On the Data Freshness for IoT Traffic Modeling in Real-Time Emergency Observation Systems

Kemal Cagri Serdaroglu, Sebnem Baydere
Yeditepe University
Istanbul, TR
{kserdaroglu,sbaydere}@cse.yeditepe.edu.tr

***Abstract*—Internet of things (IoT) and fog computing based observation systems are gaining more importance as Internet becomes the main infrastructure to augment the pervasiveness in remote monitoring of the physical world. Considering the explosion in the number of connected ”things”, the increase of data traffic density on interconnection devices (i.e., IoT gateways) becomes an important problem for scalable real-time emergency detection and monitoring. Thus, data traffic analysis and modeling of fog services become an important research area to get more insights into real-time behavior of such systems. The outcomes of such analysis are important for prediction of IoT system behavior in a given network topology. In this paper, we elaborate on an architectural solution for periodic data acquisition from a wireless sensor network(WSN). To this end, we propose a publish/subscribe (P/S) based observation scheme which simultaneously interconnects clients to different kind of sensor devices over a fog layer service. Then, we examine the data freshness which is a critical traffic modeling parameter for real-time emergency observation. With using such scheme, we devise an analysis for understanding the behavior of the overall system in the context of data freshness. The results obtained from our experimental setup illustrate the appropriateness of freshness time calculation methods for obtaining the required service quality.**

I. Introduction

Monitoring and control of agricultural, industrial or in-home health-care emergency detection process necessitates periodic observation of a physical phenomenon by an ever-increasing amount of sensors and actuator devices. Those devices are typically involved in Low Power Personal Area Networks (LoWPANs) [1] [2] [3] or so called sensor networks. In order to provide a complete observation system, sensor networks interoperate with remote observation clients in WANs via interconnection devices which are fundamental inter-operation parts in Internet of Things (IoT) [4]. In addition, these systems should serve different kind of services with different Quality of Service (QoS) requirements by using several kind of sensors which have different states of observation in the same time period. Whereas such systems traditionally have been deployed using dedicated and proprietary protocols, a recent trend in the domain is towards adopting IoT technologies with Publish/Subscribe (P/S) manner [5] and using fog computing which is the paradigm based on determining sensitive and critical data on the edge rather than streaming and processing huge raw sensor data on the cloud [6].

Clients of a P/S based periodic observation system generally subscribe one or combination of the devices in a sensor network. These devices are called as sensor nodes and they generate virtual data representation of one or a combination of the physical phenomena. Generally, those devices publish data to the subscribed clients via IoT interconnection devices (i.e., gateways or proxies) such as P/S brokers [7] deployed on the edge network.

The emergency observation process may generate huge sensor data traffic on the IoT infrastructure, especially on the interconnection devices. Increase in the number of sensing devices deployed for an observation system is an important drawback for real-time emergency observation. In addition, in order to meet clients' SLA and real-time requirements of the system, a cache based [8] time stable and periodic emergency decision support mechanism should be used on the edge devices. Such systems generally favor a service-centric asynchronous push / push data integration scheme [8] for scheduling, resulting an emergency decision to be marked as old. Thus, data freshness rate becomes a crucial traffic modeling parameter to be observed in these systems and there is a need to examine freshness time calculation methods in terms of achievable freshness rates.

In this paper, we analyze the data traffic and freshness characteristics of a fog based emergency observation system which potentially can scale to large numbers of sensor devices. We consider a subset of the emergency detection scenarios which necessitate frequent acquisition of scalar sensor data with high sampling periods.

The organization of the paper is as follows: In Section II, we examine the state of the art solutions related to our study. In Section III, we introduce the P/S based data collection and emergency detection scheme for a fog proxy server which subscribes multiple clients to the emergency detection scenarios according to the demands of clients. In Section IV, we analyze the data freshness parameter and data freshness time calculation methods in a centralized fog server. In Section V, we give the results of our empirical study for data freshness analysis in which we observe the data freshness rates with different parameters. In Section VI, we conclude our study with elaborating on the outcomes and the future works of our study.

