Online Data Stream Mining in Distributed Sensor Network

IVETA ZOLOTOVÁ, TOMÁŠ LOJKA
Department of Cybernetics and Artificial Intelligence
Technical University of Košice
Letná 9/B 042 00 Košice
SLOVAK REPUBLIC
iveta.zolotova@tuke.sk, tomas.lojka@tuke.sk

Abstract: - Amount of end devices like sensors produce a much bigger amount of data. Collecting, transmitting and centrally processing amount of data from amount of devices is problematic. This paper deals with ways of reducing communication and distributed processing of data. With help of data mining and processing over real time stream of data we design an algorithm. This algorithm is based on data mining techniques, which we use to find important changes in the monitored environment. After finding interesting changes in the raw of data, the algorithm uses classification to create event in network which will share the new detected information. For communication we used Service Oriented architecture (SOA) and REST, which is described in designed architecture.

Key-Words: - Data mining, distributed network of sensors, entropy, monitoring system, sensor motes, Service Orientated Architecture (SOA)

1 Introduction

Nowadays we live in the world of complex monitoring system with many sensors around us. A lot of places, buildings or machines have own network of sensors. These sensors get a part of our everyday life. Actually we used them to immediately find, diagnose and solve problems. For example in health care are used sensors to monitor patient health state. Or we use sensors to identify traffic jams, to find the most effective way to work or home, finding incidents in workflow and so on.

Monitoring system is used to monitor environments or objects in environment through sensors and help people to be more effective in their work. Exactly the sensors are for assurance that unawares anomaly will be immediately noticed even predicted with the monitoring system. To identify or predict anomaly we need to have a real-time monitoring system. The system should monitor problems of itself (system resources) and monitored environment or objects [1]. Because of complexity of monitoring systems it usually consists of large number of sensors. The sensors produce huge volume of data. This data are from distributed sensors networks. Therefore sensor data might have a different consistency and they are dynamic [2].

Monitoring system contains sensors which are used for monitoring environment and its instances. Amount of data from sensors in monitoring systems is very big. It can be said that amount of data is enlarging with every day. Data from sensors are usually send directly like data streams to system which processes, uses and saves them. The overload of network is bigger and some not important data are sending over the whole network from sensors to master. This problem can be explained with “data rich but information poor” [3].

Bigger overload of network increases a time-latency of whole system. This is dangerous for real-time control and handling unawares errors, alarms and events might be unsatisfactory.

Geographically distributed and heterogeneous system produces huge volume of data which causes communication problems. This problem can be solved with reducing communication overload (II).

Monitoring system creates unique research opportunity to solve problems, how to design network of sensors, which will solve communication, lower power consumption, hazardous deployment, sensors failure tolerance, distribution of sensors and processing data from sensors.

2 Distributed and Centralized Sensor Data Processing

Researches in sensor network are trying to decrease number of transmitted messages, retransmitted messages, latency, but increase throughput of network. Today we can find concepts and solutions that describe centralized or distributed data processing from sensors.

Centralized data processing is implemented in monitoring system that has one centralized computing resource. This centralized computing resource collects data from all sensors in the network and process them. After processing them the centralized computing resource saves them to DB or a computing unit might sent fused data to the clients, which request the data.

Related work is Centralized Dynamic Clustering (Bajaber, F., Awan, I.) [4]. Nodes are group to cluster. Cluster is managed by cluster head. Cluster head is responsible to collect data from all nodes inside cluster [4].

Reference [5] cares about distributed sensors network like a network that consists of autonomous sensors. These autonomous sensors are placed over a large area [5]. Each node of the network has a computing resource, but the resources are limited. Opposite, a sensor measurement task is performed by large number of tiny computing resources (sensors) [5]. Benefits for this autonomy sensor network are:

- Overall monitoring system is more robust to failures [5].
- Sensors are small, have small computing power, but have less power consumption. It is easier to replace them and can be used in small spaces [5].
- Overload of sensor network traffic is spread out [5].

Distributed sensors data processing can be process explicitly (each node process its own data) or implicitly (nodes are connected and share resources). More used is explicitly algorithm, but disadvantage opposite the implicitly algorithm is that explicitly algorithm do not have better information result in describing monitored environment.

Explicitly algorithm does not cooperate with surrounded nodes. Therefore explicitly algorithm can't reach better monitoring results, but it decrease traffic overload.

Disadvantage of explicitly algorithm is in cooperation, fault tolerance and quality of measurements [4].

