What-if Analysis in Wireless Sensor Networks Using Workflow Provenance (Position Paper)

Gulustan Dogan
Yildiz Technical University
Istanbul, Turkey
gulustan@yildiz.edu.tr

***Abstract*—Over the time sensor network readings become large datasets. A user reading the sensed data will not totally comprehend the readings without learning the path taken and understanding the dataset. As this is an accepted fact, the idea of including the provenance data while publishing sensor readings has been around for many years. First, the readings were annotated with data provenance such as reading time, node id. Since only keeping data provenance was not sufficient, the idea of storing workflow provenance arose. Workflow provenance illustrate the path taken to produce the readings and provenance models capture a complete description of evaluation of a workflow. As provenance is crucial for wireless sensor networks to support reproducibility, debugging and result comprehension, they have been an increasingly important part of wireless sensor networks. In our paper, we argue that sensor network provenance systems should support what-if analysis and debugging in order to allow users do modifications, see the results visually without actually running the workflow steps and be able to debug the workflows to figure out the anomalies in a wireless sensor network.**

Keywords*—*Workflow Provenance, Trust, Wireless Sensor Networks, Fault Tolerance, Distributed Intelligence, Self Organization.

I. Introduction

Wireless sensor networks (WSN) are used in many applications such as battlefield surveillance, air pollution monitoring, forest fires detection, biological, chemical attack detection. Due to their nature, wireless sensor networks are more error-prone than traditional networks. However in some critical sensor networks there is no central authority monitoring to find faults. Fault management and trust assessment are very crucial in order to sustain these real-time and mission critical networks. Information trust in a WSN depends on several factors such as its path, the trust of the source (sensor), time elapsed after the transmission, past behaviors, trust history of the nodes. Doing what-if analysis involves understanding the causal chains of past events which is provided by provenance. With provenance there are solid references of the phases data goes through and the event chains [2].

In our model provenance is kept for trust history and other information such as node location, node type, data type. Moreover provenance is used in order to find out causes of faulty behavior, to figure out the circumstances that will determine the connectivity of the network, to produce trusted data after elimination of the causes.

Previously we have done several work on provenance and sensor networks [3, 4, 5, 6, 8, 11]. In one of our work [3] we have built an architecture called ProTru for trust assessment using provenance. Our architecture is a generic architecture for all network types. Trust assessment of the nodes are done locally and network restructuring is done based on the trust calculations such as deleting untrusted nodes. Our trust metrics are data accuracy and data freshness. We decide about accuracy based on data similarity. In this paper, we modify this model for improved self organization and cognitive capabilities in wireless sensor networks. We extend our previous model so that it can remember the past network snapshots by storing dataflow provenance graphs. For wireless sensor networks remembering the past dataflows is very helpful for self organizing of the data. We came up with this idea by looking into nature. There are many examples of path recordings in the nature. One example is foraging behavior in ant colonies. Ants leave pherohormones on the paths they follow so that they can later remember the paths they took when carrying food to their nests. The central dataflow provenance repository gives the wireless sensort network the ability to remember the paths data followed as pherohormones in ant colonies.

This paper is organized as follows. Related work is presented in Section 2. In Section 3 and Section 4 we describe our architecture briefly and explain the central dataflow provenance model. Section 5 concludes the paper.

II. Related Work

There has been some work on fault tolerance in sensor networks. Paradis and Han survey fault-tolerance techniques for sensor network applications such as ESRT, PSFQ [1, 14, 17]. To our knowledge, there is not any work on using provenance specifically for fault tolerance in sensor networks. The interest of our paper differs from all of the above as we are using provenance locally to do what-if analysis and to capture network snapshots for self organization.

There is some work on cognitive networks. Traditional networks only deal with amount of data transmitted however cognitive networks also deal with content of the information delivered. It is closely related to provenance in the sense that provenance can keep information about the content [13]. In cognitive networks, elements in networks have states that are changing based on the content of the information received. This idea is close to our idea of nodes in a network that are in specific states at a given time and are behaving according to a Finite State Machine [13]. A cognitive model takes the data and converts it into intelligent information. Apart from the provenance research, there have been many ideas of increasing the intelligence within a multihop network. Intelligence can mean a range of behaviors from a sensor that turns on a light to much more complicated computing and

actions. We cannot relate all possible uses of the term here, we use it in a broader sense meaning the capability of the network to provide an immediate and detailed data trust.
There are several research threads that can be differentiated from the use in our architecture. One important common theme in making intelligent decisions within a network has been to better balance the traffic. Kelly provided a technique that makes use of local knowledge at a node to improve the traffic flow versus link capacity within the network [10]. Heo and Varshney made use of mobility to better position sensors in an area to improve coverage and energy efficiency [9]. Close to the ideas presented in our paper, Zahedi et al. have considered a two-tiered fault detection system for a sensor field that is collecting information [20]. Fusion node for a group of sensors weighs the usefulness of the inputs based on how accurate the result is compared to its likeliness for a misbehaved value. Our model is broader than the approaches listed as it is a general architecture applicable to different wireless network types. It is also more powerful as it is making use of provenance to create a distributed intelligence. Our approach is also novel in the sense that while storing rich trust and provenance information in vectors, we transmit one trust value over the network conserving network bandwidth utilization and reducing energy consumption. In addition, the two way communication (push and pull) between intermediate node and its children makes it possible to have an up-to-date trust picture of the network. Moreover by centrally storing the network pictures, our network gains a very valuable capability of remembering past flows.