II. Related Work

Data freshness is studied in several Content Centric Network (CCN) and Opportunistic Network approaches with distributed manner. Jaber et al. [9] proposed a distributed and time synchronization based data freshness model for sensor nodes in CCNs. Lee et al. [10] developed a real time data freshness model for distributed networks considering global time synchronization. Systems adapting these models should consider network time re-synchronization and temporal refreshing which requires extra packet transmission, meaning high traffic load in the sensor network.

Several centralized data freshness models for low and high data sampling rates are also studied for data integration systems (DISs) [11], [12], [13], [14]. The general drawback of these studies is that they fail to meet the required scalability of IoT systems in real time applications.

We propose a fog-based solution where a P/S server caches on-the-fly sensor information in the fog layer rather than streaming data to the cloud. Our solution does not require network time synchronization in the sensor nodes, thus, no extra control traffic is imposed. We adapt the caching based data integration model to utilize more transient sensor data for real-time freshness calculations. As data freshness is critical in real-time emergency detection, we provide means of integration of freshness concept with our P/S scheme.

One of the most widely accepted standard protocols for P/S implementation is MQTT data transmission protocol [15], [16]. Although MQTT is designed to hide the complexity of the infrastructure, on publish process, the publisher sensor node should place a MQTT topic string in each message for determining its subscribers [17]. This is an additional overhead for real time performance of a resource constrained sensor node and overall system [18]. To get rid of this drawback, in our architectural model, sensors layer is unaware of the topics [19]. Our model only requires an identifier of the sensor node in the sensor data packet to determine the right sensor agent. Therefore, for publish operation, a sensor node in the sensors layer performs only periodic sensing and sending without the need of managing topics in itself.

III. System Architecture

The overall architectural model for proposed fog based emergency detection system on IoT backbone is depicted in Fig. 1. The gateway in the fog layer is the hub between clients and the sensor world and performs the P/S based emergency detection with several modules. In addition, this gateway uses time stable and periodic emergency decision making scheme based on fully asynchronous push / push data integration model [8]. The functional modules used in the gateway are given below.

Sensor Agent (S.A): An S.A. is the virtual representation of a sensor node in the P/S server. Each state of this agent represents a primitive event for emergency detection operation performed. S.A caches sensor node related information consisting of its arrival timestamp, last inter arrival time (IAT) and the sensor value(s) which can be processed directly from a client rule. For scalar sensor data, every sensor value in an S.A. represents the value of the raw sensor data acquired from a sensor node.

User Agent (U.A.): A U.A. interacts with a client out of the fog domain. It subscribes the client to the server, gets the client rule stored in the database and prepares it for emergency detection operation. It interacts tightly with an emergency detector (E.D.) and sends response generated by its E.D. to the client.

Emergency Detector (E.D.): An E.D. uses a timer which expires in every Client Period (CP) defined in the client rule. In every expiration of the timer, an E.D. performs the emergency detection operation. It uses client defined rule and controls the states of S.A.s associated to it for deriving an emergency situation. In addition, it determines the freshness of the derived result.

Sensor Connectivity Service: Sensor connectivity service operates between the gateway and the sensor network in the fog domain. It acquires raw sensor data from the communication interface of the sensor network. For further operation, it controls the type of the raw sensor data acquired. If it is data from a scalar sensor, it directly forwards the data to the S.A.

IV. Data Freshness Models

With our P/S based emergency detection scheme discussed in Section III, we aim to achieve a time stable and client defined period (CP) based emergency detection process which minimizes the effect of IAT fluctuations of the sensor network. However, this process could result in lower data freshness rates if the sensor network connected to the fog has high IAT fluctuation characteristics.

Fig. 2 illustrates the time flow charts related to emergency decision freshness analysis in detail. The first line in the figure shows the time flow and the lines which are perpendicular to it illustrate the expiration of the CP and the moment of emergency decision making. If any primitive event is cached in S.A. within the time span depicted by dark gray label, it can be considered as fresh by an E.D., else it can be considered as old. The time length of this time span is the freshness time (FT). The second line in the figure shows one CP time interval and FT in detail.

