# Automated Sensor Verification using Outlier Detection in the Internet of Things

Nauman Javed and Tilman Wolf
Department of Electrical and Computer Engineering
University of Massachusetts, Amherst, MA, USA
{njaved,wolf}@ecs.umass.edu

***Abstract*—Cyber-physical systems uses sensors, actuators, and computation to solve problems that cross the physical world and computational world. Traditionally, the systems have been designed to be specific to one application domain and to be managed by a single entity. However, to achieve scalability, it is necessary to reuse sensors and actuators across multiple application domains. This broad use of sensors and actuators, as envisioned for the Internet of Things, raises many new problems related to interoperability, resource sharing, security and trust, and economics. In this paper, we address the problem of trust – specifically how to verify sensor information that is gathered from multiple sensors that are managed by different entities using outlier detection. We present a technique for automatically deriving a model of the physical phenomenon that is measured by the sensors. This model is then used to compare sensor readings and to identify outliers through spatial and temporal interpolation. We evaluate our system and demonstrate its effectiveness in the context of weather sensing. Nevertheless, the technique is applicable to any application domain where the underlying phenomenon is continuous.**

***Index Terms*—Cyber-physical systems, trust, modeling, interpolation**

## I. Introduction

Cyber-Physical Systems (CPS) cross the boundaries between the physical world and the computation world. The basic operation of a CPS is shown in Figure 1. Using sensors, CPS obtain information about the physical world. This information is used in the computational component of the CPS. As a result of this computation, the CPS performs an action in the physical world (i.e., actuation). This action may result in changes to the physical world that can be sensed, thus closing the feedback loop.

There are numerous application domains where this basic operational principle can be applied. These domains range from focused control problems (e.g., industrial automation, aircraft control, etc.) to larger-scale problems (e.g., home automation, smart grid, environmental monitoring). Recently, there has been considerable interest in CPS since there is hope that this technology can help solve problems in important societal challenges, such as sustainable energy, smart transportation, health care, etc.

In most cases, existing CPS designs are *domain-specific*, i.e., sensors, computation, and actuation are optimized for a specific domain. In addition, the CPS components are also deployed and managed by the same administrative entity. As such, these CPS follow a "stovepipe design." While stovepipe designs are effective at solving problems within their domain, they also raise a number of concerns. In particular, it is difficult to deploy large cyber-physical systems since the cost of sensors and actuators becomes prohibitively expensive. This lack of scalability also inhibits innovation since new ideas (e.g., novel computational solutions) cannot easily deployed due to the cost of complexity of having to deploy sensors and actuators.

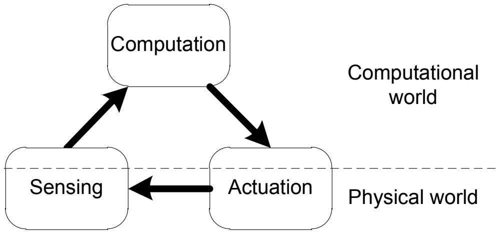

Fig. 1. Principles of Operation of Cyber-Physical Systems.

To address these problems in existing CPS architectures, there has been a recent trend toward looking at CPS in a slightly broader context. In particular, there have been a few examples of CPS applications that use sensor information from multiple domains to solve problems *across domains*. For example, GPS information is used to infer driving habits that determine insurance rates, weather information is used to control smart home operation, etc. In such systems, sensor information and actuator control crosses application domain boundaries. Once sensors and actuators can be shared across domains, scalability problems can be solved since the cost of the devices interacting with the physical world can be amortized over many applications. In addition, novel CPS uses can be deployed more easily since they can use existing infrastructure. In the limit, this cross-domain use of sensors, actuators, and computational components is captured in the "Internet of Things" (IoT), which draws on an analogy from the Internet where many different computers interact using a common data communication network.