In our work we focus on sensors which have different physical principles and do not monitor the same instance.

Table 1. Difference between centralized and distributed sensor data processing

Attribute	Centralized	Distributed
Energy resource	Non-bounded	Bounded
Computation power	High	Low
Flow of data	Stationary	Continues
Data length	Known	Unknown
Update speed	Low	High
Passes	Multipass	Single
Time of processing	Non-real-time	Real-time
Memory resources	Large	Small

3 Distributed and Centralized Sensor Data Processing

Usually data are processed centrally and offline. In this paper our goal is to implement distributed processing of data and used computation resources of nodes in sensor networks. For implementation of reducing amount of sensor data we described methods and made a comparison of these methods.

3.1 Methods for Decreasing Amount of Sensor Data

Big amount of data which has no information value for control or prediction of system is wasting of resources and bring into system bigger uncertainty and variety (Heisenberg uncertainty principle). Options how to increase amount of information and decrease amount of useless data are:

1. Finding the best sampling frequency. Sensor will send measured data only if it is needed. This will decrease amount data, but do not decrease information value of data.
2. Using threshold to find only interesting data. Unnecessary information is cut with low threshold value. There might be used mean shift vector [6]:

$$M(\mu) = 1/n_x \sum_{i=1}^{n_x} (x_i - \mu_0) \qquad (1).$$

Where μ_0 is initial value, n_x is number of measured values with definite sampling frequency, x_i is measured value.

3. Sensors immediately process data and only information is sending from sensors. Sensors has implemented algorithm that gives output with description that is useful for actuators.
4. Sensor will send packages of measured data in compressed form. This means that only change of information is noticed.

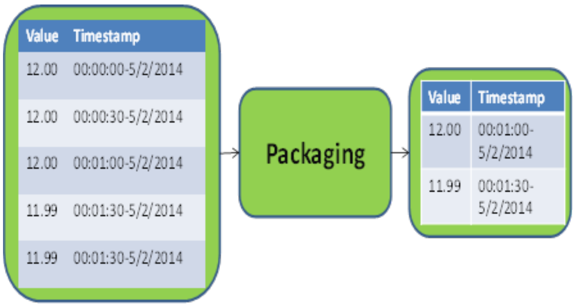

Fig. 1 Packaging data sending from sensors

5. Events based solution for transmitting data from sensors. Every change of information in sensors is presented like an event. This event calls master for listening and sends valuable information to the master network. The event reduce amount of data, but not decrease an information value.

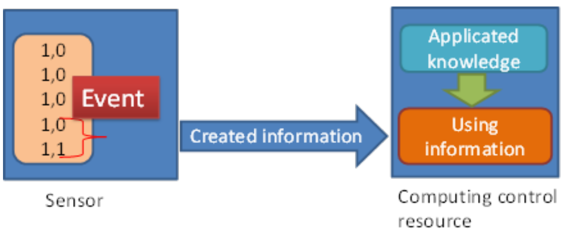

Fig. 2 Event based data transmission

6. Using data mining to reduce amount of data coming from sensors [4]. This option leads to reduce amount of information that are transferred from sensors, decrease overload of network and decrease a variety and uncertainty of whole system.

3.1 Comparison of Methods for Decreasing Amount of Sensor Data

In this paper we focus on reducing amount of data, finding and sending only valuable information and so reduce network traffic. We compare positive and limitations of mentioned options to reduce data traffic from sensors. We chose data mining in combination with time sampling, threshold and events.

Table 2. Options for reducing traffic from sensors

Opt.	Appearance		
	Description	Positive	Limitations
1.	Uses time sampling	Reduce amount of data	Involve data with not needed information
2	Uses threshold	Less data	Not all data has valuable information
3.	Produce information that the actuators can use.	Produce control information	Needs bigger computing resources
4.	Uses reducing of data by monitoring	Reduce amount of data	Not all data has valuable information
5.	Uses event to send information	Work with information	Need higher computing resources
6.	Uses data mining	Work with information	Need higher computing resources

4 Data Mining in Sensors

Sensors produce data. This data are processed to reach information. Then the information is used to reach knowledge. The knowledge says something about monitored environment or instance. This process is known as data mining. Data mining can be used for:

• Describe the data from sensors

• Predict data from sensor, exactly predict sensor output.

Data mining is process which is not conventionally design for sensors. Usually data mining works with static data types, has unlimited processing time, high computational power, stationary data flow, non-real-time response time, and has mulitipass algorithms [2]. These features of data mining are not feasible with sensors. Sensors has constrains like power, computing resource, memory, and communication [2].