In this study, we elaborate on freshness time calculation methods which are used to obtain maximum data freshness rates (FR). We classify these methods in two categories. The first category is composed of stateless methods where freshness time calculation is not based on any historical IAT or FT values. The second category contains stateful methods where data freshness time calculation is based on historical FT and on-the-fly IAT values.

A. Freshness of Emergency Decision

Before examining FT calculation models, we give some definitions about freshness of emergency decision that we used in our analysis.

Fresh Emergency Decision : The emergency decision which is marked as fresh by an E.D.

Old Emergency Decision : The emergency decision which is marked as old by an E.D. It is uncertain data for the time domain of the time stable emergency detection system.

Pessimistically Fresh Emergency Decision : An emergency decision can be considered as pessimistically fresh, if

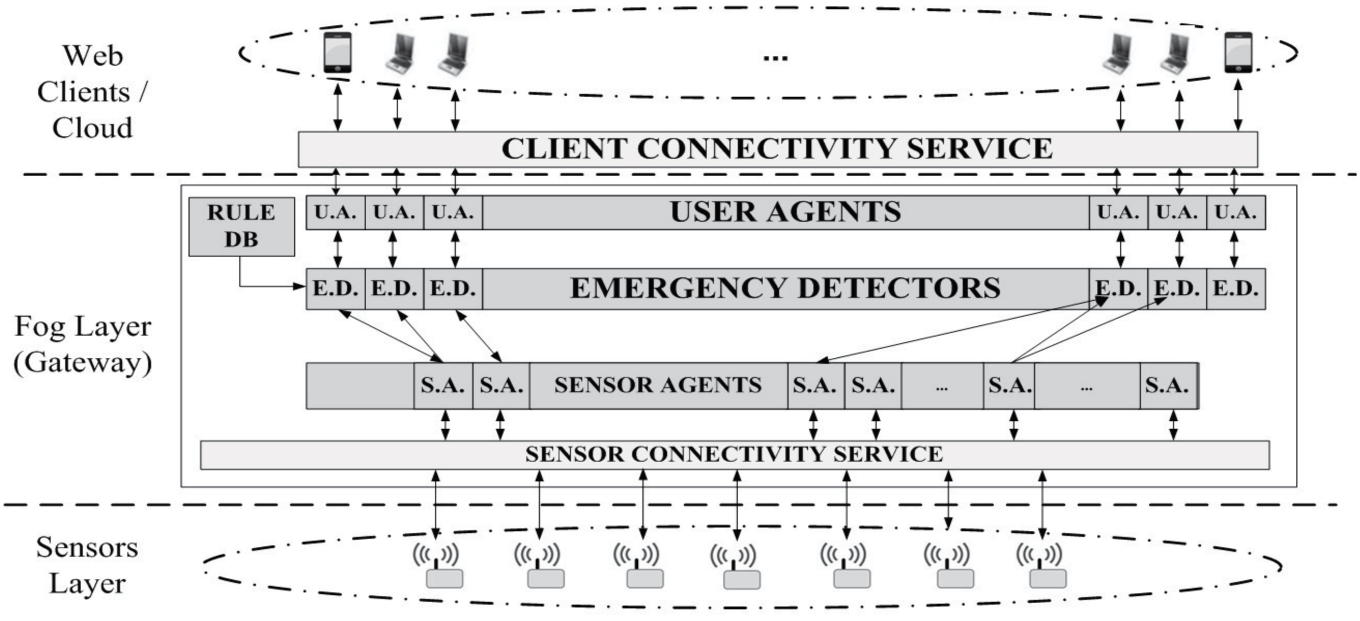

Fig. 1. Architectural View and Layers of P/S based Emergency Detection System

an E.D. detects all primitive events related to the emergency decision as fresh.

Pessimistically Fresh Emergency Decision with a percentage n : If an E.D detects that at least n percent of the primitive events related to its emergency decision are fresh, this emergency decision is considered as fresh with pessimism percentage of n. For example, if n=25 and the number of sensor nodes (N)=8, marking at least two of the primitive event in the related S.A.s as fresh is enough to determine the emergency decision result as fresh.