Provenance has been studied in Sensor Network community. Provenance aware sensor data storage systems are proposed. In these systems, sensors collect provenance information of the data they are sensoring or the processes they are running [12]. Furthermore, provenance information associated with sensor data has been used in answering domain specific complex queries [15]. Park and Heidemann explore the need for data provenance in a information network to understand how processed results are derived and to correct anomalies [16]. In addition, provenance-aware Open Provenance Model based sensor systems have been implemented in different domains [7, 18]. There has been work presenting frameworks for provenance-aware information networks where data fusion methods are implemented [19]. However, to our knowledge this is the first work where what-if analysis using provenance in wireless sensor networks is discussed.

III. Network Model and Architecture

In this section, we briefly summarize our architecture ProTru [3]. More detailed theoretical explanation of the framework is presented in our paper.

A. Leaf Nodes

They are the source nodes (identified by a unique id). Leaf nodes collect and then disseminate information but do not receive information from other nodes. All nodes in our network as well as leaf nodes have vectors of reputation and provenance. $trust_{accuracy}$ and $trust_{freshness}$ values computed using the values in the vectors are forwarded along with the data while provenance and reputation vectors are kept at the nodes. In this system, nodeid is concatenated to the (data, $trust_{accuracy}$, $trust_{freshness}$) tuple forwarded along in order to have the dataflow information which is required for creating the graphs at the central provenance graph repository.

B. Intermediate Nodes

They are computationally more powerful nodes receiving information from a group of nodes, doing calculations on the received information such as fusing, and transmitting the information for the group forward. Intermediate nodes are identified by a unique id and they are the leaders of a group of leaves. Intermediate nodes receive two trust values along with the data; $trust_{accuracy}$ and $trust_{freshness}$. As the information is computed and fused, outlier nodes are found out by the intermediate node as the result of comparisons of several (data, $t_{accuracy}$, $t_{freshness}$) tuples received. For instance an intermediate node can fuse temperature data coming from two nodes, let's say $node_1$ sends the tuple (d_1, $t_{accuracy1}$, $t_{freshness1}$) and $node_2$ sends the tuple (d_2, $t_{accuracy2}$, $t_{freshness2}$) with trust values very close to each other. However the fusion node might realize that quality of d_2 is much better than d_1 and it can send a *decrease your trust value* message to the $node_1$. The algorithms used for data value comparisons and fusion are described in more detail in our paper [3]. Differently than ProTru, nodeid numbers of the sensor nodes which trasnmitted the data used in the fusion are concatenated to the (data, $trust_{accuracy}$, $trust_{freshness}$) tuple forwarded along in order to have the dataflow information which is required for creating the graphs at the central provenance graph repository.

C. Central Node

It is the top level of hierarchy which is a central station receiving values from intermediate nodes and calculating the final value. Intermediate nodes will send computed fused data, corresponding trust value and nodeid numbers of the tranmsitting nodes to central node. Due to distributed nature of our architecture, central node does not make decisions, it calculates the final result by taking the weighted average of incoming (data,trust) tuples. It also stores the coming result and dataflow graph at the central provenance storage after labeling it either bad or good based on a trust threshold.
ProTru architecture is extended by adding a central provenance repository of dataflow graphs. In our previous work, we designed a system called DustDoctor which troubleshoots a sensor network by doing mining on provenance graphs [11]. In this work we will use the same approach in DustDoctor to troubleshoot the wireless sensor network and find out the untrusted nodes. In DustDoctor we reported the faulty node by a GUI however the system we are proposing in this paper is much more powerful in the sense that based on the outcome the network will re-organize and heal itself.

IV. Provenance

In wireless sensor networks, time of creation of the leaf node data, the id of the Leaf Node creating the data, how much energy is left on the Leaf Node, how many times Leaf Node was turned of, the id of the central node leaf node is reporting to (dataflow) are part of the provenance data.
In most network systems, there is a central or distributed provenance storage system [12]. Provenance is stored for later

reuse and reference. We will transmit the workflow provenance to a central storage. We will keep the amount of provenance data at the required minimum level not to consume much energy for transmission.
All the workflow provenance data flowing over the network will be kept in the Central Storage. Workflow provenance records will be stored in the Central Storage. The historical data will also be available in the Cental Storage for network maintenance such as figuring out the leaf nodes that are silent for very long time, determining the group of leaf nodes that are misreporting. Final decision regarding the location estimation will be done at the Headquarter.
Provenance collecting and processing is very costly. However richer provenance is better for more efficient network restructuring. Therefore how much provenance to collect is an important choice.

