Modelization of Temporal Mechanisms for Sensors Networks

Pascal Lorenz[1], Marc Gilg[1], and Joel J.P.C. Rodrigues[2]

[1] University of Haute Alsace
34 rue du Grillenbreit 68008 Colmar, France
`{pascal.lorenz,marc.gilg}@uha.fr`
[2] Instituto de Telecomunicações, University of Beira Interior
Rua Marquês D'Ávila e Bolama - 6201-001 Covilhã, Portugal
`joeljr@ieee.org`

Abstract. There are a lot of real-time communication requirements, for example for networks in a factory, in a car, in an airplane. Communication networks, especially sensors networks, need more and more to respect real-time constraints and the non-respect of the time constraints can cause the dead of peoples and/or the destruction of materials. A real-time network realizes a set of functions in which the start and/or end times of execution must respect the temporal constraints imposed by the environment. This paper presents technical specifications to define temporal constraints for communication networks. Temporal statuses are generated to test the temporal validity of the exchanged data. Simulations are done to prove the validity of our model for sensors' communications.

Keywords: temporal systems, sensors networks, real-time systems, modeling.

1 Introduction

For some applications using sensors networks, it is necessary to respect real-time constraints [4], [7], [15]. For example, in a car using sensors networks, the communication between the sensors connected to the brake and the actuators connected to the wheel should respect real-time constraints (such as millisecond or nanosecond). The non-respect of real-time constraints introduces danger for peoples and for materials. Today very few networks offer real-time or time-critical mechanisms [1].

Therefore new temporal mechanisms should be used to enable to respect real-time constraints in networks sensors [5], [10], [14].

The rest of this paper is organized as follows. Section 2 describes the different temporal mechanisms with the introduction of Temporal Window (TW) and Variable validity time Window (VW). It develops a modeling approach of the temporal mechanisms. Simulation of the temporal mechanisms applied to sensor networks is conducted in section 3. Section 4 concludes this paper and points further research directions.

J. Rodriguez, R. Tafazolli, C. Verikoukis (Eds.): MOBIMEDIA 2010, LNICST 77, pp. 111–122, 2012.
© Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2012

2 Temporal Mechanisms

Temporal mechanisms are very important to provide quality-of-service (QoS) mechanisms for real-time applications [6], [8]. Temporal intervals used for a communication can be divided into several sub-temporal intervals to enable the localization of an error and to know if the global temporal constraints have not been respected.

The three major steps are the following:

- *i*) production of the information by the sensor,
- *ii*) propagation of the information through the medium,
- *iii*) consumption of the information by the actuator [2], [13].

At each step of the communication, temporal statuses should enable to know if the temporal constraints have been satisfied or not.

To obtain useful knowledge about the temporal constraints, the communication between entities can be decomposed into several sequential steps:

- production of information,
- progress through the different layers of the OSI model at the producer entity,
- sending and propagation on the medium,
- progress through the different layers of the OSI model at the consumer entity,
- consumption of the information.

The production and sending temporal statuses (denoted respectively PS and SS), which are related to the temporal constraints at the production entity, are transmitted with the data to the consumer entities. Likewise, the receiving and consumption temporal statuses (denoted respectively by RS and CS), which have to respect the temporal constraints at the consumption entity, are added to the received data and to the production entity status. The elaboration of the different statuses can be represented as shown in Figure 1.

In this work, we will suppose that local clocks of production and consumption entities are synchronized. If this hypothesis is not respected, it is not possible to manipulate temporal constraints because the fact that a data production date is superior to the consumption date for the same data is not possible [3], [9], [11], [12].

We will use the notion of Temporal Window (**TW**), in which information should be managed at a given time, not too early (not before the beginning of the temporal window) and not too late (not after the end of the temporal window), but just in time.

The three activity temporal windows are the following:

- production of the temporal window (noted to prod);
- transmission of the temporal window (noted to trans) composed by the sending and the receiving tasks,
- consumption of the temporal window (noted to cons).