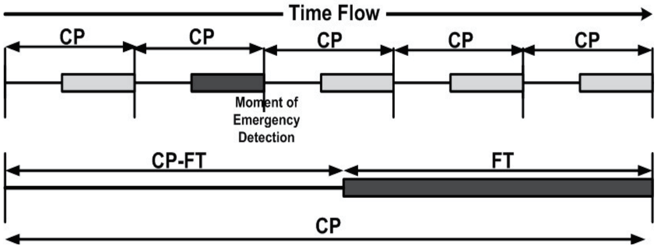

Fig. 2. Time Flow Chart of Emergency Detection, Client Period and Freshness Time Illustration

B. Freshness Parameters

We use the following parameters in our analysis of the data freshness models.

- h : The gain coefficient for the deviation of historical FT.
- g : The gain coefficient for the new IAT value.
- $Me_{i,j}$: i^{th} IAT measurement read from j^{th} S.A. related with emergency decision.
- $Err_{i,j}$: Difference between the measured $IAT_{i,j}$ value and A_{i-1}. It is calculated with Equation 1.

$$Err_{i,j} = Me_{i,j} - A_{i-1} \quad (1)$$

- A_i : Smoothed IAT value in the i^{th} data freshness time calculation.
- $A_{i,j}$: Smoothed IAT value in the i^{th} FT calculation for j^{th} S.A. related with emergency decision. The calculation of this value is obtained from Equation 2.

$$A_{i,j} = A_{i-1} + g * Err_{i,j} \quad (2)$$

- D_i : Smoothed mean deviation of IAT values obtained in i^{th} data freshness time calculation.
- $D_{i,j}$: Smoothed mean deviation of IAT values obtained in i^{th} FT calculation for j^{th} S.A. related with emergency decision. The calculation of this value is obtained from Equation 3.

$$D_{i,j} = D_{i-1} + h * (|Err_{i,j}| - D_{i-1}) \quad (3)$$

C. Stateless Data Freshness Time Calculation Methods

We examine two stateless data time calculation methods in our study. In the former, any E.D.in the server marks the decision as old or fresh according to a fixed FT in a client topic. In the latter, we calculate FT to obtain maximum data freshness rate with considering the data traffic characteristics of the sensor network connected to the fog. So, for this method, we consider IAT distributions of the sensor data coming from the sensor network to the server. We use the time guaranteed model presented in our previous work for this method [20].

D. Stateful Freshness Time Calculation Method

Data freshness time can be calculated with an exponential averaging method using on-the-fly IAT values. For this method, we adapt the Jacobson's stateful TCP Retransmission Timeout (RTO) calculation [21]. Steps of the method are listed below.

(Step 0) : E.D. sets its FT_0 to CP, A_0 to 0 and D_0 to $CP * h$ initially.

(Step 1) : For each S.A. required for emergency decision, at i^{th} iteration, P/S server uses IAT value in j^{th} S.A. as $Me_{i,j}$ and calculates $Err_{i,j}$ value using Equation 1. Then, the server calculates $A_{i,j}$ using Equation 2, $D_{i,j}$ using Equation 3 and $FT_{i,j}$ using Equation 4.

$$FT_{i,j} = A_{i,j} + (1/h) * D_{i,j} \tag{4}$$

(Step 2) : At i^{th} iteration, an E.D. finds an S.A. which gives the maximum FT value ($FT_{i,max}$) from all candidate $FT_{i,j}$ obtained in Step 1. max is the index of the S.A. which gives maximum FT. Then, it sets FT_i to $FT_{i,max}$ and updates A_i to $A_{i,max}$ and D_i to $D_{i,max}$.

V. Experimental Setup and Tests

We conducted some tests to evaluate the data freshness models described in Section IV. We realized the architectural scheme discussed in Section III in our testbed as illustrated in Fig. 3. Table I gives the parameters used in the tests. In order to focus our analysis on data freshness models and eliminate the impact of rule complexity on the performance, we defined simple client rules in which only sensor data arrival times and IAT values stored in S.A.s. are sufficient to generate an emergency decision.