A. What-if Analysis

Workflow provenance will help wireless sensor network in making cognitive decisions in case of a failure. An example scenario where workflow provenance will be useful is as follows. When a computation node is not receiving correct and sufficient input from the nodes it is data dependent, it should take an action to find a correct result. One possible action can be asking to another Intermediate node. This behavior can be implemented by a control flow statement such as "If the incoming edges do not send reliable data then ask to Intermediate node X".
There is also what-if analysis available in our system using the historical data stored in the Central Node. Mining the data, it can be found out how the network will respond if the target moves along a specific path.

- By what-if analysis, administrators can understand the impact of their decisions before they are actually made.
- Using historical provenance data stored in the system, what-if analysis can be done.
- By inspecting provenance graphs, network behavior under certain circumstances can be modeled.
- Solutions will be modeled and verified before deployment.

We make the workflow provenance store-able so "what-if" scenarios can be tested to further improve the debugging capabilities. One feature which will greatly improve the capability of having workflow provenance (graphs) will be to be able to mine the workflow provenance. That is, in effect, the current idea of workflow provenance as a graph is similar to a flow chart or an xlm diagram. Suppose instead of the nodes being labeled with the provenance, the system also stores the path followed to generate the data. The idea is that given the set of input values, and the sequence of the execution of the historical workflow provenance would give same outputs.
Below we will list some of the benefits of storing the workflow provenance. If for some reason the system (including the person reading the output) feels there are unusual results, one could manually request the workflow to be analyzed by graph mining algorithms. Or more powerfully one could add or subtract nodes (or branches from the provenance graph to examine the changes. An example is if one suspects a sensor that is collecting data that is being fed into the system to be deviating from accurate results, one could remove it from the workflow and re-analyze.

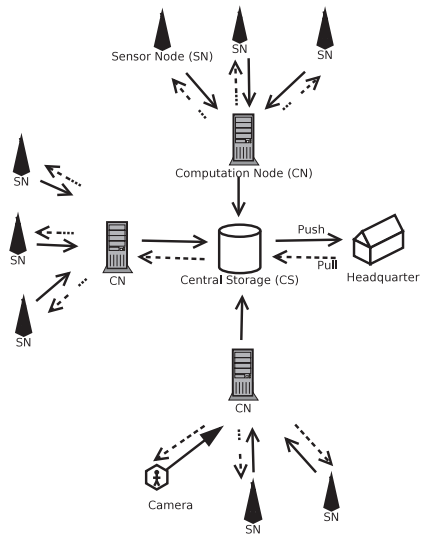

Figure 1: Network Architecture

provenancegraph	timestamp	trustvalue
	2011-01-19 03:14:07	λ
.....	.....	.....

Figure 2: Central Workflow Provenance Storage Scheme

Examples of Provenance Queries that can be run on workflow provenance:

- Find the groups that have been sleeping in the last six time intervals
- Find the intermediate nodes with an averaged trust value of less than ξ in the last time period
- Find the closest and most-trusted nodes to add to the group with decreasing trust value
- Find the intermediate nodes sending conflicting computation results with the rest of the network
- Find the intermediate node (closest one) to take control of the untrusted intermediate node group

Workflow provenance will be used in analyzing dataflow graphs of the wireless sensor network. For example in a case that an intermediate node is waiting for information coming from a leaf node but that node fails to send the information, the intermediate node will have the information of other possible leaf nodes that might have the same kind of information it

is waiting for. With this model we have a snapshot of the network at specific time intervals which we can refer to do some conclusions such as regrouping the fused leaf nodes, omitting a leaf node, changing the dataflow scenario.
In addition to support for alterations of processes and data provenance graphs should also support meta analysis such as concatenating provenance graphs which can be useful for researchers in combining the results of different experiments. In addition to this, for efficiently doing the what-if analysis, sub-graphs should support independent modifications to illustrate conditions and changed subgraphs should be recomposed. These features will give researchers the opportunity of discovering creative experiments.

V. Conclusion

Dealing with provenance in systems where data moves along such as wireless sensor networks is an open research area because it is hard to manage provenance when objects are mobile or distributed. Various solutions have been proposed to this problem but solutions are often domain-specific. A true solution will require architectural changes to the applications at the main levels such as hardware, network, operating system. Using provenance for what-if analysis is novel method with a low communication overhead compared to other approaches. Moreover transmitted data for what-if analysis is kept small making the model lightweight making our model efficient for wireless sensor networks. In addition support for multiple metrics makes the architecture flexible for different wireless sensor network domains such as industry, military, healthcare. Dataflow graphs become a reference and serve as a memory for self-organization of the network. Besides the use we have illustrated in this paper, the central provenance graph repository can be used for many purposes as various intelligence can be deduced from the recorded provenance graphs which is an area that we will explore as the continuation of this work.

