ISSN 2348 – 7968

Efficient handling of Big Data Analytics in Densely Distributed Sensor Networks

Rudresh.M.S[1], Shashikala.S.V[2], Dr.Ravikumar.G.K[3]

[1] P.G.Scholar, Dept. of CSE, B.G.S Institute of Technology,
B.G.Nagar, Karnataka, India

[2] Professor and Head, Dept. of CSE, B.G.S Institute of Technology,
B.G.Nagar, Karnataka, India

[3] Professor, Dept. of CSE, B.G.S Institute of Technology,
B.G.Nagar, Karnataka, India

Abstract

The elaboration of wireless sensor networks has reached a point where each specific node of a network may store and convey a massive amount of (sensor-based information at once or terminated time). Hence in the forthcoming future, densely linked, enormously dynamic distributed sensor networks such as vehicle-2-vehicle communication setups may hold even greater knowledge potency. This is often due to the increase in node complexity. Subsequently, data volumes will become a problem for traditional data aggregation strategies traffic-wise as well as with regard to energy efficiency. For that reason, in this paper we suggest to call such scenarios as big data scenarios, they pose similar questions and problems as traditional big data concepts and granting the major focus mostly on business intelligence difficulties. Consequently our scheme would be propose an aggregation strategy tied to technological prerequisites which enable the efficient use of energy and the handling of large data volumes in an open source Hadoop frameworks with single/multi clustered architectures. Together with, we demonstrate the energy conservation potential based on experiments with actual sensor platforms in a distributed context.

Keywords: *Big data energy consumption, Hadoop Scheme, Distributed Networks, Sensor nodes, Efficient power usage.*

1. Introduction

The total amount of user data (data payload) to be stored or processed doubles every two years [1]. This fact raises several problems regarding data management and time-critical data processing tasks. The permissible scenarios to handle these issues, researchers all over the world are concentrating their work under the topic “big data”. If we talk about big data research, we consider novel approaches for the processing of huge amounts of data from different, heterogeneous sources from various database platforms. Key problems include data dissemination, automated analysis, search strategies as well as the visualization and post processing [2]. Big data environments in a traditional manner deal with massive, centralized computing resources, e.g. high performance computing centers and high-speed storage systems. Typical scenarios focus on data mining scenarios, financial computing (fraud detection) and scientific data evaluation [3] as well as pattern recognition. The majority of research and development activities in this field focus on existing information in larger volumes than the amount of data usually handled with relational database systems [1]. Today, the actual research focus changes rapidly. Several big data projects deal with huge amounts of multi-dimensional data in randomly distributed systems.

Suitably, a different application context requires different strategies. For example, if we consider next generation driver assistance systems, Vehicle-2-Vehicle (V2V) or Vehicle-2-Roadside concepts, a large amount of sensor data is generated and needs to be fused and evaluated [4]. Additionally, such tasks require local preprocessing techniques for distributed scenarios, for instance the evaluation and classification of data received by imaging systems [5]. Besides automotive applications, further scenarios address advanced sensor and monitoring systems as well as smart metering approaches, which are operating in a highly integrated and connected environment.

2. Related works on Distributed Big Data

Over Related research work for big data in distributed systems correlates with delay tolerant network strategies (DTN) [6] and adapted concepts of data aggregation or data fusion [7]. In dissimilarity to traditional big data infrastructures, the relevant goal is to shrink the relevant data payload directly at the source (or in-network) instead of transmitting the entire raw data to the sink for long-term storage. Regularly, the key challenge here is not the amount of the locally measured sensor data from distributed nodes, but rather the large number of distributed subsystems (nodes) and the changing communication infrastructure between these nodes in distributed network. In the automotive environment, energy resources as well as buffer storages are not as strictly limited as in other mobile embedded sensor platforms. On the other hand, particular data processing approaches have to use the given resources efficiently. At this point, key parameters are the latency behavior, communication time as well as the communication range and the high level of mobility [8].