One key characteristic of the Internet of Things is that CPS components are managed by different administrative entities. This main difference to traditional CPS raises many interesting new problems:

- Interoperability: Components need to be able to exchange data and control information using well-defined formats.

- Resource sharing: Multiple entities may needs access to a single resource (e.g., an actuator); this access needs to be managed.
- Trust: Different entities may not trust each other, which requires mechanisms for verification. This aspect also requires techniques for establishing security and controlling privacy.
- Economic exchanges: To provide incentives for sharing sensor information and access to actuators, mechanisms for economic exchanges need to be in place.

In our work, we address one specific problem that is essential to the resolution of the lack of trust between entities: *How to automatically detect outliers in the CPS sensor information at runtime?* Since sensors are managed by different entities, the information reported by a sensor can be incorrect due to malicious intent, due to calibration errors, etc. The incorrect sensor values can show up as outliers in a model of the CPS observations. These outliers can also appear as a result of genuine extreme values in the (correctly sensed) physical phenomenon. Computational components that use a sensor's data thus can use results from our system to decide if this data can be trusted.

Our key insight is that sensors typically report on a property in the physical world. In many cases, sensors obtain information from physical phenomena that are continuous. That is, sensors in close proximity read similar values since the physical phenomenon typically does not change drastically over a short distance. Example of such phenomena are: air temperature, barometric pressure, air quality, traffic congestion, etc. Thus, we develop a system that evaluates sensor readings to determine which ones are outliers in the context of other sensor readings. The outlier detection process makes use of multiple types and sources of sensor data to determine if the reported observations match the temporal and spatial context.

An important requirement for our work is that the mechanism for determining if a reading is an outlier should be done completely automatically. In a large-scale Internet of Things, it is not feasible to have humans in the loop to provide domain-specific expertise. Thus, our mechanism determines automatically which types of sensor information to use for outlier detection and what kind of model to use to represent the physical phenomenon captured by the sensors. This automation allows the outlier detection process to be "domain-agnostic" and thus broadly applicable to many different phenomena.

We show the effectiveness of our approach in the context of weather sensing. This domain provides multiple different types of sensor sources (e.g., temperature, barometric pressure, etc.) that can be used for cross-correlation. Thus, we can develop an internal model of weather in a particular region and determine which readings are outside what is considered normal. Despite the evaluation of our work in the context of weather, our work is not limited to this domain. The outlier detection technique presented in this paper can be applied to any domain where physical phenomena are continuous.

The specific contributions of our paper are:

- Design of an outlier detection system that evaluates sensor readings within a temporal and spatial context.
- Techniques for automatically determining the type of internal model that is used to represent the physical phenomenon and to detect outliers in the sensor data.
- Evaluation and demonstration of effectiveness of the proposed technique in a weather sensing scenario.

The remainder of the paper is organized as follows. Section II discusses related work. The general problem statement for sensor outlier detection in the Internet of Things is presented in Section III. Our sensor outlier detection technique is presented in Section IV. Section V presents evaluation results that show the effectiveness of our approach. Section VI summarizes and concludes this paper.

## II. Related Work

Design challenges in cyber-physical systems [10] have been studied widely [9], [11], [12], [17]. Many existing CPS solutions are based on stovepipe architectures. Such examples include applications in energy [7], transportation [18], and health care [2].

The technical requirements for CPS have been discussed in [15]. Specialized CPS extensions to programming languages (e.g., [14]) and real-time operating systems (e.g., [3]) have been proposed. Security issues in cyber-physical system, which are related to the focus of our paper, have been explored in various contexts (e.g., control systems [5], [16]).

Detecting anomalies in sensor data is a problem that appears in many domains. One example is anomaly detection in network traffic [1], [6], [8]. In our prior work, we have developed techniques to combine anomaly detection information from multiple sources to get more accurate results [13]. In our work, we use an approach that is conceptually similar (albeit different in the details), where information from multiple sensors is correlated to identify anomalies. For temporal extrapolation, we can use Holt-Winter forecasting, which has also been used to estimate network traffic volumes [4].