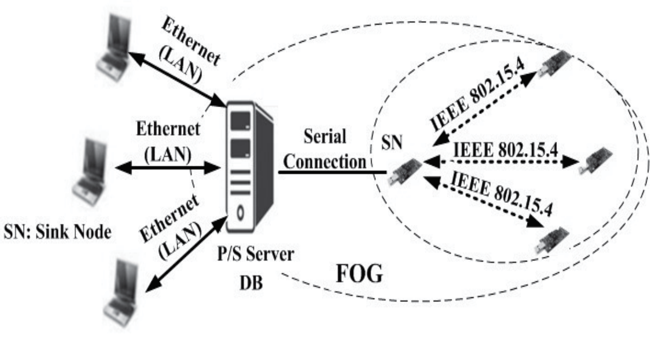

Fig. 3. Testbed setup

A. Traffic Trace Collection for Analysis

For the evaluation of freshness time calculation methods, we collected comprehensive traffic traces from the testbed with changing N, CP, P values in both WiFi and Zigbee mediums. We saved sensor data traces to log files in the server. These traces consist the sensor data arrival timestamp with the sensor node id and the timestamp when periodic timer of E.D. expires with the id of the corresponding client.

TABLE I. Sensor Network Real Testbed Parameters

Parameter Name	Value
Data Sending Periods (P)	2000ms, 5000ms
Number of Sensor Nodes in Real Testbed (N)	1 - 8
Client Defined Emergency Detection Period (CP)	2s, 5s,10s
Sensors	Temperature, Humidity, Light (Type 1)
Physcial Data Acqusition overhead (PA)	250ms
Serial Interface Rate (SR)	115200 bps (∼ 10ms)
Payload Size (PS)	80 Bytes
Network Communication Stack	Rime [22]
MAC for LoWPAN	IEEE 802.15.4
MAC for WiFi	IEEE 802.11a
Communication Topology	Star
Obtained Time Trace Number from each sensor node	600

B. Analysis of Stateless Freshness Time Methods

To analyze the stateless fixed freshness time method, we apply FT starting from 1s to 7.5s with different n percentage into the time trace set. Fig. 4 illustrates the FR results for different N and FT for P=2s. Fig. 5 shows the results when P=5s. Fig. 6 gives the results for WiFi nodes with P=2s and 5s. Different CP values affect the freshness rates with smaller ratios, so, we give results for CP = 2s only. All results for this analysis reveal that, especially for the Zigbee environment, we can obtain a maximal freshness rate when we apply FT greater than $P + PA$. If FT is much smaller than $P + PA$, freshness rates decrease dramatically. Increasing N, while keeping the same FT results in lower freshness rates. Although, WiFi based tests give the same freshness rates even we increase N up to 8, the freshness rates decrease when FT is less than $P + PA$.

The results obtained from this analysis reveal that maximal freshness rate is strongly related to P, PA and the condition of the sensor network connected to the fog. Thus, use of freshness time without considering P, PA and the characteristics of the sensor data acquisition medium could lead to low freshness rates of the emergency decision results.

In order to test the stateless and the standard deviation based method, we collect significant IAT sample and perform a offline calculation to find the mean and standard deviation of IAT values. After that, we apply these values to the E.D.s. The detailed steps of calculations in this method is given in our previous work [20]. The evaluation of this method is given in Section V-D.

C. Stateful Data Freshness Time Analysis

To analyze the stateful method, firstly, we set g value to 0.125 and h value to 0.25 like in the Jacobson's TCP's RTO calculation method. Secondly, we apply different g and h values to observe any change in the freshness rates. For each, we calculate FTs with Equation 4 and find the ratio of pessimistically fresh emergency decision. The evaluation of these two methods is given in Section V-D.