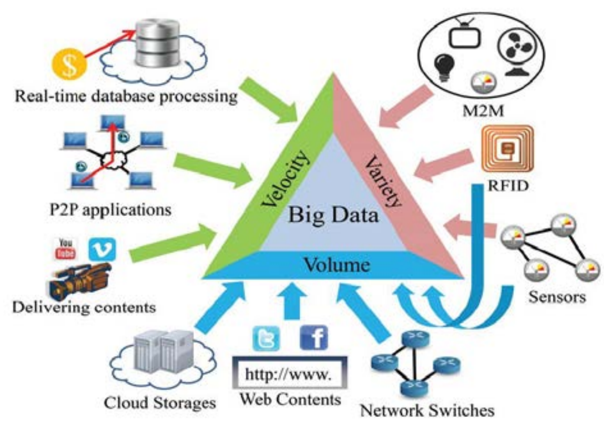


Fig. 1 Major trends of big data gathering

Additionally, vehicles as network nodes offer more possibilities for sensor usage and ultimately information they are able to deliver. In [9], hence the key elements of big data are identified through the "3V model" which mainly involves volume, velocity and variety in a big data strategy. Also pool of these points can also be transferred to future embedded, wireless sensor networks, as follows:

- → Volume: provides high node density and/or nodes having a high information potential.
- → Velocity: Now, there is already a high velocity of changing information in DSN in many application contexts.
- → Variety: It is designed to support various themes designedly heterogeneous DSN is a main goal of many data aggregation strategies (meaning a multitude of different network nodes).

Sometimes veracity is also included but then the focus is more on acquired business data than on actually measured information. Due to these justifications, it does not fit DSN very well. Thus, we are going to focus on the three points above. Accordingly, the idiom big data for compactly distributed, sensor networks in this paper shall embrace wireless, embedded sensor networks with data volumes (on individual node-level as well as on network level) larger than in traditional embedded contexts. Moreover, networks with a high number of embedded nodes or high node density shall be included.

Since resources like processing power, time and energy are limited in DSN; the objective of big data applications in this context should not be collecting as much information as possible from the sensor nodes. The assembly of as little data as possible and still satisfying the user's need for information is required instead of collecting unusually available data in end nodes. So the focus does not lie on data mining in large, existing databases but rather on deciding which readily available information from a network or cluster should be sent in which form to the client.

3. Energy-Efficient Concept in Big Data

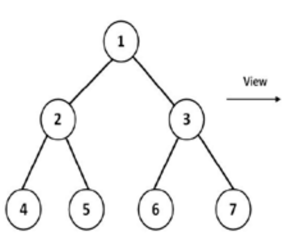
In broad-spectrum, sensor data collected within a sensor network may have different validity periods. Specific sensor values measured at a given point in time have a rather short validity period and are considered as one-shot (or even real-time) data. In order to differentiate this, sensor values which are fused over time may lead to higher level conclusions or implications that are of longer validity. For a time interval, Let us consider a VANET (Vehicular ad hoc network) where each vehicle on a street is a sensor node. Since the middle of the road is congested to some extent, so the cars which are part of the congestion may report congested to neighboring cars or a centralized institution (local information as seen by the individual node).

However, the global perception of the street remains congested, although the value of the (moving) nodes change depends on time. Since the ever increasing node numbers in distributed wireless sensor networks there is a need to retrieve local information and fuse it to global information as efficiently as possible. The expanding data volumes held in local node storages make in-network processing mandatory. From the available working platform, traditional proactive or schedule-based aggregation methods in distributed wireless sensor networks will reach their limits with the applications targeted by future sensor networks. These constraints will occur mostly with regards to energy efficiency and data volume. In established big data scenarios relational databases have to be succeeded by more apt concepts (such as NoSQL databases).

