Weighted Probabilistic Data Dissemination (WPDD)

Falko Dressler
Dept. of Computer Science 7
University of Erlangen, Germany

Technical Report 05/06

Abstract

Most routing methods for wireless sensor networks (WSN) are based on general-purpose ad hoc routing techniques. Usually, on-demand path determination based on unambiguous addresses is used to control and manage the current network topology. In WSN, such address-based approaches often lead to severe problems due to several issues: node mobility, energy-saving long sleep-cycles due to infrequent message exchange, and the more general problem of dynamic addressing. Data-centric routing schemes such as flooding or gossiping solve these problems but may lead to congestion or starvation. In this paper, we propose a new data-centric forwarding technique, named weighted probabilistic data dissemination (WPDD), which is based on gossiping. In contrast to other proposals, this method includes an inherent adaptation to changing network conditions. Based on several priorities and probabilities, congestion control is supported as well as prioritized data communication. This paper outlines the working behavior of WPDD and its additional features. A simulation model was created to analyze the approach. The results promise an efficient deployment in different scenarios.

1. Introduction

With the proliferation of wireless sensor networks (WSN), the need for more robust and efficient communication methods increases [8]. A sensor network is build of a large number of small devices consisting of sensors, wireless radio communication interfaces, small processing and storage components, and a battery [2]. Due to the strong resource restrictions and the limited available energy, many approaches have been developed to make applications and network protocols energy-aware. Therefore, WSN differ from other ad hoc networks in terms of energy-efficient operation [3]. At the same time, sensor networks show different communication behaviors. Depending on the scenario, sensor nodes are mobile, i.e. the topology of the ad hoc network changes rapidly over the time, and one can observe varying frequencies of data transmissions, i.e. most of the time a few messages per hour have to be transmitted while sometimes many messages per second need to be forwarded.

These properties have strong influence on the communication in WSN. Usually, classical ad hoc routing techniques are used in sensor networks. Such algorithms can be divided into two classes: pro-active and re-active routing protocols [1]. While pro-active routing mechanisms manage topology information all the time prior any data communication, re-active solutions initiate the setup of a communication path with the presence of the first data packet to be transmitted. In the area of ad hoc routing, the best-known example for pro-active routing protocols is DSDV [17]. AODV [16, 18] and its successor DYMO [6] are frequently used examples for re-active solutions. Depending on the scenario, both solutions have advantages and drawbacks, e.g. in terms of overhead for routing messages and the time to set up a route. Nevertheless, their use in sensor networks is limited due to some reasons. First, the previously mentioned specific characteristics of WSN, i.e. the low-energy and low-resource operation and, possibly, the node mobility, make it impractical to collect and manage global state information for optimized path calculation. Secondly, the precondition for all these routing techniques to have unambiguous addresses assigned to each node is hard to achieve. Considering a scenario with thousands of sensor nodes randomly deployed in a field that have to organized themselves to measure data and transmit them to a proper destination. It is not feasible to pre-program all nodes with a unique address. Even if it would be possible, it makes the replacement of failed sensors even more difficult. Therefore, address-less communication methods are needed. Data-centric communication intends to fill this gap. Starting from different flooding schemes [13], probabilistic methods have been studied to reduce the overhead caused by flooding each message though the entire network. The most prominent solutions are rumor routing [5] and gossiping [4, 10]. Admittedly, these approaches have their own challenges. Either, the message overhead can lead to network congestion or the message delivery may fail ("the gossip dies out").

In this paper, we present a new approach, named *weighted probabilistic data dissemination* (WPDD). WPDD is based on the concepts of gossiping, i.e. it used node probabilities to determine the ratio of messages that is forwarded to a particular neighbor. Additionally, it supports the weighting of messages. This weighting is used for several reasons. First, it allows to *prioritize messages*, i.e. to increase the probability for these messages to reach their destination. Secondly, the same weighting – dynamically used – allows to adapt the number of messages per time to be forwarded. This concept is used to provide an inherent *congestion control* feature. Thirdly, a specific property of sensor networks can be exploited to reduce the number of messages in the network. Aggregation schemes can be used for *sensor data fusion*, i.e. the combination of multiple messages into a single aggregated one. In this case, the weighting is an importance factor, i.e. the more aggregated the data is, the higher the probability to reach its destination. The basic model of WPDD is described in this paper and performance evaluation measures are included that show the capabilities of our approach. Using a simulation model, we compared WPDD to pure flooding and other gossiping solutions.