## III. Distributed Sensing in the Internet of Things

In many application areas, the required information about the physical environment is sensed by a group of distributed sensors, each of which accomplishes the sensing task independent of the other sensors. The individual, independent bits of information complete the picture about physical phenomena spread over a particular field of interest. Such is the case, for example, in weather sensing, where weather stations deployed over a certain geographical area individually sense one or more weather variables, each in its own vicinity. Examples of weather variables sensed are air temperature, air pressure, visibility, and dew point etc.

Let us suppose there are $n$ sensors distributed over a geographical space, each capable of sensing $k$ different variables of interest, denoted by $X_1, X_2, \ldots, X_k$. Sensor $i$ senses and reports a time series of the sensed variable $X_j$, denoted by $X_{ji}(t)$. So, at time $t'$, each of the $n$ sensors would sense and

TABLE I
WEATHER VARIABLES USED AS SENSOR INPUTS.

| Variable | Example value |
|---|---|
| Time | May 31, 2011, 5:00 a.m. |
| Latitude | 41.37N degrees |
| Longitude | 73.48W degrees |
| Elevation | 458 feet |
| Visibility | 5.0 miles |
| Temperature | 61 Fahrenheit |
| Dew Point | 61 Fahrenheit |
| Pressure | 1022.4 millibar |

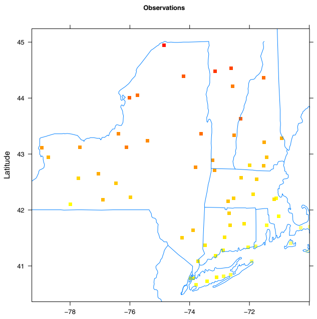

Fig. 2. Observed Values of Pressure by 90 Weather Stations.

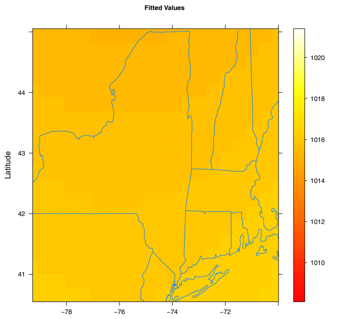

Fig. 3. Model with 2nd Degree Polynomial.

report the value of $Xj$ in its own vicinity, represented by the set $Xj(t')$ as follows:

$$X_j(t') = \{X_{j1}(t'), X_{j2}(t'), ..., X_{jn}(t')\} \qquad (1)$$

Before $X_j(t')$ can provide a coherent picture about the variable $X_j$ at time $t'$, each value in the set needs to be verified. Suppose a sensor $m$ reports erroneously about $Xj$. The first step to verify sensor $m$'s report is to see if it is an outlier in the context of other sensors' reports. Hence, our objective is to automatically figure out if the sensor $m$'s report is an outlier using the other sensors' reports.

The sensor reports at time $t'$ about the variable $X_j$, contained in the set $X_j(t')$, may be physically related to a similar set, $X_p(t')$, denoting sensed values of variable $X_p$ at time $t'$, or multiple such sets pertaining to different variables, as well as to the spatial parameters. But such physical relationships are not known, and we assume no such prior domain-specific knowledge in our case. Hence, in our case, all $X_j(t')$, for $j = 1 \dots k$, are simply sets of numbers representing different variables of interest.

To validate our results, we focus on a weather sensing application. As an illustration of multiple sensed variables from multiple, distributed sensors in a CPS, Table I shows values of four weather variables sensed by weather sensors deployed across the northeastern United States, mostly the New England area.