ISSN 2348 – 7968

In stand out, this very concept can come to the rescue in distributed, wireless sensor networks where established aggregation strategies do not fit the problems well anymore. Since the data collected by single nodes in future sensor networks increases gradually, in most common use cases the user (or client) does not need to extract single values from the sensor network, Instead of this pattern they demand a combination of values such as average of an area, highest value, and the interest for stable bank balance etc. [10].Gathering and storing every available dataset in an archive for later inspection often serves for node tracking and is therefore not within the scope of our research works. Apart from this, we mainly focus on the aggregation of combined value sets of node subsets within a distributed sensor network for immediate or short-term usage (such as regulating actor actions based on sensor measurements). In addition to this criteria, suppose there may be data collected by nodes which is irrelevant to the current application. If it is make over a filter this from the view on the network (by queries) allows to effectively reducing network traffic and storage necessity.

For that reason, in order to handle data volumes which are larger than the information stored in current distributed sensor networks, we recommend or rather reintroduce the concept of database-oriented data aggregation on the Hadoop frameworks. The basic concept of database-oriented approaches in wireless sensor networks is viewing or modeling the network as a virtual database table form, where each sensor node corresponds to a row. An illustration is given in Figure 2, where each single node of the sensor network has sensors (and therefore data) for temperature and pressure. Subsequently, each sensor of a node is represented by a column in this virtual table (which is referred as node attribute value). Attributes may be simple sensor readings (such as pressure, temperature etc.) Additionally, in the static routing case, every node knows its parent node established by a chosen routing approach. When using a dynamic routing, selection of route to take for a query may be determined during query execution.



Node-ID	Parent-ID	Temperature	Pressure
1		19°C	110kPa
2	1	21°C	110,05kPa
3	1	21°C	110kPa
4	2	20°C	
5	2	20°C	
6	3	19.5°C	
7	3	20°C	

Fig. 2.View on a sensor network as virtual table

Database-oriented systems for sensor networks already in existence are hadoop frameworks in a UNIX platform. We run the whole sensor node network using a real time network simulator version 2.0(ns2) to create a nodes for sensing the data from the distributed sensors and to store the collected data value in a sophisticated sink node, since the Hadoop framework is mainly built upon the Unix OS, which is also surrogate operating system for sensor nodes based on the C dialect nesC. Apart from this, it only supports specific hardware platforms since an extended tool chain is needed to connect with the real time simulator and the Hadoop framework. We have incorporated a connective solution using the shell scripts in any comfortable scripting languages like Ruby, Python etc, As far as our research, the code which has to be run by the sensor nodes (QueryProxy) has never been made publicly or officially available. Hence it is also one of the challenging tasks to collaborating two specific frameworks in one module.

4. The Proposed Strategy on Big Data via DSN

We think that the database-oriented approach fits the issues produced by big data in collective distributed sensor networks very well. Furthermore, from our point of view the systems in existence lack several functionalities needed for future networks, we propose the system developed by repository packages called: PLANetary. PLANetary is a light-weight, database oriented data aggregation system which is platform independent and focuses on energy efficiency. In this paper, we want to focus on the data aggregation strategies to handle specific large data items and therefore the used routing strategy is predefined and arbitrary. PLANetary does not enforce a single routing strategy but can support in finding optimal routes through the sensor network. Self organized, tree-based routing strategies for distributed sensor networks have been proposed as part of the nanett (nano system integration network of excellence) project.

In general, the database-oriented approach consists of two phases, similar to those in relational database systems [14]:

- → Query compilation / optimization
- → Query execution / runtime

In our approach we call these two phases query propagation and aggregation, since the query has to be delivered to all concerned network nodes after its statement. After that, information at the nodes has to be aggregated and fused in network and the results need to be sent back to the client. This process is shown in figure

ISSN 2348 – 7968

2.It should be noted that query propagation and aggregation may be executed in parallel. If a query reaches a leaf node the results are immediately sent back if the query conditions are met. Also the propagation (as well as aggregation) it is parallel since the queries (or its results) are transferred into the sensor network similar to a breadth first search. During the query propagation we try to select as few nodes as possible to send the query to. A node may be needed to execute a query when one or both of the following two statements are true for it:

- → The query conditions may be satisfied by this node and need to be evaluated.
- → The node is needed to forward results from the sub tree whose parent it is,

As it can be seen easily, intermediate nodes are only needed if their subtree contains at least one node where the conditions have to be evaluated. Henceforth, all nodes for which the second case is true are implied by the ones for which the first one is true. In consequence, we only have to determine all nodes for which the conditions may be satisfiable. After that, all intermediate nodes which are needed to transfer the results to the sink become clear. In order to select a node subset, we have to pre-evaluate the conditions of a query before its propagation to determine nodes where the conditions are impossible to satisfy.