Sensors quickly produce queues of data. This data queues should be real-time processed by data mining techniques. Hence, cyclic or more step data mining algorithm are inconvenient. To implement algorithm which will be able of reaction in so closer to real-time as possible is better to have one pass algorithm, which might be less accurate, but more effective. It can be implemented multipass algorithm which will operate on the background. But the end devices like sensors have low power and computing resources. Implementation of multipass algorithms will only slowdown whole data processing of and the reaction of end device will be more delayed.

Sensors are usually representing distributed system. In distributed system are data changes very fast. To create distributed network of sensors with good time response, we need to implement an online data mining. Despite, sensors computation power is limited (less like a central computer of sensors networks), sensors create distributed system of computation power.

5 Data Stream Mining Techniques in Sensors

Data mining in sensors has big potential. It can be used for monitoring, classifying, power saving, communication reduction, and prediction [2]. Using data mining in sensors creates distributed network of computing power and solve problem with growing amount of data, which is problem to store and index. In our solution we consider on communication reduction in distributed network of sensors.

Network contains sensors which dispose with low memory and do not have database system inside them. Therefore we should think about stream of data and low memory sources. This memory source represents a buffer with fixed length. The buffer type is FIFO and contains last n measured values. For this goal we use sampling to estimate entropy a_n.

$$S = \langle a_1, a_2, a_3 ... \rangle \rightarrow S_i = \{a_1, a_2, ..., a_n\}, \quad n = \{1,2,...\} \tag{2}$$

Every time the buffed is full of values is consider as a frame of length n.

We identify process of data handling to reduce communication workload, which is depicted on Fig. 3. Our process consists of points, which are written below. Every point has its specific implementation and differ from traditional data mining procedure. It was corrected to satisfy wanted goal of communication reduction by decreasing not important data and transfer only information, to which is sensor sensitive.

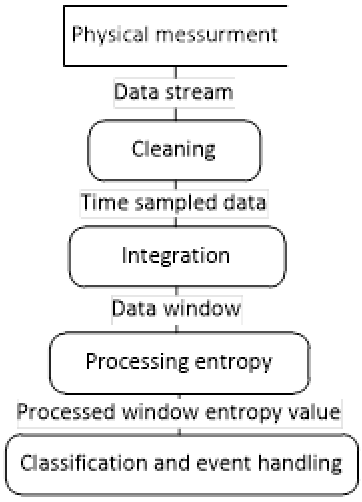

Fig. 3 Process of data handling to reduce communication workload

Physical measurement – presents a process of measuring external environment based on physical principles. Output from this process is created data stream.

Cleaning – presents a process of cleaning irrelevant data from data stream. Sensors produce a stream of data. Irrelevant data are cleaned by setting propriety sampling frequency. With data sampling can be reduce usage of costly computing and memory.

Regular next step is **data integration**. Because our explicitly definition of sensor network does not produce heterogeneous data we partially skip this step in this work. We integrate data only into frame. Frame represents specified number of measured values ordered into array.

Data transformation – presents a process of transformation data in suitable form for data mining. In our condition it is represent by rounding measured values to defined decimal place, which will speed up computing and decrease memory consumption.

Processing entropy – we choose algorithm for selecting data, which are important for analyzing frames information value.

We need to separate data with low information value from data with higher information value. One option was to choose, Shannon entropy non-conditional empirical entropy (3).

$$H(S_i) = -\sum_{j=1}^{n} p_j \log p_j \quad i, j = \{1,2,3,...\} \tag{3}$$

Where represent count of measured values in one frame and is probability of value in current frame. Next statement represents frequency of i-th value in frame. The empirical probability distribution is defined in (4).

$$p_i = \frac{f_i}{\sum_{j=1}^{n} a_j} \quad , j = 1,\ldots, N \qquad (4)$$

Another option was to choose entropy defined by Tsallis (1988) [5].

$$S_\alpha(p) = 1/(\alpha - 1)\left(1 - \sum_{j=1}^{n} p_j^\alpha\right),$$
$$0 \le \alpha \,,\quad \alpha \ne 0 \qquad (6)$$

Tsallis defined entropy which is additive and contains parameter α. This parameter enable to setup influence of probability distribution [7].

Because of fluent of data stream we chose one pass algorithm. This way we can save sensor computing power for next frame of streamed data. One pass algorithm decrease time computing and enable sensor to work in fast real-time (Measured delay was 1.5ms opposite each measured value).