## IV. SENSOR OUTLIER DETECTION

The physical phenomena represented by the sensed values of the distributed sensors is assumed to be continuous and thus have some regularity to it. If this regularity can be discovered from the sensed values, then erroneous sensor reports can possibly be detected. The patterns in the observed phenomena are of two kinds: spatial and temporal. Finding the appropriate spatial patterns or models can help determine the expected values at different spatial coordinates at an instant in time. The temporal models help determine the expected values over time at each spatial coordinate. We exploit these two facts in determining the expected value of variables of interest at all coordinates in the spatial domain of interest at any point in time.

### *A. Spatial Models*

For multiple variables affecting a certain variable of interest, all spread over a geographical space, multiple regression can provide a viable model. In our experiments, we utilized a data set pertaining values of four weather variables, Visibility, Temperature, Dew Point, and Pressure, collected by 89 weather stations deployed at different locations in the North-Eastern United States. This data set contains weather variable values reported at one hour intervals between May 30, 2011 to June 15, 2011. As mentioned earlier, we do not make use of any domain-specific knowledge, in this case meteorology, and assume no prior knowledge of any relationships among these four weather variables as well as their relationship to the spatial coordinates or time.

An example of multiple regression applied to determine values of the weather variable Pressure is shown in Figures 2 and 3. Here, a spatial model of barometric Pressure fits a 2nd degree polynomial of the three other weather variables and the three positional variables, Latitude, Longitude, and Elevation.

With the assumption of no prior relationship among the variables of interest, the first challenge is to discover these relationships. Even though multiple regression provides a sound theoretical basis for model formulation, finding the model is difficult because the significance of different variables, and their polynomial degrees, toward determining a specific dependent variable are not known. There are some statistical methods available to search for the appropriate regression model, such as “Step-wise Regression” and “All-Subsets Regression.” But these methods do not provide a fully automated way of model finding. Statistical model finding, as a result, is an effort involving a lot of human judgement and “eye-balling” verification of the observations and models. To avoid this human factor in the loop, we develop mechanisms to automate this process.

### *B. Validation of Spatial Models*

In multiple regression, usually increasing the degree of the polynomial being fit to the data (observed or sensed values) results in a “better” estimate of the observed values. In other words, the error of the estimated values decreases, in general, with the increasing degree of the fitted polynomial. As an example, Figure 4 shows the results of fitting different degrees of polynomials used to estimate the Pressure values at an instant in time, across the spatial domain of interest. The

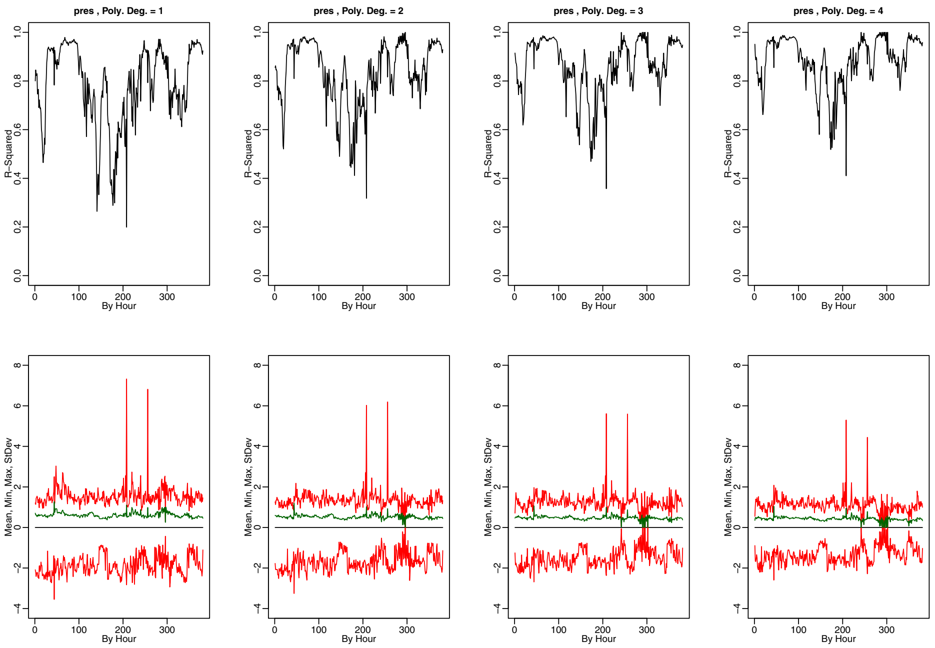

Fig. 4. Error Statistics for Different Polynomials; Hourly Data from May 30, 2011 to June 15, 2011.