There are different possibilities how we can determine this unsatisfiability. Each node in the network may have attributes which can be queried but are constant. For instance this could be the node id or the floor of a building the node has been deployed to. Since we support heterogeneous network structures there may be nodes which for example do not have a temperature sensor equipped. These nodes can be omitted from queries which select temperature values or pose conditions on temperature readings, since the evaluation result would be undefined.

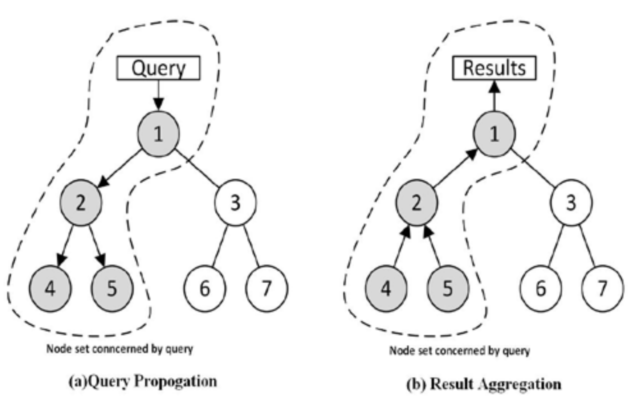


Fig. 3 The two phases of query execution in sensor networks

After deriving a restricted node subset, we may now construct the final query tree by adding the intermediate nodes of the multi-hop architecture for the static routing case. In the dynamic routing case, we may now construct a minimum spanning tree within the restricted node set (costs for instance based on node distances) to find the final query routing tree.

Naturally, in order to send a query into the network, the user needs a way of formulating it. With relational databases, queries are usually stated by database client software or manually by users using the SQL language. Database-oriented systems pursue a similar approach where data can be queried using a declarative, SQL-like query language enriched with syntax constructs useful for sensor networks. For the sake of simplicity and to emphasize its origin, we modeled the query language for PLANetary closer to the SQL dialect than that of COUGAR. In contrast to TinyDB, we replaced the SQL keywords SELECT and FROM with SENSE and AT, respectively. This is a similar concept as TinyDB's triggers but more versatile since each action command can have an arbitrary amount of parameters instead of just one as with TinyDB. These actuator commands have to be defined by the user and registered with the PLANetary core system on each sensor node.

PLANetary supports conjunctional and disjunctional links between condition statements as well as nested conditions (currently up to a depth of 2). This is an important improvement compared to TinyDB and COUGAR since these solely support non-nested, conjunctional links. Therefore, PLANetary allows stating more complex conditions to restrict the required node subset. Most certainly, this also may feel more natural for the experienced SQL user. This allows the combination of aggregation requests that require multiple queries in TinyDB into a single query with PLANetaryQL. This leads to improvements with regards to network traffic, energy consumption and the time needed for query propagation.

A sample is given in Figure 4. The possibility for nested queries to be stated allows for even more complex scenarios which may result in multiple queries with TinyDB or would be very hard to state at all. An example of a more complex, nested query is given in Figure 5. After the query has reached a leaf node of the query tree during propagation, the aggregation phase starts for this node. Therefore, we incorporated support for wake-up receiving technology into PLANetary. Wake-Up-Receiver modules are ultra low-power wireless communication modules which wait for just one signal, the so-called wake-up signal [15]. In practice, they accompany standard wireless communication modules (such as WiFi,

ISSN 2348 – 7968

XBee, and Bluetooth) which require much more energy during idle mode [16]. If there is no data transmission running, the sensor node may now deactivate the standard communication module.

