Feature Clustering for Data Steering in Dynamic Data Driven Application Systems

Alec Pawling and Greg Madey

Department of Computer Science and Engineering
University of Notre Dame
Notre Dame, IN, USA, 46556

Abstract. In this paper, we describe how feature clustering on real-world cell-phone data can be used to locate the impact area of emergency events. We first examine the effect of two emergency events on the call activity in the areas surrounding the events. We then investigate how the time series of the affected areas behave relative to the time series of their respective neighboring areas. Finally, we examine the differences in hierarchical clusterings of the time series of the affected areas and neighboring areas.[1]

1 Introduction

The Wireless Phone-based Emergency Response (WIPER) system is a proof-of-concept Dynamic Data Driven Application System (DDDAS) designed to leverage real-time streaming cell phone data to provide high-level information about an emergency situation to emergency response managers. WIPER consists of modules for automatically detecting emergency events and for running and validating predictive simulations of potential outcomes [1,2,3,4]. Schoenharl and Madey [5] describe an approach for on-line simulation validation for WIPER using streaming cell phone data as it becomes available. In this paper we address the problem of identifying the area for which the simulations should be run.

In an emergency situation, it is likely that the area of interest is small relative to the total coverage area of the cell phone network. Running predictive simulations for the entire coverage area is problematic in terms of computational requirements and the amount of data produced that must in turn be validated and presented to emergency response managers. In this paper we describe an approach for identifying the area affected by an emergency using feature clustering. We illustrate the effectiveness of this approach using two case studies of emergency events that appear in real-world cell phone data.

2 Related Work

Dynamic Data Driven Application Systems (DDDAS) are characterized by their ability to incorporate new data into running models and simulations as they

[1] This material is based upon work supported by the National Science Foundation, CISE/CNS-DDDAS, Award #CNS-0540348.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 460–469, 2009.

become available and to steer data collection, enabling the simulations to receive and utilize the most relevant data [6,7]. Plale *et al.* [8] use the amount of variance in an ensemble of weather forecast simulations to collect additional data and direct additional computational resources to the areas where additional simulation runs are needed. Flikkema *et al.* [9] uses data models to filter observations at the sensors. In this case, the interesting observations are those that do not match the data model, and it is these that are transmitted for further processing.

WIPER receives a single data stream of cell phone usage information that contains a time stamp, de-identified values indicating the individuals making and receiving the call, and the tower the caller's phone is communicating with. We have the latitude, longitude, and postal code of each tower, and we link this information with the call data. From this data stream, we generate a set of time series for spatially disjoint areas using the tower location information. For each spatial area, we count the number of calls made in 10 minute intervals, producing a vector of non-negative integers for each time step.

We can view this series of vectors as a data set for machine learning algorithms. Let the data set $\mathbf{D}$ be an $n \times m$ matrix with n data items and m features. We can view the problem of identifying the columns of interest, which corresponds to an area in the real world, as the feature selection problem.

Feature selection is the process of identifying the best subset of available features of a data set for machine learning algorithms. Feature selection serves to improve the quality of machine learning models, reduce the computation required to train and utilize these models, and provide a better understanding of the model. One approach to feature selection is to combine similar features using a clustering algorithm [10]. Feature clustering has been used to reduce large feature spaces for applications such as text mining [11].

Data clustering is an unsupervised machine learning method for grouping the rows of a data set $\mathbf{D}$ based on some distance measure. Hierarchical algorithms identify a nested set of partitions in the data. Most hierarchical methods take an agglomerative approach, meaning that there are initially n clusters, each containing one data item in $\mathbf{D}$. These clusters are iteratively merged until all of the data items belong to the same cluster. Popular agglomerative clustering algorithms include single-link and complete-link. These approaches may be implemented using a graph where the data items are represented as vertices and edges are added between two vertices in increasing order of distance between the two corresponding data items. At each step, the clusters in the single-link approach are the connected components and the clusters in the complete-link approach are the completely connected components [12].