Pressure estimates do, in fact, improve with the increase in the polynomial degree of the independent or predictor variables.

The same effect is better visualized in Figure 5, where we average out the errors over time and plot the error versus the polynomial degree used to model the four weather variables, Visibility, Temperature, Dew Point, and Pressure.

Trying to improve the model fit by increasing the polynomial degree of the predictor variables, quantified through measures such as the Standard Error and the R-Squared values, does not necessarily translate into a “good” model. This is because, although the higher degree polynomials might closely fit the data through which the polynomial coefficients have been calculated at the first place, the same polynomial may perform poorly on data that was not specifically included to calculate these polynomials. To compensate for such effects, the observed/sensed data is broken down into the Training and Testing samples. Polynomials are calculated using the Training sample, and then this polynomial model is tested on the Testing sample. A “good” model is then one that has better error statistics not only on the Training sample, but also on the Testing sample.

Figure 6 shows an example of fitting different degrees of polynomials to estimate the Pressure values with the error statistics shown for the Training as well as the Testing samples. It becomes evident that, although the error statistics for the Training sample improve with increasing degrees of polynomials fit, but the error statistics for the Testing sample actually start deteriorating after a certain point. We exploit this fact in developing an automated scheme for model finding, described in the next sub-section.

### C. *Automated Spatial Model Choice*

Our automated spatial model choice scheme is based on the realization that increasing the polynomial degree of the model can only be justified if the improvement in the error statistics on the Training set can be validated on the Testing set(s) as well. Based on this idea, our spatial model selection scheme, at an instant in time, proceeds as follows:

1) Keep increasing the degree of the polynomial until the marginal percentage increment in the R-Squared value falls below a specified threshold. Choose this polynomial degree and proceed to Step 2.
2) At the polynomial degree with the highest R-Squared value, thus achieved, test the error statistics of the model on a Testing sample. If the error statistics, for example the standard deviation of error, is within a specified limit of the Training sample error statistics, accept the model. Otherwise, repeat Step-2 on one lower degree of polynomial fit.

Thus, we are able to derive a model of the continuous physical phenomenon from our Training set. The derived model can then be used to interpolate spatially (e.g., to verify the correctness of other sensors) or to extrapolate temporally (e.g., to predict sensor values in the near future). (Note that this prediction is not equivalent to weather forecasting, which uses domain-specific knowledge and aims at predictions much longer ahead than our system.)

## V. Evaluation

Using our spatial model selection scheme as described above, we model four weather variables, Visibility, Temperature, Dew Point, and Pressure in a data set collected from sensors deployed at various locations in the northeastern

