); and
- Thorough empirical evaluation of the algorithm, corroborating the computational analysis.

Acknowledgements

This research was supported under Australian Research Council's Future Fellowship scheme (project numbers FT0990521, FT0990811, FT0991917).

References

Bose, P., P. Morin, I. Stojmenovic, and J. Urrutia (1999). Routing with guaranteed delivery in ad hoc wireless networks. In *Proc. 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp. 48–55.

Egenhofer, M. and R. Fransoza (1991). Point-set topological spatial relations. *International Journal of Geographical Information Science 5*(2), 161–174.

Estrin, D., R. Govindan, and J. Heidemann (2000). Embedding the internet. *Communications of the ACM 43*, 38–41.

Goldin, D., M. Song, A. Kutlu, H. Gao, and H. Dave (2004). Georouting and delta-gathering: Efficient data propagation techniques for geosensor networks. In A. Stefanidis and S. Nittel (Eds.), *GeoSensor Networks*, Boca Raton, FL, pp. 73–95. CRC Press.

Guan, L.-J. and M. Duckham (2009, November-December, 2009). Decentralized computing of topological relationships between heterogeneous regions. In B. Lees and S. Laffan (Eds.), *10th International Conference on GeoComputation*, UNSW, Sydney.

Jiang, J. and M. Worboys (2008). Detecting basic topological changes in sensor networks by local aggregation. In *Proc. 16th ACM International Conference on Advances in Geographic Information Systems (ACMGIS)*, New York, pp. 1–10. ACM.

Jiang, J. and M. F. Worboys (2009). Event-based topology for dynamic planar areal objects. *International Journal of Geographical Information Science 23*(1), 33–60.

Karp, B. and H. T. Kung (2000). GPSR: Greedy perimeter stateless routing for wireless networks. In *Proc. 6th annual international conference on Mobile computing and networking (ACM/IEEE MobiCom)*, Boston, MA. ACM.

Nittel, S., A. Stefanidis, I. Cruz, M. Egenhofer, D. Goldin, A. Howard, A. Labrinidis, S. Madden, A. Voisard, and M. Worboys (2004). Report from the First Workshop on Geo Sensor Networks. *ACM SIGMOD Record 33*(1), 141–144.

Sadeq, M. and M. Duckham (2008). Effect of neighborhood on in-network processing in sensor networks. In T. Cova, K. Beard, M. Goodchild, and A. Frank (Eds.), *GIScience 2008*, Volume 5266 of *Lecture Notes in Computer Science*, Berlin, pp. 133–150. Springer.

Santoro, N. (2007). *Design and Analysis of Distributed Algorithms*. New Jersey: Wiley.

Worboys, M. F. and M. Duckham (2006). Monitoring qualitative spatiotemporal change for geosensor networks. *International Journal of Geographical Information Science 20*(10), 1087–1108.