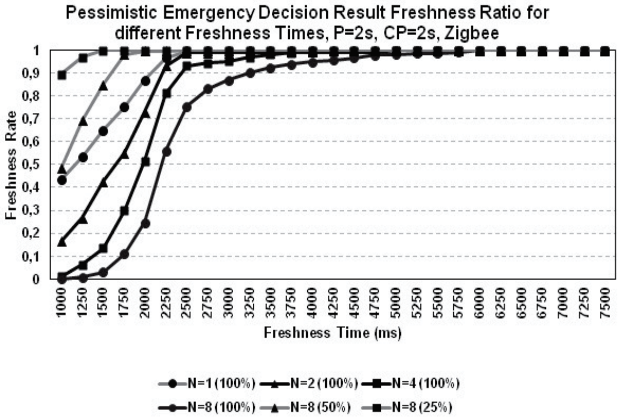

Fig. 4. Freshness Rates of stateless fixed freshness time calculation method for P=2s and Zigbee

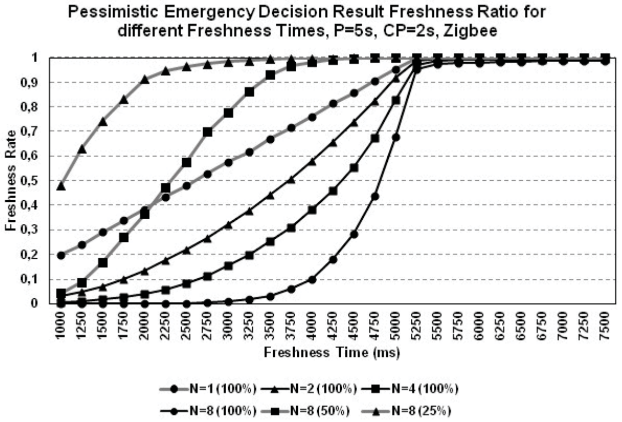

Fig. 5. Freshness Rates of stateless fixed freshness time calculation method for P=5s and Zigbee

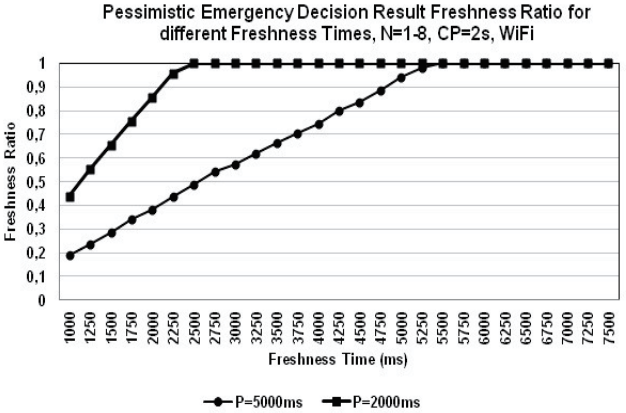

Fig. 6. Freshness Rates of stateless fixed freshness time calculation method for WiFi medium

D. Evaluation of Freshness Rates

Fig. 7 illustrates the decision result freshness ratios with different freshness time calculation methods and sensor network mediums for P=2. Fig. 8 depicts the results for P=5s. The FR results reveals that application of stateful data freshness time calculation methods results better data freshness ratios for the same P and N. As WiFi is more stable than Zigbee, the obtained FR results are better.

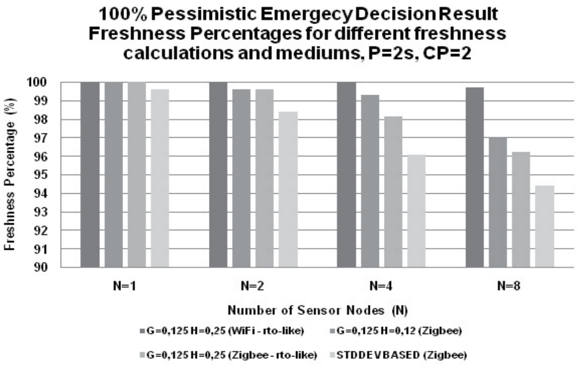

Fig. 7. Freshness Rate Results for several freshness time calculation methods for P=2s