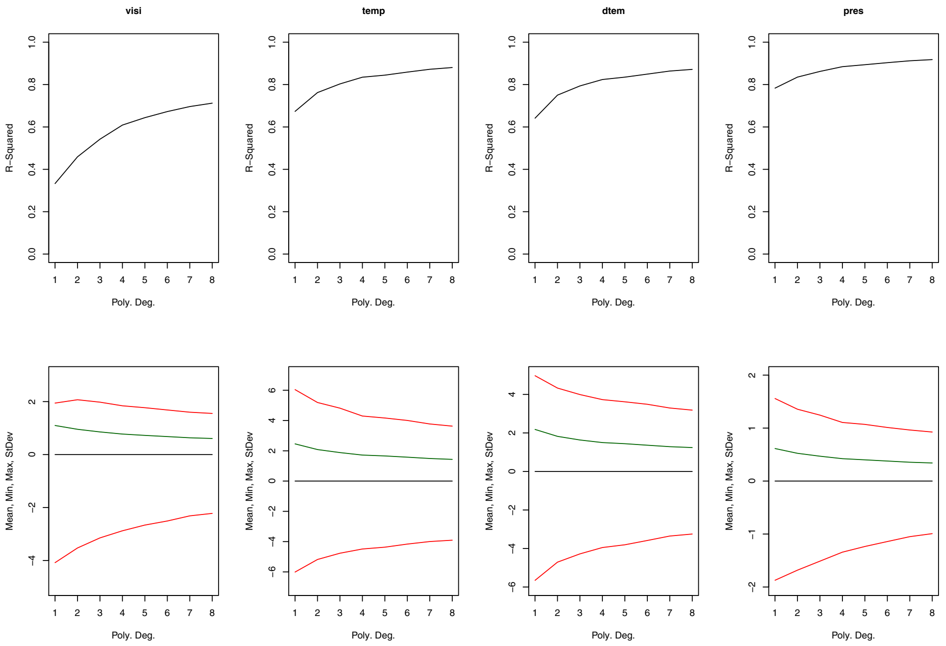

Fig. 5. R-Squared and Error Statistics for Different Polynomials.

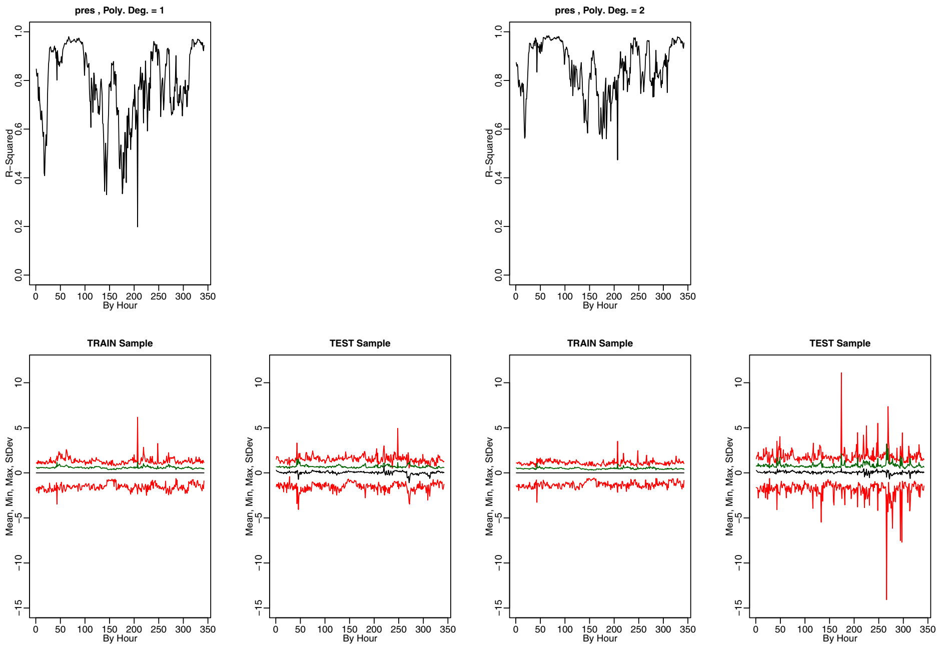

Fig. 6. Pressure Error Statistics for Training Vs. Test Samples.

United States. After an automated choice of spatial model at each time step, our scheme identifies the outliers in the sensed data. These outliers can show up in the data due to two causes: 1) because of a drastic change in weather conditions, and 2) because of erroneous report(s) by some sensor(s). Finding outliers in the data obviously depends on what is defined as the “normal” state. After a good model has been found, the “normal” state can be defined as the values lying within certain bounds of the model. This becomes a tunable parameter in our scheme, and setting different values of this parameter can reveal aspects of the model as well as the data being modeled.

