Narrowband Delay Tolerant Protocols for WSN Applications: Characterization and Selection Guide

C. S. Malavenda[1], F. Menichelli[2], M. Olivieri[2]

Selex E.S. Rome, Italy; e-mail: claudiosanto.malavenda@selex-es.com.
[2] Sapienza University of Rome, Rome, Italy; email:
{menichelli,olivieri}@diet.uniroma1.it.

Abstract. This article focuses on delay tolerant protocols for Wireless Sensor Network (WSN) applications, considering both established and new protocols. We obtained a comparison of their characteristics by implementing all of them on an original platform for network simulation, and by testing their behavior on a common test-bench. Thereafter, matching the requirements linked to each application with the performances achieved in the test-bench, allowed us to define an application oriented protocol selection guide.

1. Introduction

The subject of this work is the presentation of a comparative analysis between communication protocols for Wireless Sensor Networks (WSNs), allowed by the development of an original simulation framework, and its link with the requirements of typical WSN applications in order to produce a comparative application-oriented selection guide for WSN protocols.
The focus of our work is on delay tolerant network (DTN) protocols. This category of protocols is particularly suited for mobile WSNs where the topology is dynamically constructed and there is the possibility to momentarily lose the connection between nodes.
The paper is organized as follows: Section 2 selects a set of protocols that will be the subject of our investigation and introduces the metrics that will be considered for the evaluation of the protocols. Section 3 analyzes and compares the protocols behavior when the number of nodes is increased and a new evaluation metric is introduced to measure the power consumption of the protocol. Section 4 matches the results with the requirements of the considered applications, offering a new application-oriented selection guide for delay tolerant protocols that suit WSN communications.

2. DTN Protocols Set

In this section we describe a comprehensive set of protocols relevant in DTN applications [2]. We consider both established protocols and novel ones, the latter designed as variations of existing ones [3] [4][5][6].
The protocols are the following:

- Controlled Flooding (C.F.) [10]: a custom version of the controlled flooding protocol. The message is logically forwarded just once from each relay node. The relay node

assumes that the packet is relayed when a maximum number of retransmission is reached or when an ack packet is received before the maximum number of retries is reached.

- Epidemic [11]: a variation of the conventional controlled flooding protocol. When a node is in contact with a new one it starts a "anti-entropy" phase where two nodes exchange a message vector containing information about messages that each node stores. This phase is followed by the message exchange. Each node implements heuristics to determine if it wants to receive an unseen message from its neighbor. This protocol is particularly resource-hungry.
- Spray and Wait [12] (SnW): a node generating a new message has to deliver it to certain number ('*L*') of different neighbors. This is called "Spray Phase". When a relaying node receives the message, it starts the "Wait phase", where each relay node will transmit the packet only if the destination node is found (direct delivery).
- Prophet [13]: this algorithm associates a delivery probability to every node. When a node meets another, their stored "encountering" probabilities are updated and merged with each other data. Packet transmission occurs only if the encountered node has a better probability to meet the destination node than the source one.
- Controlled Flooding with Hop Limitations (Lim.): an improved version of the above-mentioned "Controlled Flooding" algorithm. In addition to its base version, this protocol limits the number of packet copies to fixed number of replies. This protocol uses a sort of Time to Live parameter embedded into each packet, each hop decreases that parameter, when it reaches zero, no receiver node can reply to the packet.
- Controlled Flooding SnW: a custom version of controlled flooding protocol where copies are limited on the network with a SnW-like strategy. (Referred below as C.F. SnW or C.F. Lim. SnW).

All protocols have been tested in the simulator "ONE" [7] [8], a protocol level simulator chosen for the speedup it offers over architectural level simulators [9]. Established protocols, such as "Epidemic", "SnW" and "Prophet" have been simulated using a set of optimized parameters embedded in the simulator itself. The new protocols introduced in this paper have been optimized with an exploration of their parameters that is presented in the following.

Results are presented using a set of metrics suitable for both the asynchronous "Medium Access Control" (MAC) layer [1] and the delay tolerant "Logical Link Control" (LLC) level.

The adopted evaluation metrics are the following:

-*Delivery Probability*: the ratio between the number of successfully delivered data packets and the number of packets generated by source nodes.

-*Latency*: time delay of the message transmission from the source node to its arrival to the destination node. The *average* measurement is the arithmetic mean value of measured latencies. The *mean* measurement is the value positioned in middle of all ordered vectors of measured latencies.