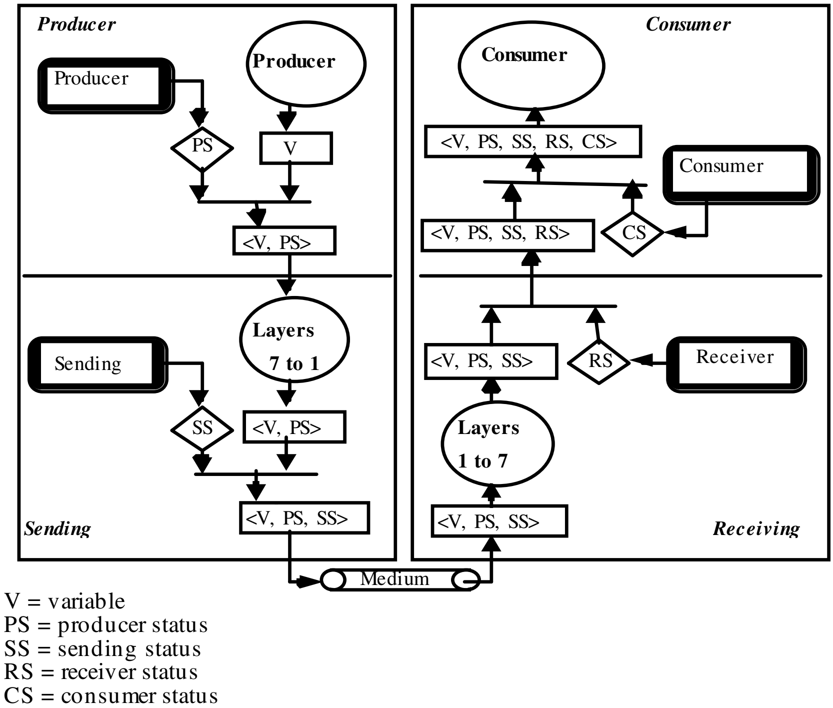


Fig. 1. Mechanisms for the elaboration of temporal status

i and *v* parameters are used to represent the i^{th} temporal window used for a given variable *v*. Then, a time window can be described as follows:

State Function TW (v, i) with
State = (**S**tart, **E**nd or **D**uration) and
Function = (production, transmission or consumption)

Some rules can be elaborated, for example the fact that the start of a TW appears always before the end of the same TW. This rule can be described as follows:

Start Function (v, i) < End Function (v, i)

At a given time, a produced data has duration of life which can changes according to consumers entities. For statistical applications, data validity can be unlimited but not for time critical applications. So data validity depends on the utilization of this later. For example, the validity of a temperature data can be considered valid (as long as a new updated data has not been produced) even after the end of the temporal window.

To formalize the concept of data validity (duration of life), the term of Variable validity time Window (noted **VW**) will be used. A temporal window for data validity is described as follows:

State VW (*v*, *i*) with
State = (**S**tart, **E**nd or **D**uration)

3 Modeling of Temporal Mechanisms

At a given time, it is important to know if the current state is located inside or outside a TW and a VW; and to know if VW (v, i-1) exists or not.

The different possible situations for a TW and a VW are:

- located out of a TW, out of a VW and it does not exist a previous VW for a given variable v,
- located in a TW, out of a VW and it does not exist a previous VW for a variable v,
- located in a TW and in a VW; this state is called *True* state,
- located in a TW, out of a VW and it exists a previous VW for a variable v,
- located out of a TW and in a VW
- located out of a TW, out of a VW and it exists a previous VW for a variable v.

The six different possible situations described below are represented in the figure 2.