Classification and event handling – present a process of evaluation of truly valuable information and representation. In this process we can compare already computed entropies to find which is over defined dash. If the frame has entropy which is higher or equal that (static/dynamic) defined dash, then the sensors create an event. Other words sensor has recognized unusual behavior in measured data stream. Using this event, the sensor tells about data stream anomaly to the master of sensor network or central PC. If the entropy has less value as defined dash the whole frame is erased.

Creating event will reduce overload of network traffic. Events are created only if valuable information will occur in sensor and every data sample will be send to central network computer.

Fact is, that lot of data, which might be useful in the future are erased. In this solution we propose network which is not for critical measurement and control in real-time yet.

6 Sensor Data Mining Architecture

We defined architecture which consists of sensors, gateways and central computer (Fig. 4). All nodes in architecture have some computation resources, what creates powerful network for data mining based on sensors with small computation power and memory.

6.1 Sensors

Sensor is used for monitoring patients. In our solution we have already implemented accelerometer sensor. Accelerometer measures vibrations, which are caused by movement of patient (for example we used accelerometer to identify patient movements during the patient was standing up from bed).

First of all we will describe sensor architecture (Fig. 4). Sensor measures vibrations. We propose sensor architecture to process data stream of measured vibrations. Whole data processing in sensor is described on Fig. 5.

Sensor measure vibrations and produce data stream of measured values. Then they are processed by computing unit. In the computing unit, the measured vibration data stream is sampled. Data stream is sampled to reduce amount of data. We need to reduce the amount data to save energy and computation resources. Sensor measurement dynamic and amount of wanted information allow us to sample input data stream.

Sampled data goes to the next part of computing unit (finding information). In finding information part of processing unit we implemented data mining algorithm to find valuable information in stream. Other data from stream are removed to free sensor computing unit memory. Finding information part of computing unit is based on entropy. We used Tsallis's definition of entropy. Detail algorithm is described in chapter 8.

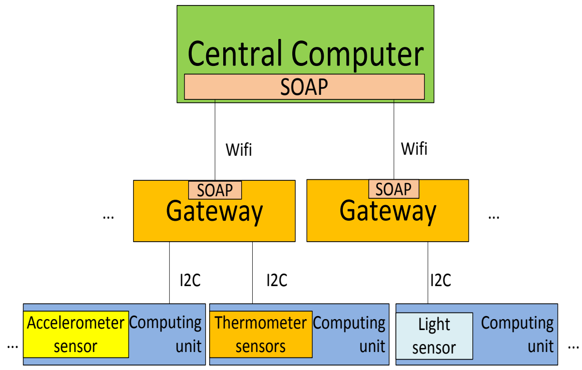

Fig. 4 Sensors network architecture

Process of data filtering is followed by process of classifying information. Information is classified into groups. Every group creates events with defined priority level. Detail algorithm is described in chapter 8.

Outputs from computing unit are events. Special parts of sensors are Conditions and Working storage. Sensors usually consist of unit for monitoring sensor system and unit for controlling system [8]. In our sensor architecture (Fig. 5) is monitoring system represents by (Stream processing, Event, Working storage) and was described above. Unit for controlling system is depicted with thicker border on Fig. 5. Controlling unit is presented by Conditions block. It is static block of sensor data which is used to initialized, setup, storing data mining rules, and defines calibrations attributes.

Working storage is used for storing working data. During communication are important results saved to Working storage. Then they are used in process of finding information or classifying information [1].

Process of classification has an information input and event output. This unit defined the importance of information and classifies it into defined event classes. We have already defined only two event classes:

1. Alarm class – this class represent information which contains critical data. This group is defined by rules which are stored in sensor Condition block.
2. General class – represent a class where information from sensor does not represent any critical situation.

Rules for classification was set experimentally and saved in Condition block in sensor. Rules can be edited, because Condition block presents a unit for sensor controlling system.

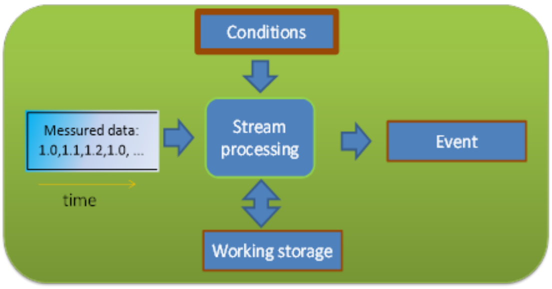

Fig. 5 Sensors architecture

6.2 Gateway

Gateway represents communication node, which connects network of sensors and Wi-Fi network. Gateway is part of monitoring distributed sensor network which integrates:

• Communication part for connecting two different networks [11].