Feature clustering applies clustering techniques to the transpose of a data set. Rodrigues *et al.* [13] describe an algorithm for clustering the features of a data stream. The algorithm is a divisive-agglomerative algorithm that uses a dissimilarity measure based on correlation along with a Hoeffding bound to determine when clusters are split. The algorithm utilizes the fact that the pairwise correlation of the time series, $corr(\boldsymbol{a}, \boldsymbol{b})$, can be computed using a small number

of sufficient statistics. The key observation by Rodrigues *et al.* [13] is that it is only necessary to maintain a small number of values to compute the correlation of two time series. For each time series it is necessary to keep track of $\sum_{i=1}^{n} a_i$, $\sum_{i=1}^{n} b_i$, $\sum_{i=1}^{n} a_i^2$, and $\sum_{i=1}^{n} b_i^2$. For each pair of time series $\sum_{i=1}^{n} a_i b_i$ must be updated with the arrival of each data item. Additionally, the number of data items that have arrived so far, n, must be known. Rodrigues *et al.* [13] use correlation distance, $diss(\boldsymbol{a}, \boldsymbol{b}) = 1 - corr(\boldsymbol{a}, \boldsymbol{b})$, as a dissimilarity measure.

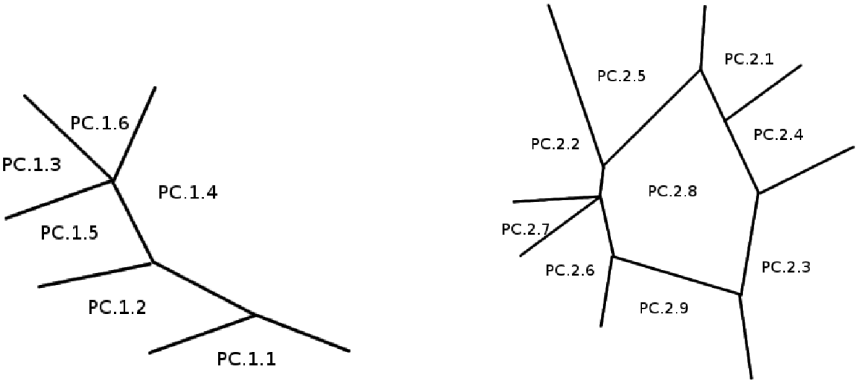
In this paper, we explore the possibility of using feature clustering for selecting spatial areas of interest in the WIPER system. We examine the distance between the time series of neighboring postal codes using the correlation distance described above and Euclidean distance, and we use these distance measures in conjunction with the single-link algorithm to visualize changes in the relationship between the call activities in neighboring postal codes when an emergency event occurs.

3 Experimental Setup

We examine the expression of two emergency events in real-world cell phone usage data. The first emergency event is an explosion, and the second is a riot. The two emergencies occur in different geographic locations and take place at different times of the year. First, we establish that each emergency event produces a corresponding change in the service usage and that the impact of the event on the call activity decreases as an increasing area surrounding the emergency event is considered. We aggregate the call data to count the number of phone calls made in a set of postal codes every 10 minutes in the city in which the emergencies occur. We study the two postal codes in which the emergency events occur and their neighbors. Next, we examine the correlation distance and Euclidean distance between the call activity time series of the postal codes in which the emergencies occur and the neighboring postal codes. Finally, we cluster the call activities time series for each set of postal codes for a normal day and the day of the emergency using an agglomerative clustering algorithm.

To measure the effect of the emergency events on the call activity of area surrounding the event, we first determine the location of the event from news reports. Using geographic information system tools, we establish an approximate latitude and longitude of the event. With this information and the latitude and longitude of the towers, we can filter the data to obtain calls made within any desired radius around the event.

For the remaining work, we aggregate the call activity by postal code. For each emergency event, we examine the postal codes containing the approximate latitude and longitude established for each emergency and their neighboring postal codes in the city in which the emergency occurs. There are 6 postal codes surrounding the first emergency (an explosion) and 9 surrounding the second (a riot). We denote the postal codes for each emergency as PC.1.1, PC.1.2, ..., PC.1.6 and PC.2.1, PC.2.2, ... PC.2.9, respectively. The first emergency occurs in PC.1.4, and the second occurs in PC.2.8. Figure 1 shows the approximate configuration of the postal codes.