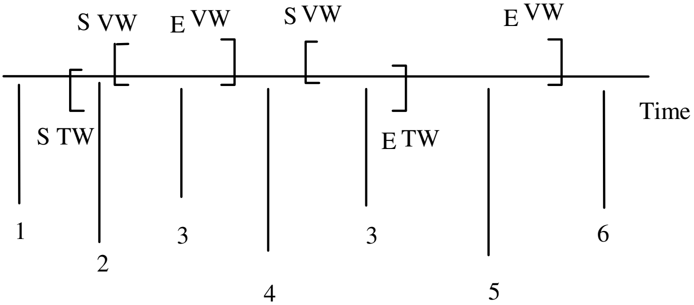


Fig. 2. Relationships between Temporal Window and Variable validity time Window

Table 1 describes the different possibilities with the previous proposed notation:

Table 1. Description of the different states

In TW (v, i)	In VW (v, i)	∃ a VW (v, i-1)	Name of the state
0	0	0	False
0	0	1	Expiration2
0	1	-	TW-Expiration1
1	0	0	Wait
1	0	1	VW-Expiration1
1	1	-	True

The symbols used in table 1 are:

0 = the condition is not verified
1 = the condition is verified.

Then the six possible combinations between the TW and VW are:

- 1 = *False state*: located out of a [Start Function (v, i), End Function (v, i)], out of a [Start VW (v, i), End VW (v, i)] and it does not exist a Start VW (v, i-1)

- 2 = *Wait state*: inside the a [Start Function (v, i), End Function (v, i)], out of the [Start VW (v, i), End VW (v, i)] and it does not exist a Start VW (v, i-1)

- 3 = *True state*: inside the [Start Function (v, i), End Function (v, i)], inside the [Start VW (v, i), End VW (v, i)], then at least one variable v has been produced respecting the timing constraints

- 4 = VW-Expiration1 state: inside a [Start Function (v, i), End Function (v, i)], out of a [Start VW (v, i), End VW (v, i)] and it exists a Start VW (v, i-1)

- 5 = TW-Expiration1 state: out of a [Start Function (v, i), End Function (v, i)], but inside [Start VW (v, i), End VW (v, i)]

- 6 = TW+VW-Expiration / Expiration2 state: out of the [Start Function (v, i), End Function (v, i)], out of a [Start VW (v, i), End VW (v, i)] and it exists a Start VW (v, i-1).

To simulate the temporal communication between sensors, we will add to each step of sensors' communications a time-out and a temporal status.

Therefore, a sensor can have three different statuses:

- data production in time (true or false),
- data transmission in time (true or false),
- data consumption in time (true or false).

For each communication step, we will have a start event (generating an event to start the TW and the VW) and an end event (generating a time-out to close the TW and the VW).

The protocol integrating the 6 different states of the TW and VW is described in the following figure 3.