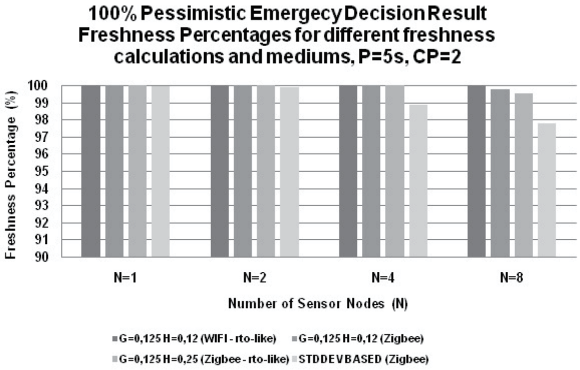

Fig. 8. Freshness Rate Results for several freshness time calculation methods for P=5s

E. Effect of CP in Freshness Rates

We analyze the effect of changing CP to the FR with applying different FT calculation methods. Figures 9 and 10 depicts the results for RTO-like FT calculation in terms of FR. As discussed in Chapter IV, increasing CP leads FR to decrease as expected. However, we do not observe dramatic decrease if we compare results for a low and a high CP values changing from 2s to 10s.

VI. Conclusions

In this paper, we propose a P/S based emergency observation and detection scheme which is based on client sampling period and time stable data integration. We examine several freshness time calculation methods for finding maximal freshness rate in the proposed emergency observation scheme. We devised experiments to evaluate these methods in a real testbed. The outcomes of the analysis can be summarized as follows: we have shown that both MAC protocol of the sensor network and the method used for freshness calculation have

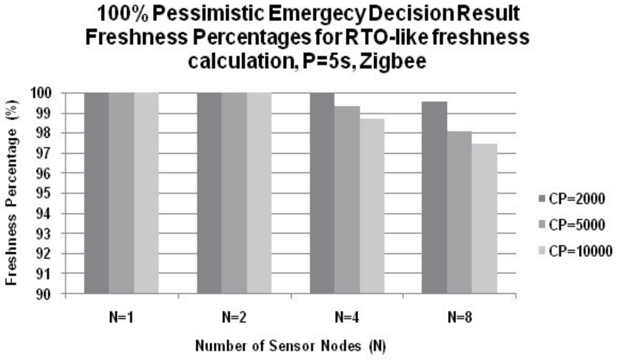

Fig. 9. Freshness Rates of RTO-like freshness time calculation method for P=2s and different client periods

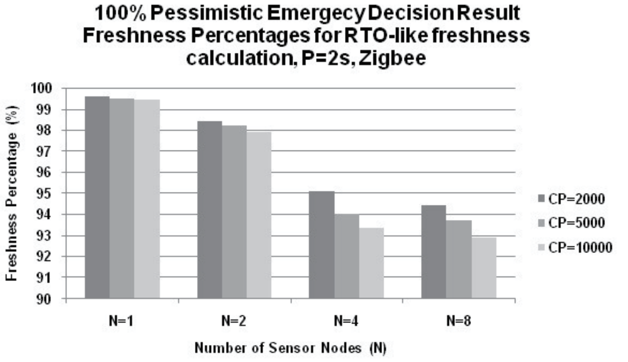

Fig. 10. Freshness Rates for RTO-like freshness time calculation method for P=5s and different client periods

significant affect on the achievable freshness rates. Secondly, low CP values result in higher freshness rates and that stateful methods give better freshness rates than stateless methods. In addition, data freshness rates obtained from analysis can be used to derive a traffic model which reflects the effects of both client SLAs and medium of the sensor network.

For the future work, we will investigate freshness rates with different combination of CP, P and N. We aim to devise experiments with different number of clients for the P/S scheme and evaluate freshness rates. We aim to use data freshness concept in E.Ds for the definition of a dynamic sliding window scheme. We also plan to investigate the impact of freshness rates for the management of emergency traffic generated in inter-fog domain. In addition, we will model the probability of uncertainty in the traffic data with the freshness ratios obtained in this study.

REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “Wireless sensor networks: A survey”, *Computer Networks*, vol.38, 2002, pp. 393-422.

[2] J.W. Hui and D.E. Culler, “Extending IP toLow-Power, Wireless Personal Area Networks”, *IEEE Internet Computing*, vol.12, 2008, pp. 37-45.