SELECT AVG(light) FROM SENSORS WHERE floor=2

SELECT AVG(light) FROM SENSORS WHERE floor=5

(a)Two queries in Hadoop Frame operation

SENSE AVG(light) AT sensors WHERE floor=2 OR floor=5

(b) One Query in Simulation Interface

Fig. 4.Comparison of retrieving an average temperature value for two different floors in a building

SENCE AVG(light) AT sensors WHERE (id > 2 AND id < 10) OR (floor=2) GROUP BY floor

Fig. 5.A query in Hadoop frameworks with more complex conditions

Depending on the usage case even its main processing unit can be deactivated while waiting for the wake-up signal. Here, a time-based scheduling would be very prone to energy waste.

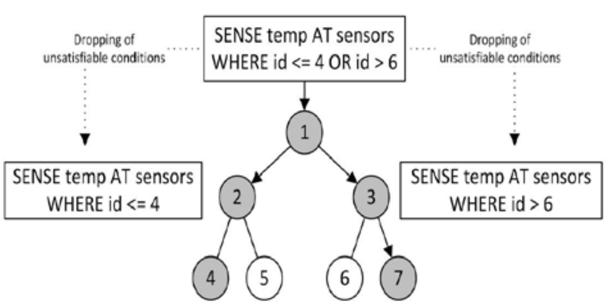


Fig. 6.Dropping of unsatisfiable condition parts in-network

Since it either would wake the nodes too often when no new queries need to be received or too seldom so that there is a noticeable delay between the query statement by the user and the arrival of results. A simple scenario is shown in Figure 6. This shows that the nodes of a subtree, which has dropped conditions, do not have to evaluate condition parts which can never be satisfied. With this, nodes which are further down in the subtree do not have to evaluate unsatisfiable conditions and therefore energy can be conserved as well as time (depending on the subtree size).

Consequently, this only works with static attributes or properties which have to be part of the routing table. Furthermore, this optimization needs additional resources on the nodes which have to check the conditions for unsatisfiability before forwarding the queries. Here, the static attributes of new nodes which are needed for the optimization would need to be synchronized often, supposedly wasting more energy than is conserved by query optimization. In contrast to TinyDB, where the virtual table must always be named sensors [11] and has no actual effect on the query processing (in fact it is entirely optional), PLANetary allows to define an arbitrary number of other virtual tables. Such a table definition consists of a table name and a predefined condition set which selects a subset of nodes in the network. There are two main advantages of virtual tables defined in this way.

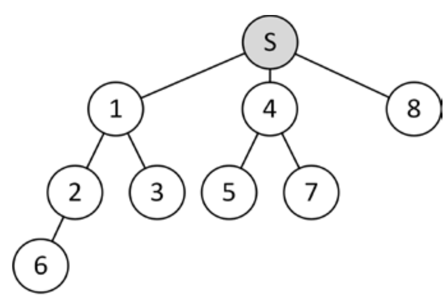


Fig. 7 Network topology used for the evaluation

Foremost, query text becomes shorter since conditions to restrict the query to a specific node subset (e.g. a specific floor of a building) are implied by the virtual table. Nevertheless, a query on a virtual table can also be further refined by using additional state of affairs. The other advantage is that definitions of predefined query sets can support the exploration and routing process by clustering nodes together which belong to the same or overlapping virtual table definitions. This concept can furthermore be enhanced to local node tables where queries collect data on a single node or a given cluster in their own virtual table, assigning a timestamp to each entry. These virtual tables may then later be queried themselves creating a historic query since the user evaluates past data. This can

ISSN 2348 – 7968

also further reduce the traffic in the network. As with TinyDB, queries may be marked as continuous and given a repetition interval. These queries will be propagated once and saved on the nodes to be re-evaluated on a timely base. This allows cutting the energy cost for queries, which are static and frequently executed.

5. Evaluation Criteria in Hadoop Frame

We tested our approach with a sensor network of nine nodes (where one node represented the data sink) of the PLANet type which supports a multitude of communication standards such as WiFi, XBee, Bluetooth. The tests were done as a proof of concept to show the energy conservation potential of the combination of database-oriented aggregation and wake-up technology. The network topology we used is shown in Figure 7 and represents a simple use case for home automation scenarios. Each node communicates by means of an XBee module and offers temperature sensor readings as well as the floor and room it was deployed to.