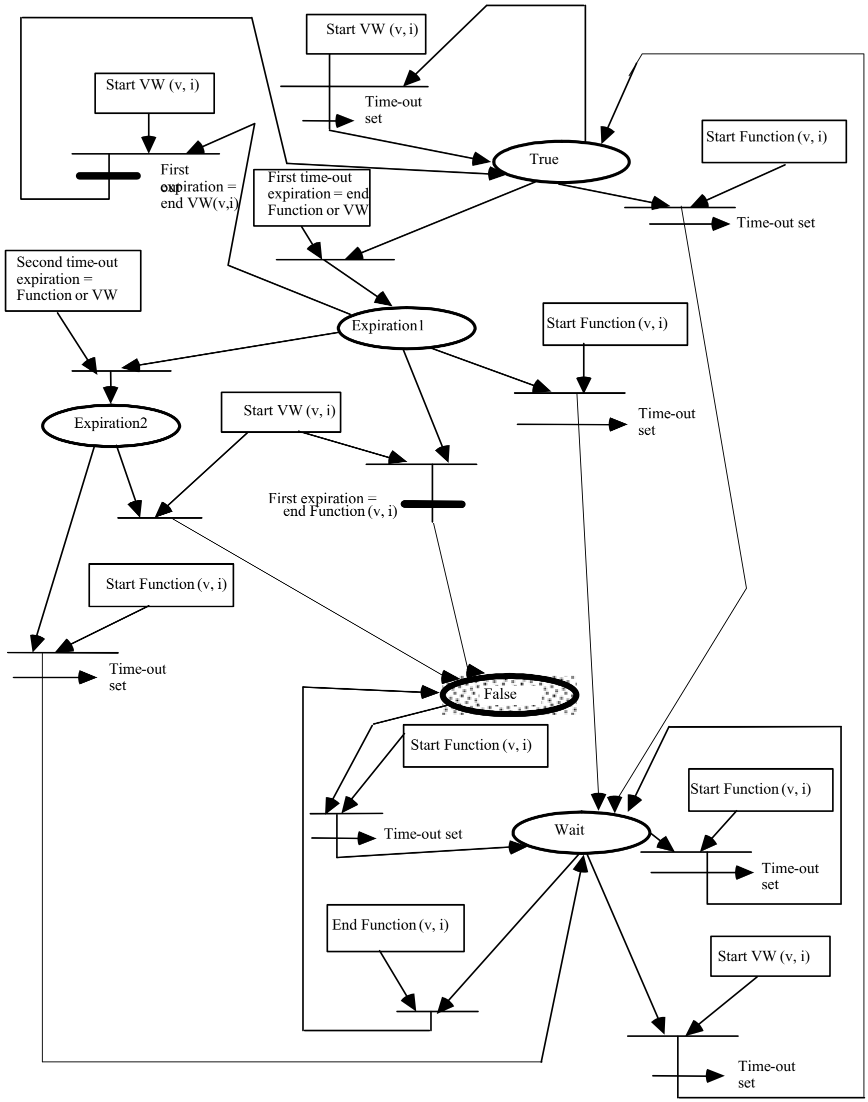
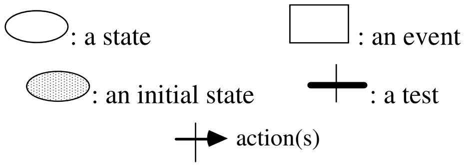


Fig. 3. Description of the protocol integrating Temporal Window and Variable validity time Window

The symbols used in figure 3 are:



When the application start, the initial state is the *False* state.

4 Simulations Applied to Sensors Networks

The simulations are done with OMNET++ and MiXiM package. Three and ten mobile nodes are broadcasting packets with a burst size of 300 packets. The nodes are moving constantly with a speed of 1mps in a random position. The update interval is 0.1s. For each packet we compute the delay between the packet creation and the packet reception. The packet is valid if the reception is done before a given deadline who defining the VW.

The TW is defined by two parameters:

- Duration of the Time Window (DTW) and
- Start Time Window (STW).

A packet is valid if it is received at the given time Tr such as:

$i * STW + (i-1) * DWT \leq Tr$

and

$Tr \leq i * STW + i * DTW$ for a given integer *i*.

The simulation is done with STW = 1 ms, DTW = 2 ms and DVW = 40 ms. The behavior of packet received by the nodes will be detailed in the next figures.

Figure 4 shows the number of VW packets received in a simulation time slot. Each slot in the graph corresponds to a received packet. The different lines in the graph correspond to different nodes. We can notice that this number is increasing until DWV = 45 ms because after this time the packets are buffered and are noted as received in a VW.

Figure 5 shows the number of packets received in a valid TW. We can notice that most of the packets are received in a valid TW. When the line is horizontal, this means that the packet is received outside of the TW.