Figure 7 shows the outlier detection performance of our scheme. The colored dots in the figure represent the error in the model applied to observed Pressure values. There are three colors assigned to the error values, Yellow, Orange, and Dark Red, from low to high values of error. The data generating the figure comes from 90 weather stations deployed in the Northeast United States, and their sensing reports considered

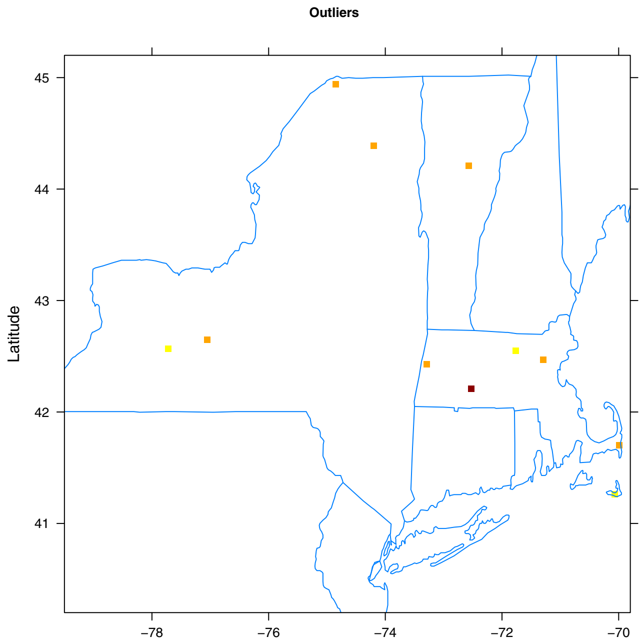

Fig. 7. Pressure Outliers Detection for June 1, 2011 Tornado.

in this figure lie between May 31, 2011 to June 2, 2011. On June 1, 2011, a devastating tornado passed between the positions at latitude = 42.10N, longitude = 72.75W and latitude = 42.10N, longitude = 71.99W. There is only one dot in the figure that is dark red, and it is the one representing the location at latitude = 42.21N and latitude = 72.53W, a weather station location that comes in the path of this tornado. This clearly shows the effectiveness of our model to detect outliers in the data.

## VI. Summary and Conclusion

In this paper, we have described a method for automatically detecting outliers in sensor values that uses the context of other sensor readings, but that is independent of specific domain knowledge. The sensor outlier detection scheme utilizes statistical modeling of the spatial parameters in the data. We have developed an optimization scheme for the choice of the model and applied this scheme in a weather sensing application. Our scheme shows promise in detecting outliers in the data, as demonstrated through our example application in the case of a tornado appearance over the area where we apply our model. A great benefit of our model is that it is not domain specific and can be applied in any application area with any types of sensors constituting a particular Cyber Physical System.

## Acknowledgements

We would like to thank Xianglong Kong for collecting the weather data used in this work.

## References

[1] P. Barford, J. Kline, D. Plonka, and A. Ron, “A signal analysis of network traffic anomalies,” in *Proc. of the 2nd ACM SIGCOMM Workshop on Internet Measurement (IMW)*, Marseille, France, Nov. 2002, pp. 71–82.

[2] R. L. Bashshur, T. G. Reardon, and G. W. Shannon, “Telemedicine: A new health care delivery system,” *Annual Review of Public Health*, vol. 21, no. 1, pp. 613–637, 2000. [Online]. Available: http://www.annualreviews.org/doi/abs/10.1146/annurev.publhealth.21.1.613

[3] A. Benveniste, “Loosely time-triggered architectures for cyber-physical systems,” in *Proceedings of the Conference on Design, Automation and Test in Europe*, ser. DATE ’10. 3001 Leuven, Belgium, Belgium: European Design and Automation Association, 2010, pp. 3–8. [Online]. Available: http://portal.acm.org/citation.cfm?id=1870926.1870931

[4] J. D. Brutlag, “Aberrant behavior detection in time series for network monitoring,” in *Proc. of the 14th Systems Administration Conference*, New Orleans, LA, Dec. 2000, pp. 139–146.