-*Average Hop Count*: The average number of nodes that a packet need to traverse in order to be delivered.

-*Overhead*: number of redundant packet copies that are disseminated in the network and extra control-packets exchanged for protocol specific purposes calculated as

$$\frac{(msgR - msgD)}{msgD}. \qquad (1)$$

Where *msgR* is the number of messages relayed by the node and *msgD* is the number of messages delivered.
In section 3 we introduce an additional indirect measurement of protocol complexity, linked to simulation time.
The simulation scenario assumes that a random sender node generates a packet addressed to a random destination-node. All nodes that sniff such packet can act as relay nodes, i.e. capture the packet and retransmit it or just drop it. The strategies that lead to such choice define the Delay Tolerant (DT) protocol itself.
In the simulation scenario, mobile nodes interact with each other with variable length messages composed by few bytes. The simulation field and node dynamicity involve fast topology changes. Table 1 shows the parameters adopted in the simulations.

Scenario Parameter	Value
Simulation Period	6h
Payload	30Bytes to 60Bytes
Transmission velocity	100kbps
Transmission Range	300 m
Mobility Model	Random Waypoint
Mobility Pause	from 0 to 120 sec
Mobility velocity	from 0 to 2 m/s
Message Buffer Dimension	2MB

Table 1: Scenario Common Parameters

2.1 Controlled Flooding - Protocol Calibration

For calibration of Controlled Flooding protocol, we used a scenario with 1Km X 500m size, populated with four mobile nodes. Two protocol parameters are explored:

- the maximum number of transmission retries before declaring a transmission failed (*Tnum*)
- the minimum wait time for™ packet re-transmission (*Trep*).

Protocol performances are analyzed for two different cases, flat traffic (nodes are characterized by small data size periodic transmission) and burst traffic (nodes are characterized by a burst of high data rate transmission).

1) Flat Traffic Test

In this scenario each node randomly generates a message every 300 to 330 seconds. The source and destination of each message are chosen with a pseudo-random scheme. With an exploration analysis of the main protocol parameters, performed running one simulation for each parameters pair, we achieves the results in Figure 1 and Figure 2. Latency values presented in Figure 1 seem to be not affected by *Tnum*, but only by *Trep*. We see that we achieved latency values below 1 second for *Trep* in the [0.1; 1] discrete interval. Figure 2 shows the dependency of delivery ratio to *Tnum* and *Trep*. We see that for the [0.1; 1] range of *Trep* values, delivery probability has a quite flat response for *Tnum* values greater than 3.