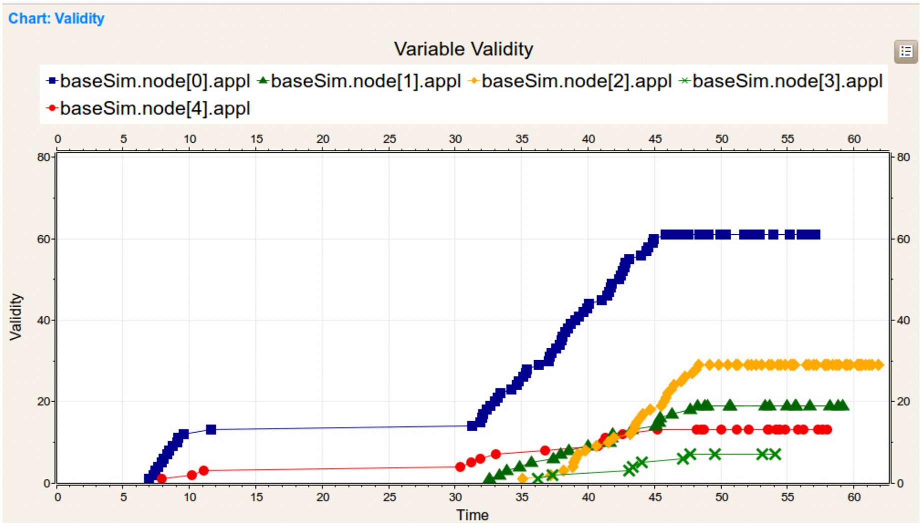


Fig. 4. Packets measured for Variable validity time Window

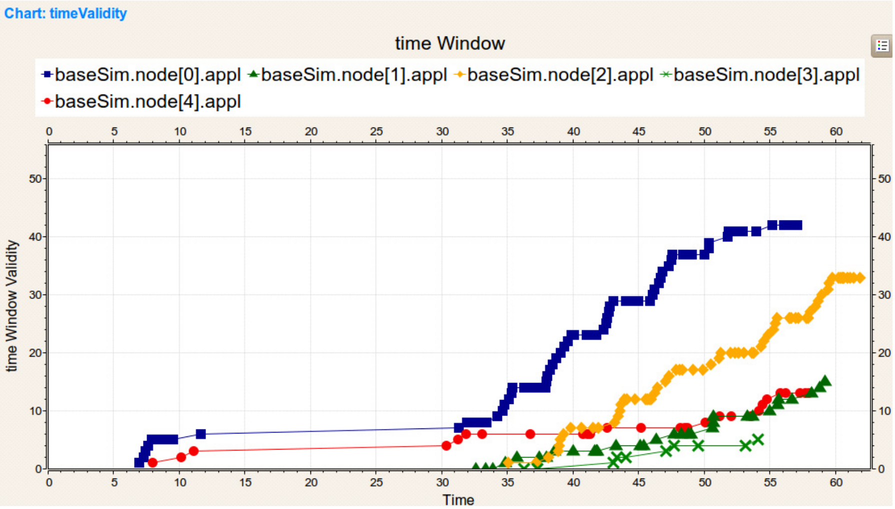


Fig. 5. Packet measured for Time Window

Figure 6 shows when packets are in a *true* state. We can notice that the horizontal line in the graph correspond to the *false* state, *VW-Expiration* or *TW-Expiration* states. This case is similar to the VW graph. After 45 ms we notice that the packet is no more in a VW, but in a *VW-Expiration* state or in a *False* state.