(a) Emergency 1 **(b)** Emergency 2

Fig. 1. Approximate configuration of the postal codes. Emergency event 1 occurs in PC.1.4, which is at the edge of the city. Emergency event 2 occurs in PC.2.8, which is in the center of the city.

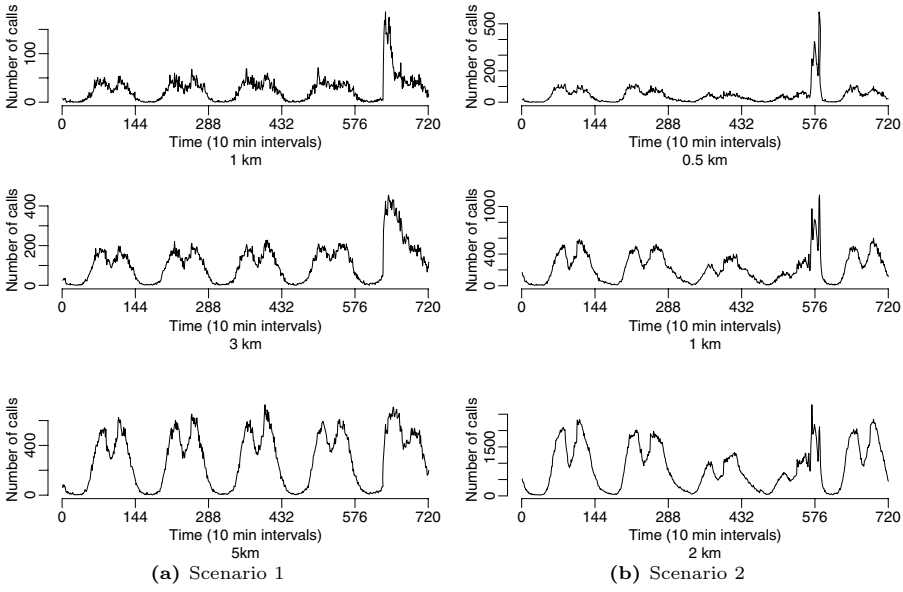
For each emergency event, we examine the correlation distance and the Euclidean distance between the postal code in which the emergency occurs and the neighboring postal codes for two weeks leading up to each event. We examine both the cumulative correlation distance and utilize a sliding window. Euclidean distance is computed only over a sliding window of 1 day of data. Both sliding windows contain the most recent 144 observations (taken at 10 minute intervals).

Finally, we compare time series clusterings for the two emergency events with those of normal call activity. We use single link agglomerative clustering with correlation distance and Euclidean distance, and we visualize the clusters using dendrograms.

4 Results

The columns in Fig 2 show the time series of call activities for the five days leading up to each emergency. Each row, from the top of the figure to the bottom, includes data from a greater area surrounding the location of the emergencies. The first emergency (left column) occurs at about 11 A.M. on the fifth day, and we see a corresponding increase in call activity at this time (approximately 640 minutes). The severity of this spike in activity decreases as the radius of the area increases from 1 km to 5 km. The second emergency (right column) occurs at approximately 2 o'clock on the morning of the fifth day, though we see elevated call activity even before midnight. As with the first scenario, the spike in call activity becomes less dramatic as a larger area, up to 2 km in radius, surrounding the emergency is included.

Figures 3 and 4 each show the correlation (both cumulative and over a sliding window) and Euclidean distances between the postal codes in which the emergency events occur and the neighboring postal codes for two weeks leading



(a) Scenario 1 **(b)** Scenario 2

Fig. 2. This figure shows the effect of the emergency situations on the call activity through the surrounding cell towers. The left column shows the time series for the first situation, which occurs at approximately 11 A.M. on the fifth day in the time series (approximately 642 minutes). The right column shows the time series for the second situation, which occurs at approximately two o'clock on the fifth day (approximately 588 minutes). In both cases, the severity of the activity spike decreases as a greater area is considered.

up to the emergency events. The left columns show the cumulative correlation distance used by Rodrigues *et al.* [13]. The center and right columns show the correlation distance and Euclidean distance, respectively, over a one day sliding window.