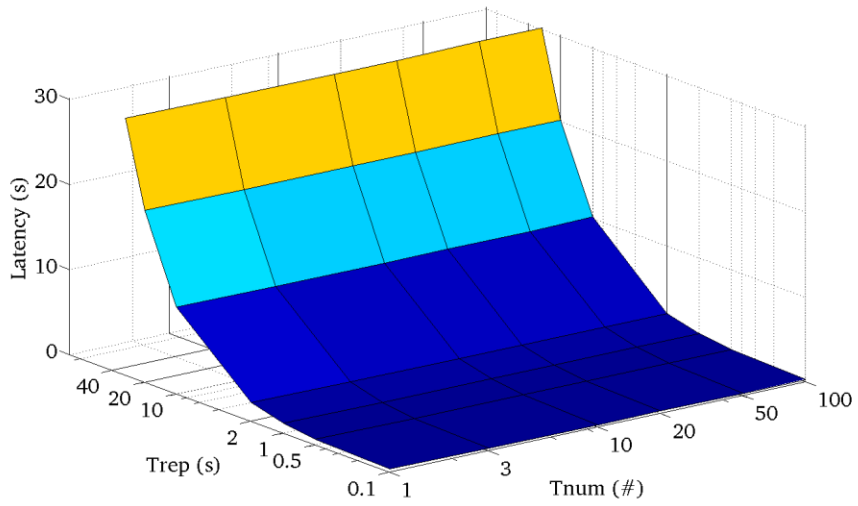

Figure 1: Pre-Operational Test – Latency

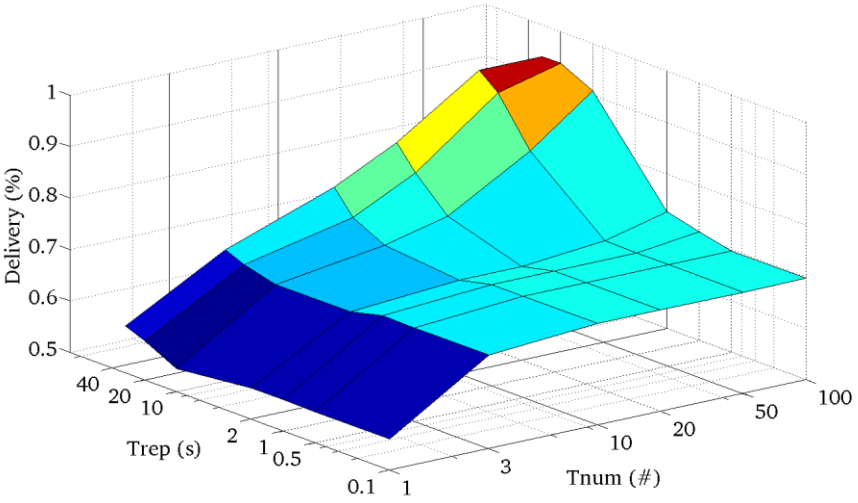

Figure 2: Pre-Operational Test - Delivery Probability Transmission Peak Test

This test evaluates protocol metrics when a burst of transmissions is initiated during the first minute of simulation, with a new message every second (the minimum value allowed by the simulator). Figure 3 traces metrics evolution with *Trep*, considering a *Tnum* value of 3.

We see that all metrics report performance degradation for increasing *Trep* values, so we selected *Trep* = 0.1 s for this protocol.

In order to set the value of *Tnum* for the two scenarios considered, we report the simulation results in Figure 4, obtained for increasing *Tnum* values. Considering the huge increase of the latency and also the increase of energy needed for re-transmissions, we decide to set *Tnum* to 3, even if it is characterized by lower delivery probability.

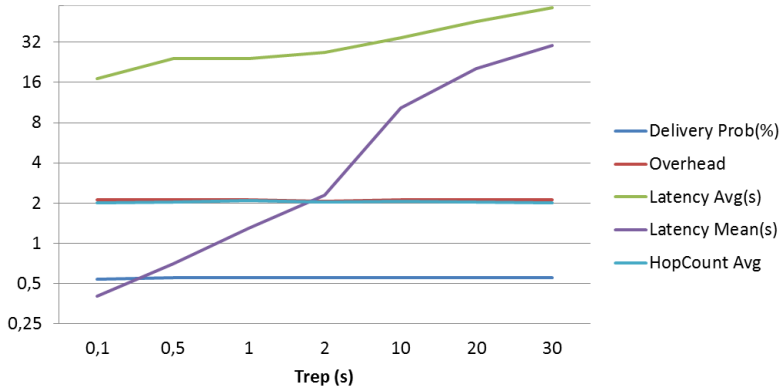

Figure 3: Evaluation metrics vs Trep

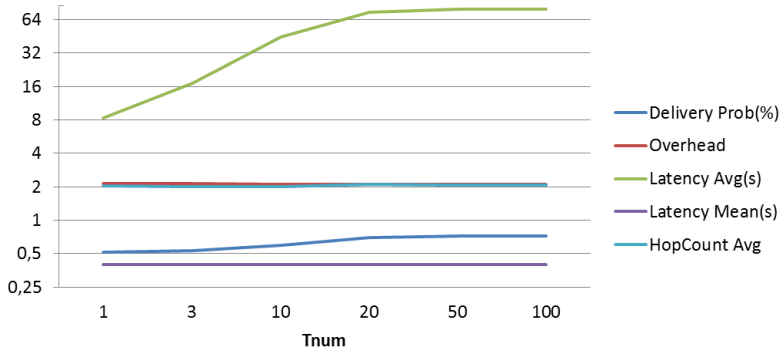

Figure 4: Evaluation metrics vs Tnum

2.2 C.F. with Hop Limitation - Protocol Calibration

The scenario parameters are presented in Table 1. The traffic simulated is the same as in 2.1).

This protocol also introduces a new parameter, referenced as *Max Hop* number. It is the maximum number of Hops that a packet can perform before being discarded. In Figure 5 it is possible to see how our metrics are dependent to this parameter, while *Tnum* and *Trep* have been assigned the values *Tnum* = 3 and *Trep* = 0.1.