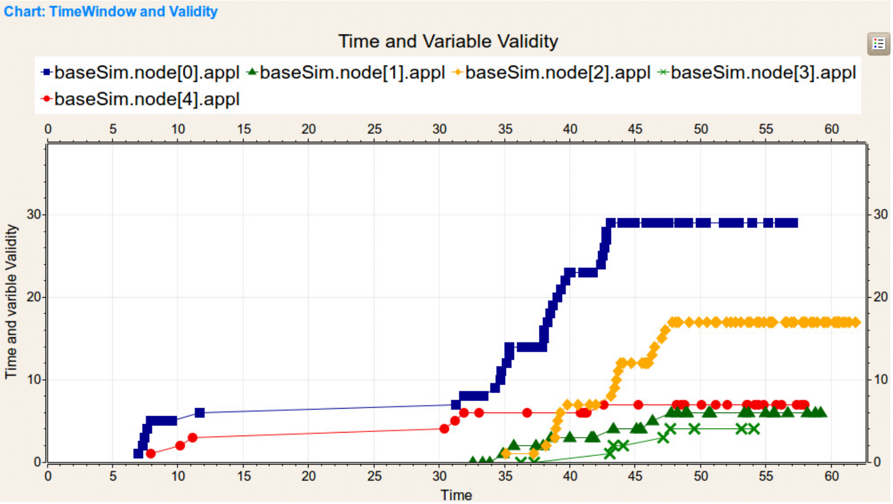


Fig. 6. TW and VW combination

The simulation results prove that the proposed temporal mechanisms enable to have temporal information that can be used to locate the processes who do not respect the temporal constraints.

The next simulations are done with 10 nodes. The nodes are moving in a random direction with a continuous speed of 100mps. The burst size is of 300 000 packets. Each node broadcast packets to his neighbors. A packet is valid if it received the packets in less than 40ms. The figure 7 shows the number of valid packets received in function of the time. The stagnation of the curve means that there is no valid packet received during that time.

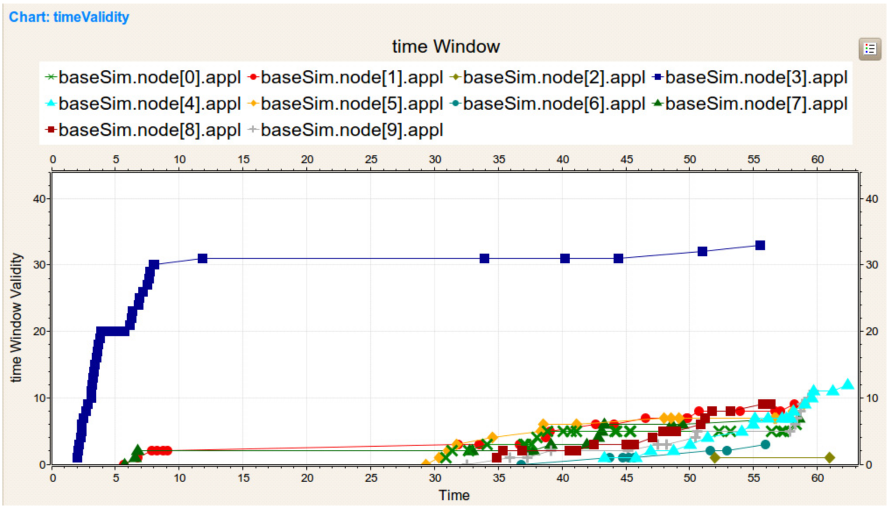


Fig. 7. Time Window for 10 nodes

We notice that node 3 receives packets during a time slot of 0 to 10 ms, after that we have a period of non-valid packet received until 45 ms. Some valid packets are received and the nodes are moving randomly. Node 3 is the broadcasting destination of packets send by other nodes. After 10 ms, the network is saturated and the packets are buffered. After 30 ms, the node 3 is able to reply to the other nodes because the network become available. This is confirmed by the numbers of valid packets received by the other nodes after this time. Therefore we can deduce that node 3 has a central position and he receives a lot of traffics from other nodes. After 10 ms, every neighbor of nodes 3 has finished his broadcast and node 3 is now able to reply by a broadcast to other nodes after 30 ms, when the neighbors increase the number of valid packets received.