We wanted to compare timely scheduled data aggregation with the database-oriented approach in a hadoop environment to handle large volumes of data in reality. So on the one hand, we used a time-scheduled aggregation approach where each node sent its data during a given point in time (the demanded data would be collected and extracted at the sink). On the other hand, we used queries with increasing distinctiveness where no further data processing and filtering were required at the sink.

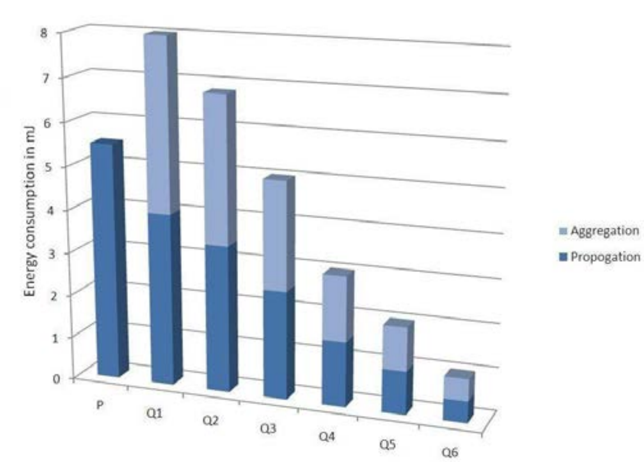


Fig. 8.Energy consumption for proactively sending all values (P) and the execution of more and more distinctive queries (only communication costs)

The queries range from selecting all values available (Q1) over getting the average temperature by floor (Q4) to selecting a single temperature value from a specific node (Q6). The energy consumption for each of these actions is given in Figure 8. For controlling the sensor network, starting and stopping the aggregation and checking the received results, we used the GREASE framework. GREASE is a generic sensor data aggregation and evaluation framework, which has been developed by an organization, and has been demonstrated in [18] and [19]. In the tested scenario, the energy needed to send a wakeup signal was negligible compared to the energy of the actual communication module.

We saw a general dependency of data volume (network traffic) on query distinctiveness. The more distinct a query is, the more restricted is the node subset on which it is executed and the less values need to be gathered. Therefore, usually the data volume decreases with increasing query distinctiveness as shown in Figure 9. Of course, the data volume required to execute a query can never fall under a certain point where the whole data transmitted is protocol data (marked as protocol overhead in Figure 9).

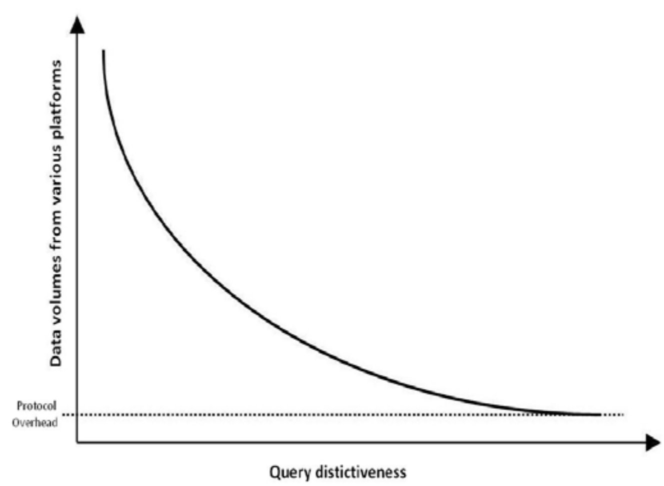


Fig. 9 General relationship between data volume to transfer and query distinctiveness

The energy consumed by the WuRx component was almost a thousandth of the energy of the actual communication module. The averaged energy consumption of the whole network for all queries we posed is given in Figure 10. The usage of wake-up technology lead to a massive conservation of energy in the given scenario (energy consumed is less than 30 percent). Naturally, the energy conservation depends on query distinctiveness (i.e. number of nodes to wake) and query frequency. But even with our test queries which were of widely different types (many values to single value) we observed a drastic energy conservation effect.