• Logic part represents implicitly sensor data processing [11].

• Hardware part creates circuit for connecting sensor and Wi-Fi adapter, computation resource and memory [11].

In our solution we used SOA to present communication with central PC. Service is running on central PC. Clients (gateways) send requests to service. Service accepts requests process them and return an acceptation message to gateway.

The best way how to send message in our option was to use handshake. Therefore gateway waits for handshake of central computer [11]. If gateway will receive handshake then the message will be erased from queue waiting for message. After the handshake will not come, the message will be send again. Handshake solves network errors and guaranties that message about event will be delivered and accepted by central PC [5].

6.3 Central PC

Central PC is PC that collects data form whole network and process them. Central PC hosts a service (SOA) and has integration potential in satisfying communication with other central hosts or other systems [12].

Central PC might be also presented in cloud and information will be process, and stored inside cloud (Fig. 6). Cloud has big computing power to do data mining with sensor data, but our purpose was to reduce communication and sends only messages with important information. This should solve communication bottleneck. Therefore this implementation reduce commination overload and create architecture for a real-time sensors data processing. This proposed architecture may contain more sensors and gateways per one central PC due communication overload reduction.

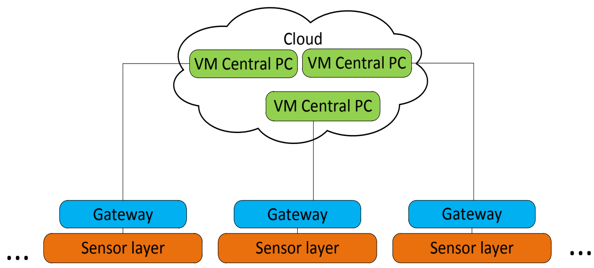

Fig. 6 Proposed sensor architecture implemented in cloud

7. Sensor Data Mining Implementation

In our solution we implemented Tsallis entropy to compute changes of information in data stream. Because Tsallis entropy operates with stream of defined length we create a frame. Our frame contains defined number of measured values. Every new value or values is/are written in array and the last value/values is/are removed. Range how the sensor will computes with new values and removed old is defined inside computing algorithm and it is configurable due configuration system of sensor.

After new values in frame are written, the computing unit starts to compute entropy. If computed entropy is less, there is no big change of information in selected stream. To be more precisely we thing about options how to build frame of data and how to not overview important information in data stream.

Our solution is based on floating frame with fields. In the configuration unit is set how values should be inserted into frame and how values should be removed from the frame. With this definition we defined a sensitivity of information change in stream.

Another part of sensitivity is controlling if the frame reach upper dash for creating event. This is used for classification, if selected value will create an event. If the computed entropy is upper than a defined dash, the sensor computing unit will start to classify data in the stream to create appropriate type of event.

Tsallis definition of entropy allows setting parameter α. This parameter was experimentally defined to value "0.9". Using the reference [8] we choose Tsallis entropy to calibrate sensitivity of computing. If the parameter α is positive then sensor will indicate values that occur often. If the α if positive then the sensor will catch measured values that happen seldom [8].

7.1 Floating Frame

In our solution we care about implementation of floating frame, which we used to process entropy. Our goal was to not overview important changes of information, but use entropy to find the changes of information.

The Frist variant for implementation of floating frame was to create a frame from every first *n* (for example 5) values. But there was a problem depicted in Fig. 7. If the frame has not implemented changes correctly into the frame, than the entropy will have the same value and no information will be recognized.

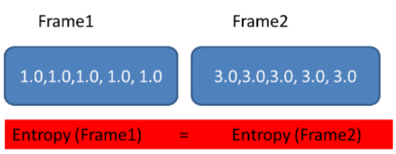

Fig. 7 Entropy problem I

The second variant was to create overlaying frames from data stream. In this variant has a problem to identify the change of information (Fig. 8). The value "8" will be not recognized as a change of entropy, therefore no information value will be catch.

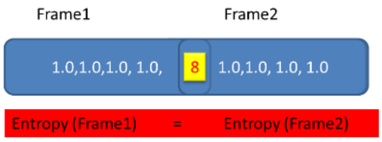

Fig. 8 Entropy problem II

The third variance is based on floating frame with fields. In the configuration unit is set how values should be inserted into frame and how values should be removed from the frame. With this definition we defined sensitivity on changing values in the stream. And another part of sensitivity is controlling, if the frame reach upper dash for creating event.