In Fig 3, we see an increase in each distance measure at the end of each time series. The cumulative correlation distance has only a slight increase at the end of the time series when the emergency event happens. These increases are more dramatic in the cases where a sliding window is used. Note that in the time series of Fig 3 there are two days of missing data, from 576 to 864 minutes. These missing data are not noticeable in the cumulative correlation distance; however, they lead to undefined correlation distances and Euclidean distances of 0 for 144 time steps when the entire sliding window contains 0 for all features. In Fig 4 we see similar increases in distance. The fact that the cumulative correlation distance shows only a small increase compared to the case where only a portion of the history of the time series is considered may indicate that this distance measure is dominated by older observations, making this cumulative measure to insensitive anomalies. The detrimental affect of old, stale data is discussed by Aggarwal in [14].

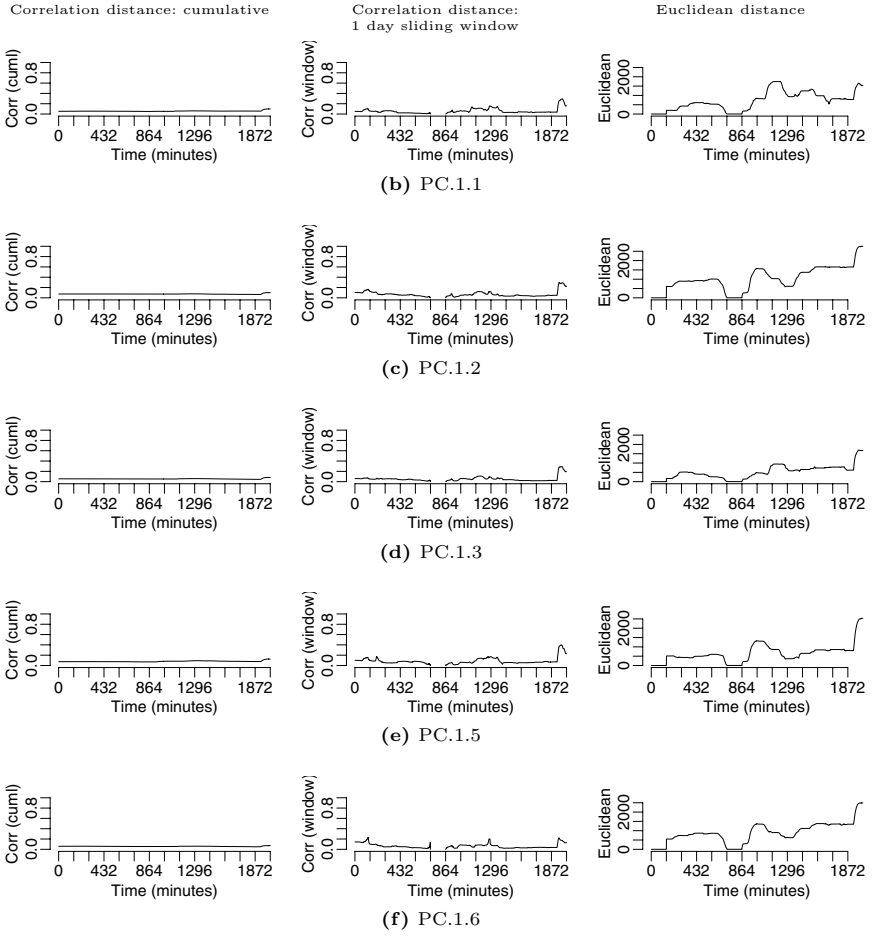
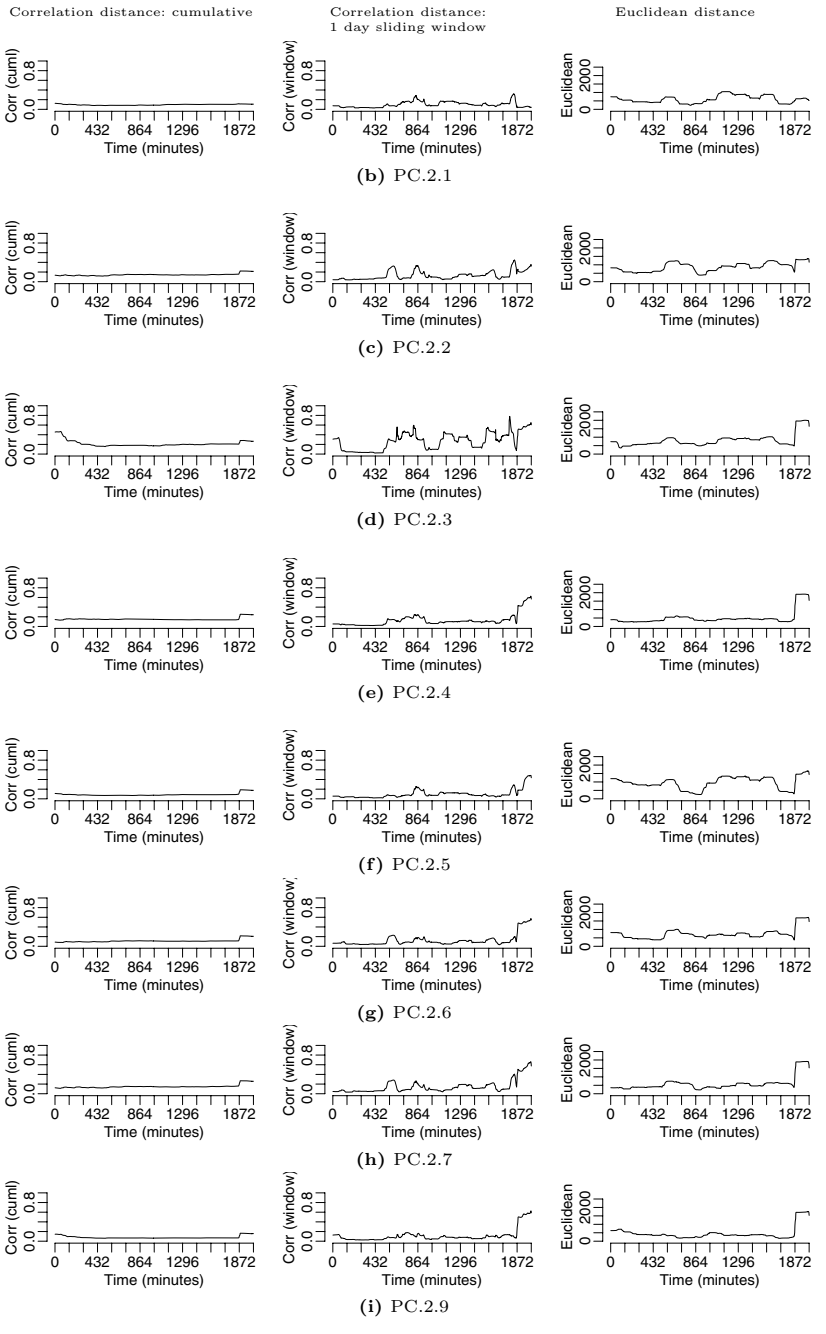


Fig. 3. This figure shows the correlation distance (cumulative and sliding window) and Euclidean distance between the postal code in which emergency event 1 occurs (PC.1.4) and its five neighboring postal codes for a two week period leading up to the emergency situation

Figures 5 and 6 show the contrast in clusterings for a day of normal activity (left column) and a day containing an emergency (right column). We cluster each day of data with the single link agglomerative algorithm using two different dissimilarity measures: correlation distance and Euclidean distance. Figure 5 shows the clusters for the first emergency situation. In both the correlation and Euclidean distance clusterings, the distance of PC.1.4, the postal code in which the emergency occurred, is significantly larger than the distance between any two clusters on the day of normal activity. In Fig 6, we see a similar separation of PC.2.8, the postal code in which the emergency occurred, along with PC.2.1