We can see that, starting from *Max Hop* = 5, all metrics have stable values. In order to verify that the protocol have the same behavior regarding the variation of *Trep* and *Tnum* parameters, we report the metric measurement obtained varying them while keeping *Max Hop* = 5. In Figure 6 we report *Trep* dependency considering *Tnum* = 3. It is possible to see that all metrics, except Latency, have constant values starting from Trep=0.1s.

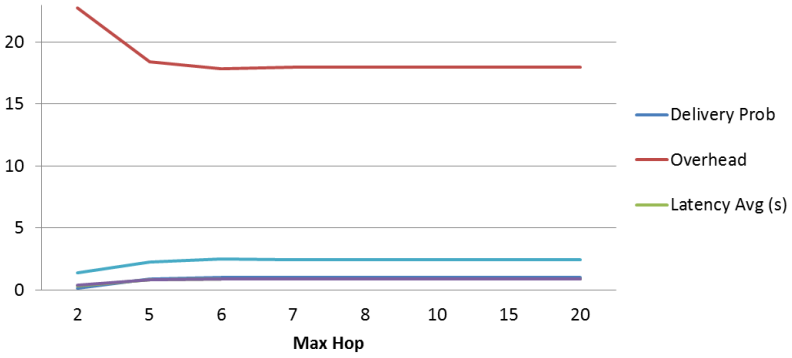

Figure 5: Evaluation metrics vs Max Hop

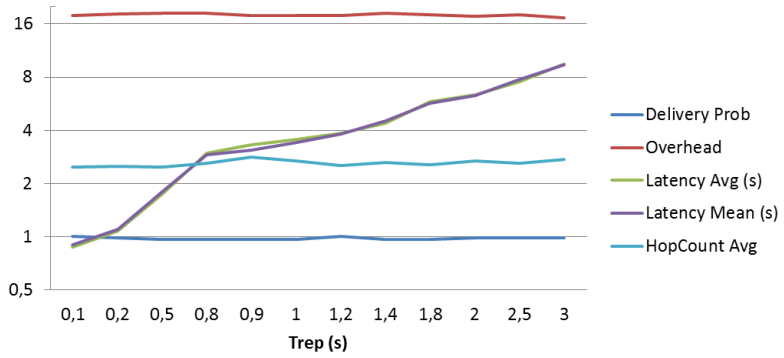

Figure 6: Evaluation metrics vs Trep

Figure 7 reports simulation results setting *Tnum* = 0.1 and *Max Hop* = 5. Metrics are all stable from *Tnum* = 3.

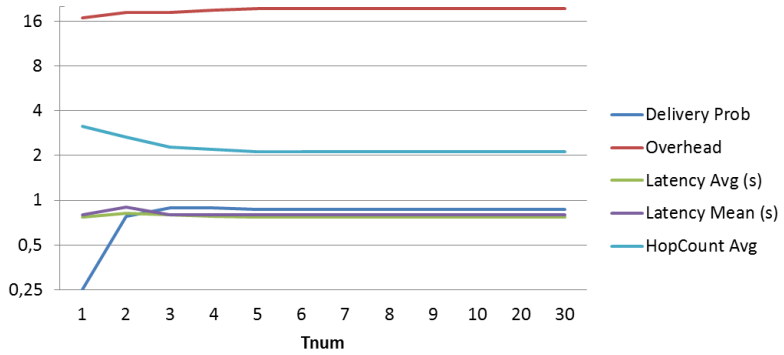

Figure 7: Evaluation metrics vs Tnum

2.3 C.F. SnW - Protocol Calibration

Here we present the results of the optimization process applied to the controlled flooding protocol with copy limiting algorithm imported by the spray-and-wait algorithm. The objective is to quantify the benefits of the SnW protocol in terms of delivery rate and low overhead [14].

The scenario parameters are summarized in Table 1 and the traffic is the same of the test described in Section 2.1).
The first set of simulations aim at calibrating protocol parameters *Trep* and *Tnum* and are presented in Figure 8. Latency has a quite stable value between 300 ms and 400 ms. The Hop count is stable around 1.7 . Delivery probability and Overhead assume constant values of 1 (i.e. 100% delivery probability rate) and 18, respectively.
Since varying *Tnum* did not affect any of metrics, we do not show the corresponding results.
According to the above results, we chose *Tnum* = 1 (as low as possible to minimize transmissions **Errore. L'origine riferimento non è stata trovata.**) and *Trep* = 0.2. These values have been used in the simulations to explore the behavior of the "number of copies" parameter (*CPY*), that introduces a limitation of the copies in the network. Increasing *CPY*, we see that the average latency and other metrics have no significant improvements, so we set this value to 3.