In our experiments we reach better performance with last variance, because this variance offers better tuning and is more adaptable to dynamic of monitored processes. Also in this variance we noticed consumption reduction of computational resources.

8 SOA for Communication

To create uniform connection we chose SOA architecture and REST. SOA interface is based on client/server architecture [13]. SOA does not differ between wire and wireless communication. It can do differences only between interfaces. But in our option we chose wireless communication. According reference [14], wireless connection offers implementation of benefits like no physical location constrain, better distribution, intelligent and self-healing [15]. These are our goals. On the other hand wireless communication can be slower for real-time control, less secure and reliable. The visualized data from sensor network might not clearly describe the monitored environment [16],[17]. Retransmission in sensor multi-hop network causes additional power consumption and delayed communication.

For industrial environment errors and erasures can cause a lot of problem in sensor network. With appropriate algorithms errors can be eliminated and erasures detected to satisfy the best transmission of information [18],[19],[20].

In implementation of wireless sensor network is appropriate to implement not only security to wireless communication, but also a prediction, which will find the unsatisfied differences in transmitted data. For less computation resources on gateways/sensors is better to implement, this prediction on central computer, which better implementation is future work.

8.1 Event Classification on Sensor Node

SOA with JSON message format helps to identify the type of transmitted message, its priority, and description of sender (sensor). We built solution with SOA to have structured messages for better representation.

9 Sensor Data Stream Mining Results

We implemented Tsallis entropy which we apply on data stream from sensor. Algorithm for computing entropy was implemented directly in the environment of sensor. First we reduce amount of information.

We definite sensitivity of frame to two and frame length to five. Next we identify entropy dash, to filter low changes of entropy caused by noise.

If the value of entropy is upper than defined dash we used weighted average of current processing frame to find the main important value of frame.

In the Fig. 7 are presented our results. We measure only x axis of accelerometer. The value was measured during patient standing up from bed.

We log sensor values and write them to the file. The values written in logs we used for later analyzing of tested algorithm. Every value has time-stamp and measurement voltage value from accelerometer x axis.

We used only x axis to test functionality of algorithm. After measurement we plot measured values from logs and compare it with values from data process with tested algorithm and computed entropy (Fig. 7). We can calibrate sensor to be more sensitive or less sensitive, to indicate more seldom or often occurred measured values, or set rules for classification. With this calibration we can tune amount of data which will be sent over network and also tune an information value of sending data.

We recognize time delay due the data mining computing. The time delay in average we round to 1ms. But the algorithm recognizes changes in data stream from sensor.

Our gateway functionality accepts data from sensor. After gateway receives value, the gateway will send it to central PC and wait for acceptation from sensor. Central PC functionality has not been implemented yet.

For our testing we used components from smartphone, which has accelerometer, processing unit, memory and Wi-Fi. In the future we would like to implement this solution on, sensor boards (microcontroller-based sensor board with set of sensors and communication interfaces) [11].

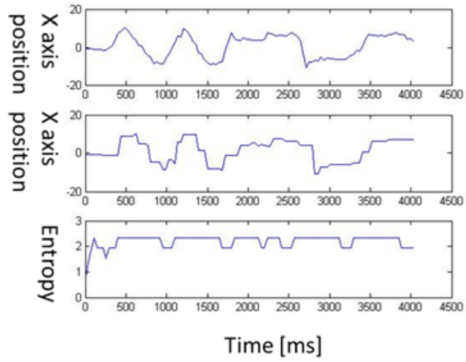

Fig. 7 Deviation comparison of real-time data from sensor, reduced data communication in real-time and entropy

9.1 Tsallis Entropy and Shannon Entropy

In the next experiments we try to confront Shannon and Tsallis entropy in creating events. We specified 0,09 change of entropy like sensitivity, when something is happening in the monitored environment. Tsallis entropy was set to be more sensitive to changes more than Shannon entropy. It was to be 56% more sensitive to regular changes than Shannon. The algorithm with Tsallis entropy recognises 15 more potential events to be classified than Shannon entropy. With bigger sensitivity is the number of potential events bigger, but it is safer to noticed more potential events and do decisions, which event should be created. With Tsallis equation can be sensitivity set up more according wanted sensitivity. In the next figure we want to present results by using Tsallis entropy.