The main contribution of this paper can be summarized as follows. Weighted probabilistic data dissemination is presented as a new approach for data routing in WSN. The main capabilities of our solution, adaptation to the application scenario, inherent congestion control and priority control schemes, and self-organizing operation, are outlined and simulation results are discussed that compare WPDD to flooding and gossiping.

The rest of the paper is structured as follows. Routing approaches for ad hoc and sensor networks are discussed in a related work part in section 2. This includes a survey of other data-centric routing solutions. WPDD is outlined and discussed in section 3. The simulation model used to analyze the behavior of the algorithm and to compare it to other techniques is depicted in section 4. This section also includes all simulation results. Finally, the paper is summarized with some conclusions in section 5.

2. Related Work

An overview to ad hoc routing in general can be found for example in [1]. Characteristics of wireless sensor networks and networking solutions are discussed in [2, 8, 15]. Ad hoc routing can be classified into two categories: address-based and data-centric routing. Basically, there is a third category, geographical routing, but we exclude these approaches here because of the limited applicability in GPS-less environments. While there are many approaches for address-based routing, there is still little work on data-centric approaches. Before we introduce the most prominent data-centric forwarding techniques, the basic communication model for WSN should be recalled. Usually, dedicated sink nodes exist that are interested in sensor data. Some scenarios rely on a single base-station while others allow for spontaneously chosen sinks. In case of a single or few sinks, algorithms that operate on some degree of knowledge about current sinks are well-suited for sensor networks. For arbitrary communications (or rapidly and spontaneously chosen sinks), such mechanisms lead to unnecessarily high overhead for sink (or interest) management. In the following, we focus on data-centric routing and discuss both types of algorithms in more detail.

Single/multiple sinks – Directed diffusion [11] is one of the best-known data-centric routing algorithms. It operates as follows. First, so called interest is propagated (flooded) though the entire network. This interest includes information about the kind of sensor data a node is interested in. Based on the interest propagation, gradients are created pointing towards the sink. Finally, if sensor data is available, it is distributed along the gradient towards the sink, usually on an optimal path. Rumor routing [5] is using agents to detect the presence of and the interest on events. Basically, it is trying to fill the gap between query flooding and event flooding. Both approaches try to detect on demand the necessity of data transfers and create state information accordingly. As mentioned before, in case of mobility or infrequent sensor measures, such state information tends to become an overhead. Concerning mobility, there is an extension to directed diffusion with mobility support available [7]. Nevertheless, regular interest flooding cannot be prevented and may lead to energy wastage and, possibly, to congestion in case of multiple sinks.

Arbitrary communication – The second category of data-centric algorithms is based on flooding principles. Regardless of their names, all these flooding or gossiping approaches [4, 10, 12-14] have the same underlying concept. A message is flooded though the entire network until it eventually finds its destination. The differences between the approaches in the literature arise in their decision process whether a message should be forwarded or discarded. Basic gossiping, i.e. probabilistic forwarding, suffers from the high possibility that messages die out on their way to the destination (if the probability fraction is too small) or that the same overloading effects appear as in flooding (if the fraction is too large). Therefore, most of the named approaches try to elaborate a self-learning system to makes good guesses whether a particular direction should receive a copy of the message or not.

Our own approach, WPDD, falls also into this category and can be seen as an extension to the gossip-based ad hoc routing scheme by Haas et al. [10]. The same measures to improve the "quality of guesses" can be applied to WPDD as well. Two properties distinguish WPDD from the other solutions: its possible adaptation to different application scenarios and its inherent congestion control features.

3. Methodology

Stateless communication in WSN with possible adaptation to the current network behavior and application demands is addressed by our proposed new algorithm "WPDD – Weighted Probabilistic Data Dissemination". The main objectives can be summarized as follows: data-centric data dissemination, i.e. address-less operation, inherent congestion control and quality of service features, i.e. overload detection and prevention and priority-oriented data transmission. In the following subsections, the algorithm is presented in detail.