ISSN 2348 – 7968

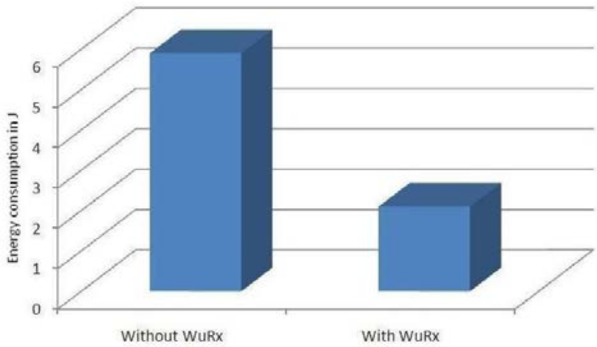


Fig. 10 Energy consumption of the whole sensor network during testing without and with using Wake-Up technology

6. Conclusion

In this paradigm, we introduced two cases where we see data aggregation in distributed, wireless networks as big data applications and modules. The first case represents sensor networks with a high node density so that the combination of the sensor data collected by each node is enormous. The second case deals with networks with high node complexity (such as vehicles) where each individual node holds a multitude of sensor data and deductions. In the interior of such networks, traditional aggregation approaches reach their limits with respect to energy-efficiency; the efficient handling of big data volumes in distributed, wireless sensor networks is growing more and more important. Therefore, if we speak about more than thousand (probably energy self-sufficient) network nodes, self organization and energy efficiency are the main points to consider and optimize.

The combination of database-orientation and wake-up technology yields a great potential in increasing the energy efficiency of distributed, wireless sensor networks and enables the efficient handling of big data applications in an Hadoop frameworks. Additionally, the proposed system may be applied on top of existing event driven data aggregation strategies. This would allow for a more detailed and flexible data acquisition for applications-specific events. However, the main focus of PLANetary lies on energy efficiency, due to its light-weight architecture, and easy incomparability into existing networks and hardware platforms. Additionally, there are more query optimization techniques to be considered.

Next steps are the simulation of highly dynamic networks with hundreds or thousands of nodes and the deployment of a demonstrator network with a higher number of nodes than shown in this paper. Hence the limitations of wake-up technology in such systems do also need to be defined and demonstrated. This includes finding the best compromise between longest possible sleep times with regard to query frequency and wake-up costs. The PLANetary core source code shall be made publicly available to open access in the future era.

Acknowledgments

This research is a combined effort of me and our guide of BGS Institute of Technology. I would like to gratefully acknowledge my organization, HOD, and our guide for their valuable suggestion and support.

References

[1] X. Wu, X. Zhu, G. Wu, and W. Ding, “Data mining with big data,”Transactions on Knowledge and Data Engineering, vol. 99, June 2013.

[2] G. Jung, N. Gnanasambandam, and T. Mukherjee, “Synchronous parallel processing of big-data analytics services to optimize performance in federated clouds,” IEEE Computer Society, 2012, pp. 811–818.

[3] Y. Demchenko, Z. Zhiming, P. Grosso, A. Wibisono, and C. de Laat, “Addressing big data challenges for scientific data infrastructure,” IEEE Computer Society, 2012, pp. 614–617.

[4] M. Rezaei, M. Sarshar, and M. M. Sanaatiyan, “Toward next generation of driver assistance systems: A multimodal sensor-based platform,” vol. 4, February 2010, pp. 62 – 67.

[5] S. Blokzyl, M. Vodel, and W. Hardt, “A hardware accelerated real-time image processing concept for high-resolution eo sensors,” September 2012. [Online]. Available: http://d-nb.info/1028174624

[6] K. Fall, “A delay-tolerant network architecture for challenged internets,”(SIGCOMM). ACM, 2003, pp. 27–34.