In the Fig. 8. is represented deviation of accelerometer's X axis. The accelerometer real-time output values were used to find interesting moments in raw of data from monitored environment. We select a part of experiment deviation which is depicted in Fig. 8. We want to present a deviation of X value and reaction of our algorithm with using Tsallis entropy. We chose Tsallis implementation due the better time results in processing algorithm.

In the Fig. 9 is depicted real-time deviation of values. Fig. 9 describes changes of entropies in measured time in milliseconds. Values were recorded in gateway during real-time measurement and processing of data from monitored environment. In the Fig. 9 is presented entropy detection value which in case of quick changes has bigger values and shows that something unexpected is occurring in the monitored environment.

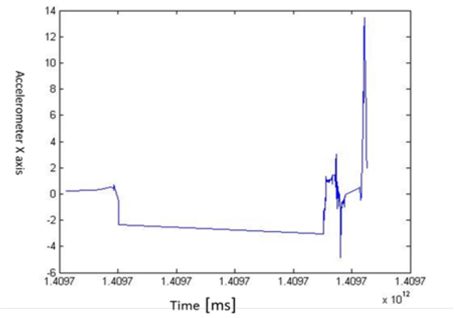

Fig. 8 Accelerometer X axis deviation in real-time. Time axis present measured time in milliseconds

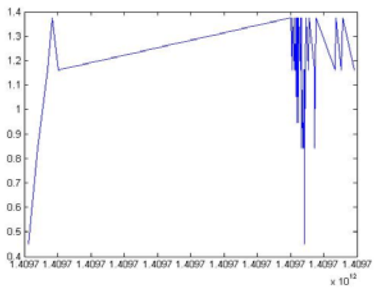

Fig. 9 Deviation comparison of real-time data from sensor by monitoring changes of entropy counted with Tsallis equation

Entropy detection value represents own named raw of values, which are outputs of Processing entropy block, depicted on Fig. 3.

We chose Tsallis entropy, but in the case of using Shannon entropy was recognised differences in deviation of *entropy detection value*, but with less accuracy to evaluate big and small changes in the monitored environment.

9.1 Computing Reduction Results in Implementing Tsallis and Shannon Entropy

In our experiments we also implemented Shannon entropy and Tsallis entropy equation. Shannon entropy contains mathematical function log, which has only software implementation in implemented SDK in our gateway device. Therefore the computation of logarithms function was longer than computation of Tsallis entropy equation. We measured not only the process of counting entropy, but whole process of updating/creating floating frame, counting new entropy, mean value of frame and classification. We pick some time differences between records which we programmatically implemented into gateway. Picked times was subtracted to find differences. According this we calculated 66 ms computing time for Shannon entropy. In processing Tsallis with defined parameters, mention in chapter 8, we measured in our conditions 43 ms. This allows us to reach better sensitivity to more dynamic measurement of processes and better adaptability for sensing level and communication reduction.

10 Future Works

In the close future we would like to test data mining techniques and find option how used this techniques over real-time data stream, tune data mining techniques for real-time deploying in sensor networks.

The global goal is to use these approaches and other methods of artificial intelligence in communication reduction and classification of event and create an intelligent behavior of sensor nodes. This intelligent will help to create M2M (Machine-to-Machine) communication between sensor nodes or actuator devices. Direct communication will increase safety, speed and reliability of monitored environment or control of processes.

Another wanted goal is implementation of these methods to create end-to-end connection between higher level devices/computers, which process big amount of data. Sensor and control for actuators will be implemented on microcontroller, which will offer enough computation and power resource for realization simple, low cost, and reliable implementation [18]. This goal is oriented on implementation in industrial environment and control, building automation, home automation, and city infrastructure automation or for monitoring of large or small areas.

11 Conclusions

Our purpose was to find way how to reduce communication overload in distributed sensor network. To reduce the overload, we implement data mining techniques and event based communication. We are using Shanon and Tsallis enropy to find only valuable information in sensor measured data and remove data with no information value. We design network architecture, where the algorithm might be implemented. Advantages of this implementation are not only in reduced communication, but in better real-time data processing in central computer. This will leverage real-time data processing and taking decisions.

12 Acknowledgments

This paper was supported by grants KEGA-021TUKE-4/2012 (60%) and VEGA - 1/0286/11 (40%).

References:

[1] L. Wang, J. Zhang, Z. Zhang, G. Sun, and G. Chen, KD monitoring system: Design, implementation, and evaluation, *Computer Research and Development (ICCRD), 2011 3rd International Conference*, Vol.2, 2011 pp.120-124.