[3] K. C. Serdaroglu and S. Baydere, “WiSEGATE: Wireless Sensor Network Gateway framework for internet of things”, *Wireless Networks*, vol.22, 2016, pp. 1475-1493.

[4] Joel J. P. C. Rodrigues and Paulo A. C. S. Neves, “A survey on IP-based wireless sensor network solutions”, *International Journal of Communication Systems*, vol.23, 2010, pp. 963-981.

[5] P. Lindgren, R. Kyusakov, J. Eliasson, H. Makitaavola and P. Pietrzak, “A SOA approach to delay and jitter tolerant distributed real-time Complex Event Processing”, *in Proc. Industrial Electronics (ISIE), 2013 IEEE International Symposium on*, 2013.

[6] Cisco, “Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are”, *Cisco White Paper*, 2015.

[7] W. Pipatsakulroj, V. Visoottiviseth and R. Takano, “muMQ: A lightweight and scalable MQTT broker”, *in Local and Metropolitan Area Networks (LANMAN), 2017 IEEE International Symposium on*, 2017.

[8] M. Bouzeghoub, “A Framework for Analysis of Data Freshness”, *in Proc. 2004 International Workshop on Information Quality in Information Systems*, 2004, pp. 59-67.

[9] G. Jaber, R. Kacimi and T. Gayraud, “Data Freshness Aware Content-Centric Networking in WSNs”, *in Proc. 2017 Wireless Days*, Mar. 2017, pp. 238-240.

[10] Sang-Hun Lee and Hyun-Wook Jin, “A Model for Analyzing Data Freshness of Periodic Real-time Communication”, *in Proc. ACM/IEEE 4th International Conference on Cyber-Physical Systems*, 2013, pp. 260-260.

[11] L. Bright and L. Raschid, “Using Latency-recency Profiles for Data Delivery on the Web”, *in Proc. 28th International Conference on Very Large Data Bases*, 2002, pp. 550-561.

[12] J. Cho and H. Garcia-molina, “Synchronizing a database to Improve Freshness”, *SIGMOD Record (ACM Special Interest Group on Management of Data)*, vol.29, 2000.

[13] A. Labrinidis and N. Roussopoulos, “Using Latency-recency Profiles for Data Delivery on the Web”, *in Proc. 29th International Conference on Very Large Data Bases*, 2003, pp. 393-404.

[14] P. Veronika, R. Ruggia and M. Bouzeghoub, “Analyzing and Evaluating Data Freshness in Data Integration Systems”, *Ingnierie des Systmes d’Information*, vol.9, 2004, pp. 145-162.

[15] A. Ghobakhlou, A. Kmoch and P. Sallis, “Integration of Wireless Sensor Network and Web Services”, *in Proc. 20th International Congress on Modelling and Simulation*, 2013.

[16] U. Hunkeler, H. L. Truong, A. Stanford-Clark, A. Kmoch and P. Sallis, “MQTT-S A publish/subscribe protocol for Wireless Sensor Networks”, *in Proc. Communication Systems Software and Middleware and Workshops, 3rd International Conference on*, 2008.

[17] MQTT Version 3.1.1, Oasis Standard, Web: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.pdf.

[18] B. Tudosoiu, “Feasibility Study: Minimum Viable Device to support Internet of Things Realization, Mobile Heights”, *in Proc. Engineering & MIS (ICEMIS), International Conference on*, 2014.

[19] K. C. Serdaroglu, T. Kadioglu and S. Baydere, “Publish-Subscribe Based Monitoring Model For Wireless Sensor Networks”, *New Advances In Internet of Things*, 2018, pp. 45-58.

[20] K. C. Serdaroglu and S. Baydere, “Data freshness model for P/S based sensor monitoring system”, *in Proc. 2018 26th Signal Processing and Communications Applications Conference (SIU)*, 2018, pp. 1-4.

[21] V. Jacobson, “Congestion Avoidance and Control”, *SIGCOMM Comput. Commun. Rev.*, vol.18, August 1988, pp. 314-329.

[22] A. Dunkels, F. sterlind and Z. He, “An Adaptive Communication Architecture for Wireless Sensor Networks”, *in Proc. SenSys’07*, 2007.