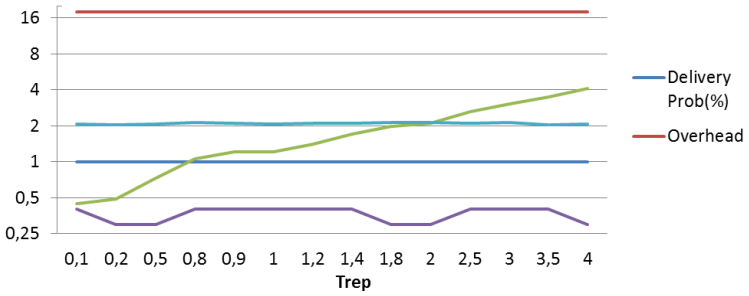

Figure 8: Evaluation metrics vs Trep

3. Protocol Performance Comparison vs Number of Nodes

In this section, we explore protocols performances considering growing nodes number (selected among 4, 10, 20, 50, 100, 200) inside the scenario described in Table 1. In particular, for our protocols we use the optimal values found in the previous section. The results are summarized in the following.
Figure 9 shows the delivery probability results. All protocols follow the same behavior of the "C.F. SnW" protocol, except for the two controlled flooding versions that have lower delivery probability for scenarios with lower number of nodes (4 to 20). The tendency of the "C.F. Limited" is peculiar, as when the number of nodes increases its delivery probability decreases. This can be explained since the protocol allows a limited number of repliers, then when the number of node increases the probability to find the destination node inside a maximum "hopping-range" decreases.
Figure 10 shows the overhead metric. We can see that all protocols follow the same tendency while increasing the number of nodes, except for the SnW, that limits its overhead to an upper bound. In the following, we see that this behavior is accompanied by higher latency values.

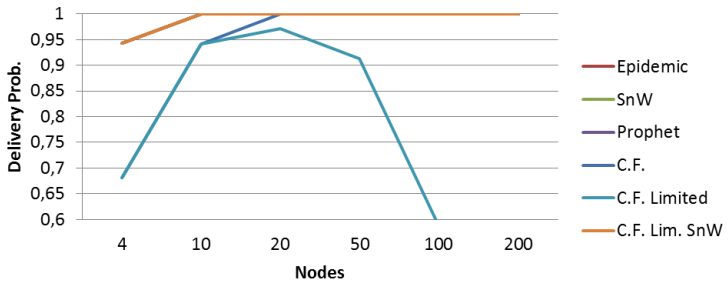

Figure 9: Delivery Probability vs number of nodes

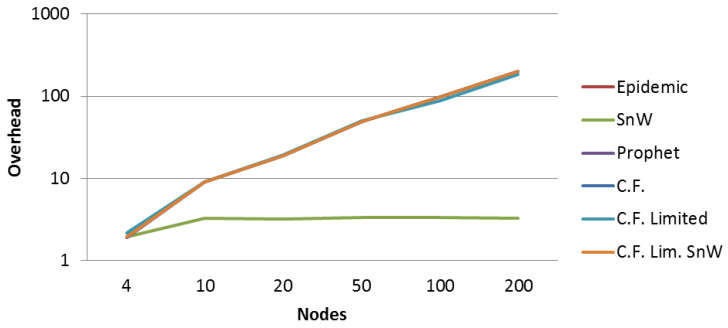

Figure 10: Overhead vs number of nodes

Figure 11 shows average Latency. The SnW protocol obtains the highest values among all. All "C.F." protocols and "Epidemic" have an upper bound when the number of nodes increases. Regarding the average latency, for scenarios with more than 50 nodes, a remarkable performance is achieved by Prophet, with same latency value achieved by "C.F. SnW".

Figure 12 shows the Hop Count metric. All protocols stay in a range from 1,5 to 2 except for "C.F" that has a linear growing tendency with the number of nodes. The "C.F." with limited number of copies seems to grow linearly up to a certain value, the hop count metric is saturated by the protocol behavior that limits the number of replies.