Figure 8 details the Variable validity time Window for 10 nodes:

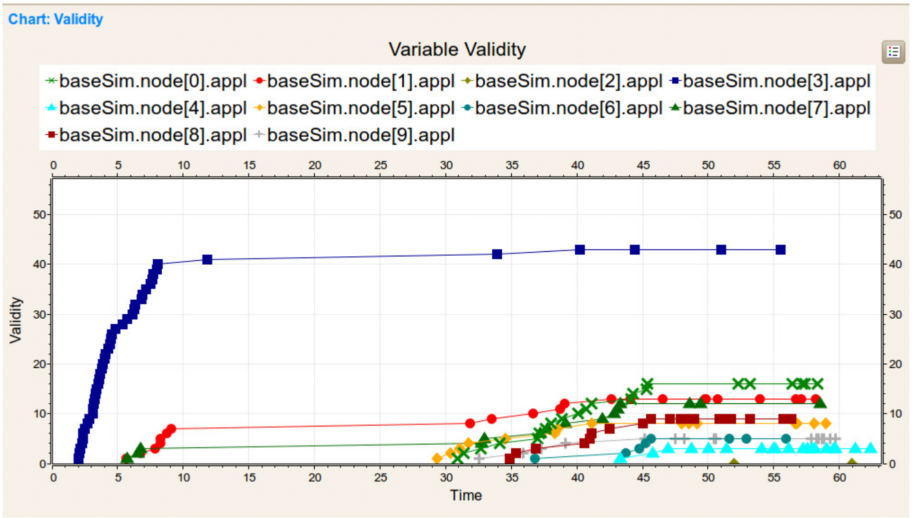


Fig. 8. Variable validity time Window for 10 nodes

Figure 8 is quite similar to the figure 7 because VW packets are received in the TW. But we can see that the curves are plotted with superior values compared to the figure 7, this is due to the fact that a variable can be valid even if it is received after the end of the time window. After 30 ms, the nodes receive a lot for VW packets.

Figure 9 shows how packets are received in Time Window and Variable validity time Window:

This figure is quite similar to figure 7 and 8 because it is a combination of the two previous ones. We notice that for the different nodes, the curve is lower than figures 7 and 8 because some packets have been received in a valid time but they have now a non-valid value.

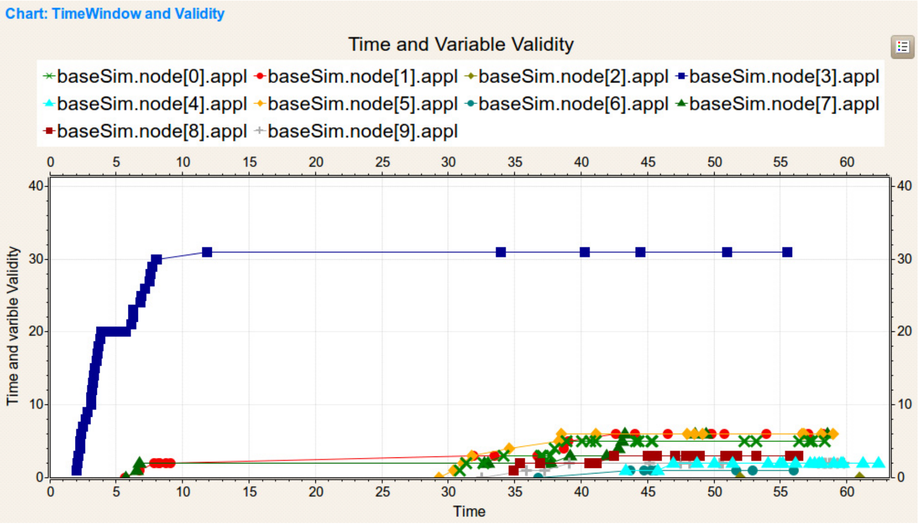


Fig. 9. TW and VW combination for 10 nodes

The validity of packets is difficult to be respected; scheduling and fairness mechanisms should be study.

5 Conclusions and Future Works

This paper has presented temporal description of time windows and variable validity time window to know the data's temporal statuses. These mechanisms enable to know, if the data temporal validity is correct, if this later can be used or not and where is located the temporal error (in the production, transmission or consumption steps). Simulations have been applied to sensors networks and confirm the validity of our model.