[7] M. Vodel and W. Hardt, “Data aggregation in resource-limited wireless communication environments - differences between theory and praxis,”(ICCAIS2012). Ho Chi Minh City, Vietnam: IEEE Computer Society, November 2012, pp. 282–287.

[8] M. Vodel and W. Hardt, “Data aggregation and data fusion techniques in wsn/sanet topologies - a critical discussion,”TENCON 2012. IEEE Computer Society, November 2012, pp. 1–6.

[9] D. Laney, The Importance of ’Big Data’: A Definition. Gartner, 2012.

[10] S. Madden, R. Szewczyk, M. J. Franklin, and D. Culler, “Supporting Aggregate Queries Over Ad-Hoc Wireless Sensor Networks,” in Proceedings Fourth IEEE Workshop on Mobile Computing Systems and Applications, 2002, pp. 49–58.

[11] TinyDB - A declarative database for sensor networks, Berkeley University of California Std. [Online]. Available: telegraph.cs.berkeley. edu/tinydb/doc/tinydb.html

[12] COUGAR, Cornell University Std. [Online]. Available: www.cs.cornell. edu/bigreddata/cougar/index.php

[13] Big Data Now: 2012 Edition, 2 ed., O'Reilly Media, 2012.

[14] V. Markl, Encyclopedia of Database Systems. Springer-Verlag Berlin Heidelberg, 2009, ch. Query Processing (in Relational Databases).

[15] M. Vodel, M. Caspar, and W. Hardt, "Critical parameters for the efficient usage of wake-up-receiver technologies," (ICCANS2011), 2011, pp. 100–105.

[16] M. J. Miller and N. H. Vaidya, "Minimizing energy consumption in sensor networks using a wakeup radio," WCNC. 2004 IEEE, vol. a, March 2004, pp. 2335–2340.

[17] www.ce.informatik.tuchemnitz.de/forschung/demonstratoren/planet/, September 2013.

[18] M. Vodel, R. Bergelt, and W. Hardt, "A generic data processing framework for heterogeneous sensor-actor-networks," International Journal On Advances in Intelligent Systems, vol. Vol. 4, December 2012.

[19] M. Vodel, R. Bergelt, and W. Hardt, "Grease framework – generic reconfigurable evaluation and aggregation of sensor data,"(ENERGY2012 / InfoSys2012), March 2012.

About Authors:

Rudresh.M.S received the B.E degree in Information Science from Visvesvaraya Technological University Karnataka in 2011, also M.S degree in Computing Technologies and Virtualization from Manipal University in 2013, and currently he is a post graduate student pursuing M.Tech in Computer Science and Engineering from B.G.S Institute of Technology under Visvesvaraya Technological University Karnataka.

He has presented 9 papers in National level conferences and also presented 2 papers in international conferences, He has published 2 papers in International Journals. His main research interests include cloud computing, and big data as well as computer networks and virtualization. He is currently doing his research project in Bigdata on distributed sensor networks.

Shashikala.S.V received her B.E degree in Computer Science from Mysore University, Karnataka in 1990 and M. Tech in Computer Science and Engineering from Visvesvaraya Technological University in 2005; she is currently working towards her PhD degree in the area of Big data over a multipath TCP.

She is a Professor and Head in the department of computer science at B.G.S Institute of Technology; B.G.Nagar Mandya and having totally 22 years of teaching experience. She has presented 4 papers in National level conferences and also presented 2 papers in International conferences, she has published 2 papers in International Journals. Her main research interests include computer networks, Big data and data mining.

Dr.Ravikumar.G.K received his Bachelor's degree from SIT, Tumkur (Bangalore University) during the year 1996 and M. Tech in Systems Analysis and Computer Application from Karnataka Regional Engineering College Surathkal (NITK) during the year 2000.He received his PhD degree in the field of Data mining and data masking during the year 2013 from Dr.MGR University Chennai. He has worked with SJBIT as Prof and HOD of Dept of CSE and ISE having around 14 years of Professional experienced which includes Software Industry and teaching experience. His areas of interests are Data Warehouse & Business Intelligence, multimedia and Databases. He has published and presented several papers in journals, international and national level conferences.