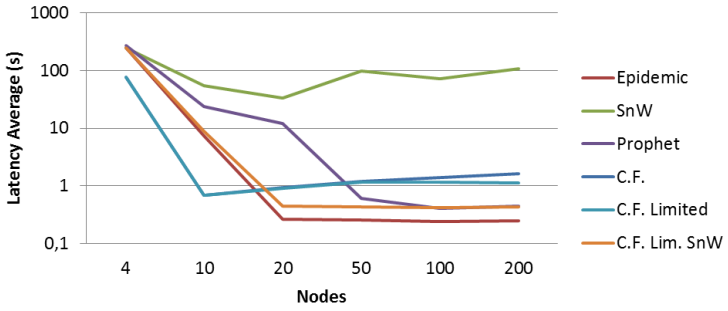

Figure 11: Latency vs number of nodes

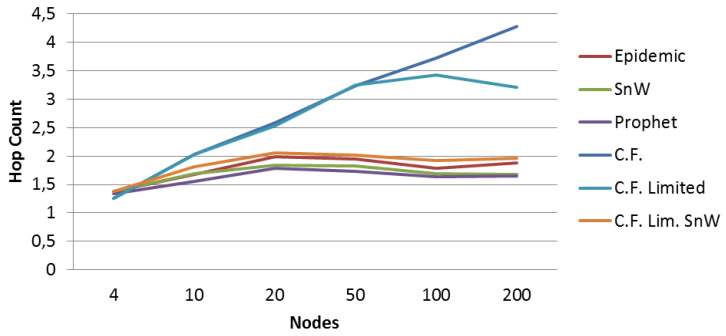

Figure 12: Hop Count vs number of nodes

Finally, we tried to measure the relative protocol complexity by using the simulation time as an indirect index. In fact, in the same scenario, the simulation time of one protocol can be attributed directly to the number of operations that each node performs. Figure 13 shows the simulation time for each protocol for increasing number of nodes. We can see that all protocols have a linear dependency with the number of nodes. As it is possible to see, the worst performance is achieved by Epidemic. The smartest ones are the "C.F." and "C.F. Lim." that achieve lowest simulation time. "SnW" achieves simulation time about twice greater than these ones.

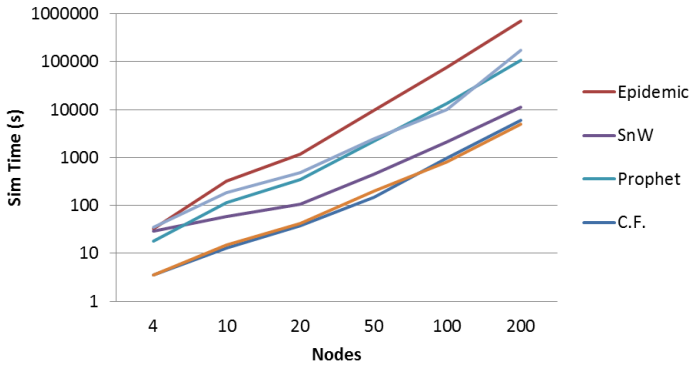

Figure 13: Simulation Time vs number of nodes

4. Matching Applications with Protocols

After having characterized the protocols in Section 3, we finally propose a selection criteria for applications considered in Section **Errore. L'origine riferimento non è stata trovata.**. We note that applications have specific constraints that lead to particular requirements, but here we take into account common driver factors for each application domain, regarding coverage area, number of nodes and maximum latency. These drivers will be given a qualitative index (Low, Medium, High) and linked to one of the metrics for DTN protocol exposed in Section 2. Such index is listed in Table 2.

	Driver	Metric Constraints	Protocol
Smart Cities	• Large area coverage • Large number of nodes • Medium/high response time • Medium Node Computational Power • Medium Mobility	A. Medium/High Delivery Probability B. Medium/Low Latency C. Low average hop count D. Medium Protocol Complexity	Prophet