(b) PC.2.1

(c) PC.2.2

(d) PC.2.3

(e) PC.2.4

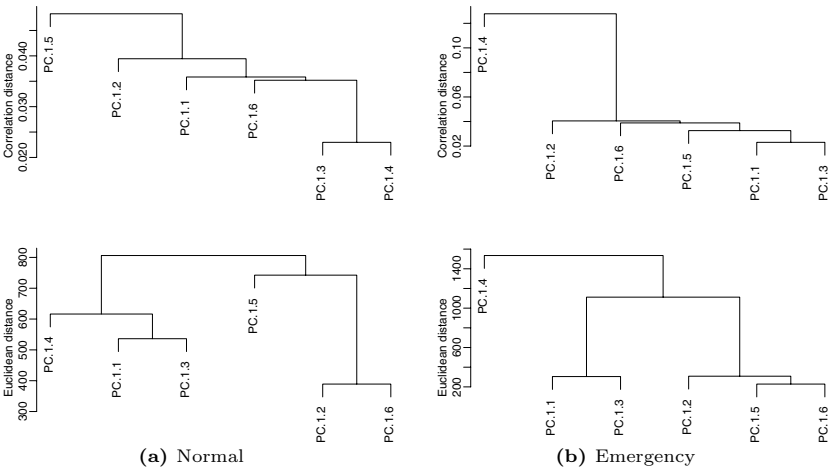
(f) PC.2.5

(g) PC.2.6

(h) PC.2.7

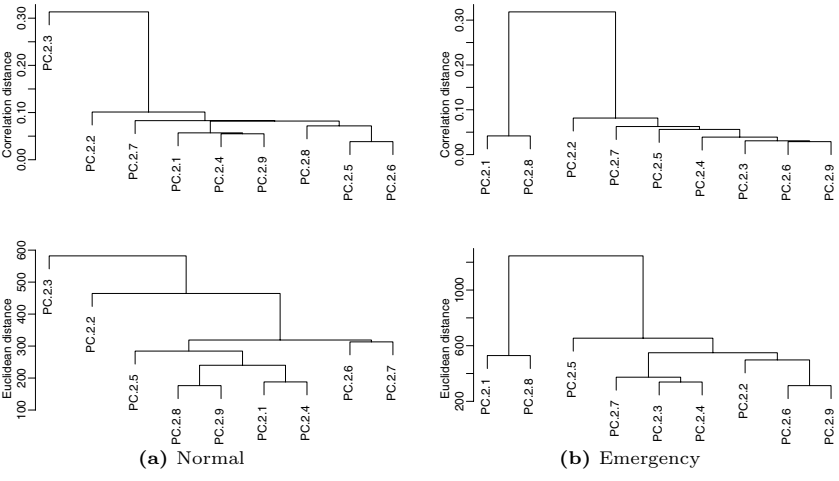
(i) PC.2.9

Fig. 4. This figure shows the correlation distance (cumulative and sliding window) and Euclidean distance between the postal code in which emergency 2 occurs (PC.2.8) and its five neighboring postal codes for a two week period leading up to the emergency situation



(a) Normal **(b)** Emergency

Fig. 5. This figure shows the clustering of the call volume time series for postal codes surrounding the first emergency event. The left column shows the clustering of one day of normal activity and the right column shows the clustering of the day of the first emergency event, which occurs in PC.1.4. Note that for both the correlation distance (top row) and Euclidean distance (bottom row), PC.1.4 is near other clusters during the day of normal activity but far from all other clusters during the day of the event.