[2] A. Mahmood, K. Shi, S. Khatoon, and M. Xiao, Data Mining Techniques for Wireless Sensor Networks: A Survey, *International Journal of Distributed Sensor Networks*, Vol. 2013, Article ID 406316, 2013.

[3] W. J. Slotnik, and M. Orland. Data rich but information poor. *Education Week*, 2010.

[4] F. Bajaber, and I. Awan, Centralized Dynamic Clustering for Wireless Sensor Network, *Advanced Information Networking and Applications Workshops, 2009. WAINA '09*, 2009, pp.193-198.

[5] J. D. McLurkin, Algorithms for distributed sensor networks, *Diss. Department of Electrical Engineering and Computer Sciences, University of California*, 1999.

[6] A. Basharat, N. Catbas, and M. Shah, A framework for intelligent sensor network with video camera for structural health monitoring of bridges, *Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference 8-12 March 2005*, 2005, pp.385-389.

[7] C. Tsallis, R. Mendes, and A. Plastino: The role of constraints within generalized nonextensive statistics. *Physica 261A*, 1998, pp. 534–554.

[8] T. Maszczyk, and W. Duch, Comparison of Shannon, Renyi and Tsallis entropy used in decision trees, *Artificial Intelligence and Soft Computing–ICAISC 2008. Springer Berlin Heidelberg*, 2008. p. 643-651.

[9] A. Lall, V. Sekar, M. Ogihara, J. Xu, and H. Zhang, Data streaming algorithms for estimating entropy of network traffic. *ACM SIGMETRICS Performance Evaluation Review*. Vol. 34. No. 1. ACM, 145-156, 2006.

[10] A. Rajaraman, and J. D. Ullman, *Mining of massive datasets*. Cambridge University Press, 2012.

[11] A.-B. García-Hernando, J.-F. Martínez-Ortega, J.-M. López-Navarro, A. Prayati, L. Redondo-López, *Problem Solving for Wireless Sensor Networks* (Book style), Springer, 19.12.2008, pp. 34-54.

[12] I. Zolotova, L. Lacinak, T. Lojka, Architecture for a universal mobile communication module, *Applied Machine Intelligence and Informatics (SAMI), 2013 IEEE 11th International Symposium*, 2013, pp.61-64.

[13] K. Žáková, Z. Janík, Remote Access to RTAI-Lab Using SOAP, *Advances in Information Science and Applications, Proceeding of the 18th WSEAS International Conference on Computers (part of CSCC '14)*, Vol. 1, 2014, pp. 177-180.

[14] M. Zahran, Y. Atia, A. Alhosseen, I. El-Sayed, "Wired and wireless remote control of PV system" *WSEAS Transactions on Systems and Control*, Vol. 5 (8), 2010 pp. 656-666.

[15] J. Liguš, J. Ligušová, *Inteligent control networks*, elfa Košice, 2013, (In Slovak).

[16] D.G. Costa, L.A. Guedes, F. Vasques, P. Portugal, Redundancy-based semi-reliable packet transmission in wireless visual sensor networks exploiting the sensing relevancies of source nodes, *WSEAS Transactions on Communications*, Vol. 12 (9), 2013, pp. 468-478.

[17] P. Peniak, M. Franeková, P. Lüley, Possibilities of control and information systems integration within industrial application area, *International Journal of Engineering: Annals of Faculty Engineering Hunedoara. Rumunsko. Tome X (years 2012)*, 2012 ISSN 1584-2665, s. 173-176.

[18] P. K. Pendli, M. Schwarz, H. D. Wacker, J. Boercsoek, Wireless communication modeling for safety related systems, *Naun International Journal of Circuits Systems and Signal Processing*, Vol. 8,2014, pp. 330-336.

[19] M. Zahran, Y. Atia, A. Al-Hussain, I., El-Sayed, LabVIEW based monitoring system applied for PV power station, *12th WSEAS International Conference on Automatic Control, Modelling and Simulation, ACMOS '10*, 2010, pp. 65-70.

[20] J. Kocian, M. Tutsch, S. Ozana, J. Koziorek, Modeling And Simulation Of Controlled Systems And Technologies, *Industrial Control. In 4th International Conference on Advanced Computer Theory and Engineering*, 2011, Dubai, United Arab Emirates. Pages: 213-217. ISBN 978-0-7918-5993-3.