[5] A. A. Cárdenas, S. Amin, and S. Sastry, “Research challenges for the security of control systems,” in *Proceedings of the 3rd conference on Hot topics in security*. Berkeley, CA, USA: USENIX Association, 2008, pp. 6:1–6:6. [Online]. Available: http://portal.acm.org/citation.cfm?id=1496671.1496677

[6] Y. Gu, A. McCallum, and D. Towsley, “Detecting anomalies in network traffic using maximum entropy estimation,” in *Proc. of the 5th ACM SIGCOMM Conference on Internet Measurement (IMC ’05)*, Berkeley, CA, Oct. 2005.

[7] M. Ilic, L. Xie, U. Khan, and J. Moura, “Modeling future cyber-physical energy systems,” in *Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*, Jul. 2008, pp. 1–9.

[8] A. Lakhina, M. Crovella, and C. Diot, “Characterization of network-wide anomalies in traffic flows,” in *IMC ’04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, Taormina, Italy, Oct. 2004, pp. 201–206.

[9] E. A. Lee, “Cyber physical systems: Design challenges,” *Object-Oriented Real-Time Distributed Computing, IEEE International Symposium on*, vol. 0, pp. 363–369, 2008.

[10] ——, “Cps foundations,” in *Proceedings of the 47th Design Automation Conference*, ser. DAC ’10. New York, NY, USA: ACM, 2010, pp. 737–742. [Online]. Available: http://doi.acm.org/10.1145/1837274.1837462

[11] R. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, “Cyber-physical systems: the next computing revolution,” in *Proceedings of the 47th Design Automation Conference*, ser. DAC ’10. New York, NY, USA: ACM, 2010, pp. 731–736. [Online]. Available: http://doi.acm.org/10.1145/1837274.1837461

[12] L. Sha, S. Gopalakrishnan, X. Liu, and Q. Wang, “Cyber-physical systems: A new frontier,” in *Machine Learning in Cyber Trust*. Springer US, 2009, pp. 3–13, 10.1007/978-0-387-88735-7-1. [Online]. Available: http://dx.doi.org/10.1007/978-0-387-88735-7-1

[13] S. Shanbhag and T. Wolf, “Massively parallel anomaly detection in online network measurement,” in *Proc. of Seventeenth IEEE International Conference on Computer Communications and Networks (ICCCN)*, St. Thomas, USVI, Aug. 2008.

[14] A. Sorensen and H. Gardner, “Programming with time: cyber-physical programming with impromptu,” in *Proceedings of the ACM international conference on Object oriented programming systems languages and applications*, ser. OOPSLA ’10. New York, NY, USA: ACM, 2010, pp. 822–834. [Online]. Available: http://doi.acm.org/10.1145/1869459.1869526

[15] J. Stankovic, I. Lee, A. Mok, and R. Rajkumar, “Opportunities and obligations for physical computing systems,” *Computer*, vol. 38, no. 11, pp. 23–31, Nov. 2005.

[16] C.-W. Ten, C.-C. Liu, and G. Manimaran, “Vulnerability assessment of cybersecurity for scada systems,” *Power Systems, IEEE Transactions on*, vol. 23, no. 4, pp. 1836–1846, Nov. 2008.

[17] W. Wolf, “Cyber-physical systems,” *Computer*, vol. 42, no. 3, pp. 88–89, Mar. 2009.

[18] D. B. Work and A. M. Bayen, “Impacts of the mobile internet on transportation cyberphysical systems: Traffic monitoring using smartphones,” in *Proc. of National Workshop for Research on High-Confidence Transportation Cyber-Physical Systems: Automotive, Aviation and Rail*, Washington, DC, Nov. 2008.