Smart Building	• Small/Medium area • Medium Number of nodes • Low response time • Medium battery life • Low Node Computational Power • Low node mobility	A. High Delivery Probability B. Low Average Hop C. Low Overhead D. Low Protocol Complexity	Spray & Wait
Tele Medicine	• Small area • Small of nodes • High response time • Low/high battery life • Medium Computational Power • Low node mobility	A. High Delivery Probability B. Low Average Hop C. Low Latency	C.F. SnW
Smart Vehicle	• Small/Medium area • Small/Medium Number of nodes • High response time • Low battery life (rechargeable) • Medium/High Node Computational Power • High node mobility	A. High Delivery Probability B. Low Average Hop C. Low/Medium Latency D. Low/Medium Protocol Complexity	C.F. SnW
Intrusion Detection / reconnaissance	• Medium/Large area • Large Number of nodes • Low/Medium response time • Large battery life • Low/Medium Node Computational • Low node mobility	A. High Delivery Probability B. Medium Latency C. Low Protocol Complexity	C.F.
Industrial / Commercial	• Medium/Large area • Medium/Large Number of nodes • High response time • High battery life • Low Node Computational Power • Low/high node mobility	A. High Delivery Probability B. Low Average Hop C. Low Latency	Epidemic

Table 2: Recommended protocol for each application cluster

5. Conclusions

In this article we have classified WSN applications according to their scope and their application domain. A set of DTN protocols, some from literature, some proposed by authors, have been analyzed and compared. Focusing on the drivers that lead a particular application domain, we have given an example of how to use simulation results in order to select a protocol, given the application domain. The proposed method have been practically used for a defense WSN called Masterzone [15].

Acknowledgement

The authors are grateful to Dr. Alfonso Farina from Selex E.S. for his valuable directions.

References

[1] J. Kim et al., "*Performance evaluation of synchronous and asynchronous MAC protocols for wireless sensor networks*", in "Proceedings of the 2008 Second International Conference on Sensor Technologies and Applications". IEEE Computer Society, 2008. p. 500-506. 23 August 2008.

[2] Delay tolerant networking research group. http://www.dtnrg.org, last access 22/07/2014.

[3] F. P. Miller, A.F.Vandome, J. McBrewster, "*Delay-tolerant networking*". Alpha Press, 2010.

[4] C.S. Malavenda, F. Menichelli, and M.Olivieri, "*Delay Tolerant, Low Power Protocols for Large Security-Critical Wireless Sensor Networks*" Journal of Computer Networks and Communications, Hindawi, 2012.

[5] C.S. Malavenda, F. Menichelli, and M.Olivieri, "*A Regulation-Based Security Evaluation Method for Data Link in Wireless Sensor Network*" Journal of Computer Networks and Communications, Hindawi, 2014.

[6] C.S. Malavenda, F. Menichelli, and M.Olivieri, "*Wireless and Ad Hoc sensor networks: An industrial example using delay tolerant, low power protocols for security-critical applications*", Lecture Notes in Electrical Engineering Volume 289, pp. 153-162, 2014

[7] "ONE" simulator web page http://www.netlab.tkk.fi/tutkimus/dtn/theone/, last access 22/07/2014.
[8] A.Keränen, J.Ott & T. Kärkkäinen, "*The ONE simulator for DTN protocol evaluation*". In: Proceedings of the 2nd International Conference on Simulation Tools and Techniques. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), p. 55, March 2009.
[9] F.Menichelli, M.Olivieri, *"TikTak: A Scalable Simulator of Wireless Sensor Networks Including Hardware/Software Interaction"*, Wireless Sensor Network Journal, Volume 2, issue 111, pp. 815-822, Scientific Research Publishing, 2010.
[10] K. A. Harras, K. C. Almeroth, E.M. Belding-Royer, "*Delay tolerant mobile networks (dtmns): Controlled flooding in sparse mobile networks*". In: NETWORKING 2005. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems. Springer Berlin Heidelberg, p. 1180-1192, May 2005.
[11] A. Vahdat, D. Becker, "*Epidemic routing for partially connected ad hoc networks*". Technical Report CS-200006, Duke University, 2000.
[12] T. Spyropoulos, K. Psounis, C. S. Raghavendra, "*Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks*", WDTN '05 Proc. of ACM SIGCOMM workshop on Delay tolerant networking, August 2005.
[13] A. Lindgren, A. Doria, E. Davies, S. Grasic, "*Probabilistic Routing Protocol for Intermittently Connected Networks*", ISSN: 2070-1721, 2012.
[14] P. Bijal, D. Krupa, P. Vyomal, "*Spray and Wait Routing Protocol in Delay Tolerant Networks*", International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 4, Issue 5, May 2014.
[15] www.selex-si-uk.com/pdf/Masterzone.pdf, last access 03/01/2014