Theses temporal mechanisms can be used for different type of networks and real tests should be done in the future to validate these later in large sensors networks.

Acknowledgements. Part of this work has been supported by Instituto de Telecomunicações, Next Generation Networks and Applications Group (NetGNA), Portugal, and by Luso-French Program of Integrated University Actions (PAUILF 2010) – Action No. F-TC-11/10.

References

1. Fontan, B., Mota, S., de Saqui-Sannes, P., Villemur, T.: Temporal Verification in Secure Group Communication System Design. In: The International Conference on Emerging Security Information, Systems, and Technologies, pp. 175–180 (October 2007)

2. Barz, C., Pilz, M., Wichmann, A.: Temporal Routing Metrics for Networks with Advance Reservations. In: Eighth IEEE International Symposium on Cluster Computing and the Grid, pp. 710–715 (May 2008)
3. Bertino, E., Ferrari, E.: Temporal Synchronization Models for Multimedia Data. IEEE Transactions on Knowledge and Data Engineering, 612–631 (July 1998)
4. Liu, H., Gupta, R.: Temporal Analysis of Routing Activity for Anomaly Detection in Ad hoc Networks. In: IEEE International Conference on Mobile Ad Hoc and Sensor Systems, pp. 505–508 (October 2006)
5. Li, H., Liu, S., Hu, B.: Research on Node Sleep/Wake-Up Mechanism in WSN Based on Fuzzy Energy Control. In: Second International Conference on Intelligent Networks and Intelligent Systems, pp. 11–14 (November 2009)
6. Tyan, H.-Y., Hou, C.-J., Wang, B.: A Framework for Provisioning of Temporal QoS in Core-Based Multicast Routing. In: 20th IEEE Real-Time Systems Symposium, RTSS 1999 (December 1999)
7. Tinnirello, I., Choi, S.: Temporal Fairness Provisioning in Multi-Rate Contention-Based 802.11e WLANs. In: Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM 2005), pp. 220–230 (June 2005)
8. Basu, K., Ball, F.: A Scheduling Architecture for QOS and Temporal Mapping. In: First Asia International Conference on Modelling & Simulation (AMS 2007), pp. 258–263 (March 2007)
9. Cardenas, M.A., Navarrete, I., Marin, R.: Efficient Resolution Mechanism for Fuzzy Temporal Constraint Logic. In: Seventh International Workshop on Temporal Representation and Reasoning, TIME 2000 (July 2000)
10. Kamat, P., Xu, W., Trappe, W., Zhang, Y.: Temporal Privacy in Wireless Sensor Networks. In: 27th International Conference on Distributed Computing Systems, ICDCS 2007 (June 2007)
11. Obermaisser, R.: Temporal Partitioning of Communication Resources in an Integrated Architecture. IEEE Transactions on Dependable and Secure Computing, 99–114 (April 2008)
12. Fu, S., Xu, C.-Z.: Quantifying Temporal and Spatial Correlation of Failure Events for Proactive Management. In: 26th IEEE International Symposium on Reliable Distributed Systems, pp. 175–184 (October 2007)
13. Cucinotta, T., Anastasi, G., Abeni, L.: Respecting Temporal Constraints in Virtualised Services. In: 33rd Annual IEEE International Computer Software and Applications Conference, pp. 73–78 (July 2009)
14. Tseng, V.S., Lin, K.W., Hsieh, M.-H.: Energy Efficient Object Tracking in Sensor Networks by Mining Temporal Moving Patterns. In: IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, pp. 170–176 (June 2008)
15. Wang, Y.-C., Hsieh, Y.-Y., Tseng, Y.-C.: Multiresolution Spatial and Temporal Coding in a Wireless Sensor Network for Long-Term Monitoring Applications. IEEE Transactions on Computers, 827–838 (June 2009)