Acknowledgement

This research was sponsored by the Technological Research Council of Turkey and was accomplished under Project Number TUBITAK 2232 114C143. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official polices, either expressed or implied, of the Technological Research Council of Turkey or the Turkish Government.

References

[1] Ö. Akan and I. Akyildiz. Event-to-sink reliable transport in wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 13(5):1003–1016, 2005.

[2] J. Cheney, S. Chong, N. Foster, M. Seltzer, and S. Vansummeren. Provenance: a future history. In *Proceeding of the 24th ACM SIGPLAN conference companion on Object oriented programming systems languages and applications*, pages 957–964. ACM, 2009.

[3] G. Dogan and T. Brown. Protru: Leveraging provenance to enhance network trust based on distributed local intelligence. Technical Report TR2011427, City University of New York, http://tr.cs.gc.cuny.edu/tr/, December 2011.

[4] G. Dogan and T. Brown. Using provenance in sensor network applications for fault-tolerance and troubleshooting. In *International Conference on Sensor Networks*. SENSORNETS, 2012.

[5] G. Dogan, T. Brown, K. Govindan, M. Khan, T. Abdelzaher, P. Mohapatra, and J. Cho. Evaluation of network trust using provenance based on distributed local intelligence. In *Military Communications Conference*. IEEE, 2011.

[6] G. Dogan, E. Seo, T. Brown, and T. Abdelzaher. Leveraging provenance to improve data fusion in sensor networks. In *SPIE Defense, Security, and Sensing*. SPIE, 2012.

[7] T. D. H. Eric G. Stephan and B. D. Ermold. Leveraging the open provenance model as a multi-tier model for global climate research. 2010.

[8] K. Govindan, W. X., M. Khan, G. Dogan, G. Zeng, K. Powell, T. Brown, T. Abdelzaher, and P. Mohapatra. Pronet: Network trust assessment based on incomplete provenance. In *Military Communications Conference*. IEEE, 2011.

[9] N. Heo and P. Varshney. An intelligent deployment and clustering algorithm for a distributed mobile sensor network. In *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, volume 5, pages 4576–4581. IEEE, 2003.

[10] F. Kelly and R. Williams. Dynamic routing in stochastic networks. *IMA Volumes in Mathematics and its Applications*, 71:169–169, 1995.

[11] M. Khan, H. Ahmadi, G. Dogan, K. Govindan, R. Ganti, T. Brown, J. Han, P. Mohapatra, and T. Abdelzaher. Dustdoctor: A self-healing sensor data collection system. In *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, pages 127–128. IEEE, 2011.

[12] J. Ledlie, C. Ng, and D. A. Holland. Provenance-aware sensor data storage. *Data Engineering Workshops, 22nd International Conference on*, 0:1189, 2005.

[13] S. Misra, S. Misra, and I. Woungang. *Selected Topics in Communication Networks and Distributed Systems*. World Scientific Pub Co Inc, 2009.

[14] L. Paradis and Q. Han. A survey of fault management in wireless sensor networks. *Journal of Network and Systems Management*, 15(2):171–190, 2007.

[15] H. Patni, S. Sahoo, C. Henson, and A. Sheth. Provenance aware linked sensor data. In *Proceedings of the Second Workshop on Trust and Privacy on the Social and Semantic Web*, 2010.

[16] J. H. Unkyu Park. Provenance in sensornet republishing. In J. Freire, D. Koop, and L. Moreau, editors, *Provenance and Annotation of Data and Processes*, volume 5272 of *Lecture Notes in Computer Science*, pages 280–292. Springer Berlin / Heidelberg, 2008.

[17] C. Wan, A. Campbell, and L. Krishnamurthy. Pump-slowly, fetch-quickly (psfq): a reliable transport protocol for sensor networks. *Selected Areas in Communications, IEEE Journal on*, 23(4):862–872, 2005.

[18] J. M. A. R. R. K. Yong Liu, Joe Futrelle. A provenance-aware virtual sensor system using the open provenance model. 2010.

[19] X. W. L. M. D. H. J. M. A. R. R. K. Yong Liu, Barbara Minsker. A new framework for on-demand virtualization, repurposing and fusion of heterogeneous

sensors. 2009.

[20] S. Zahedi, M. Szczodrak, P. Ji, D. Mylaraswamy, M. Srivastava, and R. Young. Tiered architecture for on-line detection, isolation and repair of faults in wireless sensor networks. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1–7. IEEE, 2008.