(a) Normal **(b)** Emergency

Fig. 6. This figure shows the clustering of the call volume time series for postal codes surrounding the second emergency event. The left column shows the clustering of one day of normal activity and the right column shows the clustering of the day of the second emergency event, which occurs in PC.2.8. In this case, the call activity in PC.2.1 is also affected by this event. Note that for both the correlation distance (top row) and Euclidean distance (bottom row), the cluster containing PC.2.1 and PC.2.8 is near other clusters during the day of normal activity but far from all other clusters during the day of the event.

from the remaining clusters, though the increase in distance is not as dramatic as in the previous case. It is not surprising that PC.2.1 and PC.2.8 end up in the same cluster on the day of the emergency since we do not see the same increase in distance in Fig 4 between these to features as we do between PC.2.8 and the remaining features at the time the emergency occurs.

5 Conclusions and Future Work

In this paper, we have explored the possibility of using feature clustering to identify areas of interest from a set of spatially disjoint time series from real-world cell phone data. We have shown that emergency events can cause a spatially constrained change in call activity and that the area affected by this change can be detected using a clustering algorithm.

While this approach is promising, there is more work to be done before it can be deployed in the WIPER system. We need to determine the appropriate parameters for the approach, including the time series sampling interval, the level of spatial aggregation, and the length of the sliding window. Most importantly, while the dendrograms we have presented are compelling, we must do more work to understand how the clusters change over time in the absence of emergency events to gain an understanding of their stability and the amount of variation to be expected under normal circumstances. The work in this paper has been mostly qualitative, we must now pursue a more quantitative approach to automate the detection of areas of interest using feature clustering.

References

1. Madey, G.R., Barabási, A.L., Chawla, N.V., Gonzalez, M., Hachen, D., Lantz, B., Pawling, A., Schoenharl, T., Szabó, G., Wang, P., Yan, P.: Enhanced situational awareness: Application of DDDAS concepts to emergency and disaster management. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 1090–1097. Springer, Heidelberg (2007)
2. Pawling, A., Schoenharl, T., Yan, P., Madey, G.: WIPER: An emergency response system. In: Fiedrich, F., de Walle, B.V. (eds.) Proceedings of the 5th International ISCRAM Conference (2008)
3. Pawling, A., Yan, P., Candia, J., Schoenharl, T., Madey, G.: Anomaly Detection in Streaming Sensor Data. In: Intelligent Techniques for Warehousing and Mining Sensor Network Data. IGI Global (forthcoming)
4. Madey, G.: WIPER: The Integrated Wireless Phone-based Emergency Response System (2008), `http://www.nd.edu/~dddas`
5. Schoenharl, T.W., Madey, G.: Evaluation of measurement techniques for the validation of agent-based simulations against streaming data. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 6–15. Springer, Heidelberg (2008)
6. Darema, F.: Dynamic data driven applications systems: A new paradigm for application simulations and measurements. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3038, pp. 662–669. Springer, Heidelberg (2004)

7. Douglas, C.C.: DDDAS: Dynamic Data Driven Application Systems (2008), http://www.dddas.org
8. Plale, B., Gannon, D., Reed, D., Graves, S., Droegemeier, K., Wihelmson, B., Ramamurthy, M.: Towards dynamically adaptive weather analysis and forecasting in LEAD. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3515, pp. 624–631. Springer, Heidelberg (2005)
9. Flikkema, P.G., Agarwal, P.K., Clark, J.S., Ellis, C., Gelfand, A., Munagala, K., Yang, J.: From data reverence to data relevance: Model-mediated wireless sensing of the physical environment. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 988–994. Springer, Heidelberg (2007)
10. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1189 (2003)
11. Dhillon, I.S., Mallela, S., Kumar, R.: A divisive information-theoretic feature clustering algorithm for text classification. Journal of Machine Learning Research 3, 1265–1287 (2003)
12. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: A review. ACM Computing Surveys 31(3), 264–323 (1999)
13. Rodrigues, P., Gama, J., Pedroso, J.P.: Hierarchical time-series clustering for datastreams. In: Proceedings of the First International Workshop on Knowledge Discovery in Data Streams (2004)
14. Aggarwal, C.C.: On biased reservoir sampling in the presence of stream evolution. In: Proceedings of the 32nd Conference on Very Large Databases (2006)