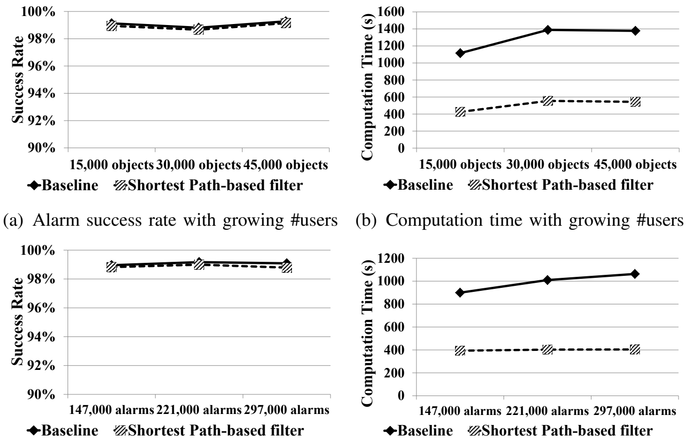

(a) Alarm success rate with growing #users (b) Computation time with growing #users

(c) Alarm success rate with growing #alarms (d) Computation time with growing #alarms

Fig. 6. Effect of growing number of users and alarms

computation cost of servers and conserving energy of mobile clients. For those applications in which high success rate is required, both the *baseline* and the the shortest path-based filter are good options. Especially, for some applications in which the battery power of mobile clients is not a serious problem, the *baseline* may be a better choice since it has a slightly higher success rate than the shortest path-based filter.

VI. Related Work

Existing research on spatial alarms and location reminders mainly focuses on the Euclidean space. [3] proposes an approach to process spatial alarms in the Euclidean space by combining spatial indexes such as R-tree and Voronoi Diagram with the safe period. [2], [5] develop a safe region-based approach for spatial alarm processing in the Euclidean space.

We would like to note that spatial alarms are fundamentally different from continuous spatial queries in terms of their objectives and target applications. Continuous spatial queries, such as finding the restaurants 2 miles around me, are defined based on the current location of mobile users for finding points of interest within a predefined range from the current location of a mobile user. [7], [14] use Euclidean distance while [10], [13] use road network distance in continuous spatial query modeling and processing. Continuous spatial queries are inadequate and incur poor performance for location trigger-based applications, such as hazard alert systems and location-based advertisement. In contrast, spatial alarms are independent of the current location of mobile users and are defined based on some future location of interest, such as "alert me when I am 2 miles to the public library in Buckhead". Clearly, (1) the focal of a spatial alarm is not the current location of the mobile client but the current location of the alarm target (e.g., the public library in Buckhead), and (2) the spatial alarm evaluation should not be triggered until the mobile client who subscribed to the alarm is in the vicinity of the alarm target. Thus, spatial alarms are essential building blocks for location trigger-based applications, such as location-based advertisement applications.

VII. Conclusion

We have presented RoadAlarm – an efficient and scalable service architecture and a suite of algorithms for processing road network-based spatial alarms. By utilizing spatial constraints on road networks and mobility patterns of mobile users on spatially constrained road networks, we have shown through extensive experiments that the RoadAlarm service architecture can significantly reduce the computation time for calculating hibernation time upon wakeups of mobile users, compared to the state of the art conventional approaches.

This paper has made three technical contributions. First, we present the RoadAlarm service architecture that defines road network-based spatial alarms as star-shaped subgraphs and uses the border points and road network distance to represent the boundary of road network-based spatial alarms. By making the spatial alarm service architecture road network aware, we are able to capitalize on the spatial constraints of mobile users' movements to improve the performance and scalability of the RoadAlarm service engine and the energy efficiency of RoadAlarm client on mobile devices. Second, we present the RoadAlarm basic algorithm that combines subscription filter with Euclidean Lower Bound (ELB) filter to reduce the number of unnecessary shortest path computations. Third but not the least, we introduce three motion-aware filters to further reduce the computation cost by minimizing the number of unnecessary shortest path computations as well as client wakeups by making use of the steady motion-based motion patterns of mobile users traveling on a road network.

Acknowledgment

This work is partially supported by grants from NSF NetSE, NSF CyberTrust, an IBM faculty award and a grant from Intel ISTC on Cloud Computing.

References

[1] Apple. Optimize your settings. http://www.apple.com/batteries/iphone.html.

[2] B. Bamba, L. Liu, A. Iyengar, and P. S. Yu. Distributed processing of spatial alarms: A safe region-based approach. In *ICDCS*, 2009.

[3] B. Bamba, L. Liu, P. S. Yu, G. Zhang, and M. Doo. Scalable processing of spatial alarms. In *HiPC*, 2008.

[4] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[5] M. Doo, L. Liu, N. Narasimhan, and V. Vasudevan. Efficient indexing structure for scalable processing of spatial alarms. In *GIS*, 2010.

[6] G. R. Hjaltason and H. Samet. Distance browsing in spatial databases. *ACM Trans. Database Syst.*, 24, June 1999.

[7] H. Hu, J. Xu, and D. L. Lee. A generic framework for monitoring continuous spatial queries over moving objects. In *SIGMOD*, 2005.

[8] gt-mobisim. http://code.google.com/p/gt-mobisim/.

[9] U.S. Geological Survey. http://www.usgs.gov/.

[10] K. Mouratidis, M. L. Yiu, D. Papadias, and N. Mamoulis. Continuous nearest neighbor monitoring in road networks. In *VLDB*, 2006.

[11] D. Papadias, J. Zhang, N. Mamoulis, and Y. Tao. Query processing in spatial network databases. In *VLDB*, 2003.

[12] P. Pesti, L. Liu, B. Bamba, A. Iyengar, and M. Weber. Roadtrack: scaling location updates for mobile clients on road networks with query awareness. *Proc. VLDB Endow.*, 3, 2010.

[13] S. Xing, C. Shahabi, and B. Pan. Continuous monitoring of nearest neighbors on land surface. *Proc. VLDB Endow.*, 2, August 2009.

[14] X. Yu, K. Q. Pu, and N. Koudas. Monitoring k-nearest neighbor queries over moving objects. In *ICDE*, 2005.