3.1 Basic Model

The basic behavior of our approach is shown in Fig 1. On the left hand side, a sensor/actuator network is shown. Sensors are responsible for two different tasks: to measure environmental information and to forward this information to corresponding sinks, e.g. actuators or analyzing entities. The dashed lines represent possible communication paths in the WSN. In our model, we do not assume a single (possibly central) base station to be the destination of all measurement data. We allow arbitrary numbers and locations of sink nodes. Additionally, the communication scheme is not fixed. The model is able to provide one-to-one (unicast), one-to-many (multicast), and one-to-all (broadcast) communication.

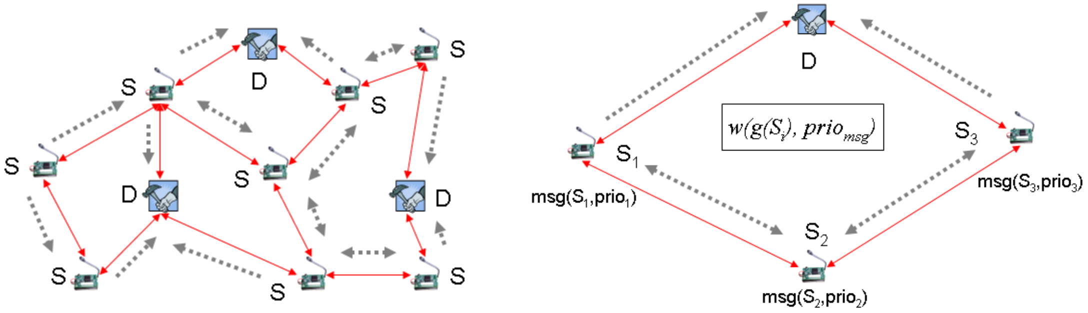

Fig 1. Data forwarding in a sensor network using controlled flooding

On the right hand of Fig 1, the node behavior is depicted in more detail. Basically, each sensor node creates messages and sends this data with a given probability to its neighbors. The messages themselves are composed of sensor information describing type and value of the content and a message priority ($prio_{msg}$). The forwarding algorithm is meant to operate on locally available information only, i.e. to prevent any kind of global state describing the overall network behavior.

Starting with pure flooding, the method for evaluating can be described in an easy way. The only way to prevent messages from circling endlessly in the network is to include a time or hop limit (TTL – time to live). The number of messages in the network then depends on the number of nodes, the number of interconnections, the network diameter, and the maximum lifetime of single messages. The forwarding decision just depends on the TTL:

$$\forall msg: flooding\ (\ msg\ ) = (\ msg_{TTL} > 0\ ) \tag{1}$$

The gossip algorithm as presented in [10] was intended to operate on a given node probability (associated to each sensor node). A GOSSIP() function was used to calculate the probability to forward the message to a particular neighbor. In the analysis of the algorithm, the authors recognized the high possibility for a gossip to die out if either the path is too long (in terms of hops) or the network density if too sparse. They developed improved solutions for the GOSSIP() algorithm to include parameters such as the guessed (or learnt) network density by analyzing the number of duplicated messages. Basically, gossiping adds an additional node specific probability used for forwarding decisions (S_i represents sensor node i):

$$\forall msg: gossip\ (\ msg, S_i\ ) = flooding\ (\ msg\ ) \wedge GOSSIP\ (\ S_i\ ) \tag{2}$$

WPDD was developed as an extension for the mentioned gossip algorithm. In addition to node-oriented probabilities, weighted probabilistic data dissemination allows for message priorities and smooth congestion control. Basically, the forwarding probability can be calculated as follows ($P(msg_{prio})$ represents the importance of a particular message and $W(S_i)$ represents the node weighting):

$$\forall msg: wpdd\ (\ msg, S_i\ ) = flooding\ (\ msg\ ) \wedge (\ P\ (\ msg_{prio}\ ) > W\ (\ S_i\ )\ ) \tag{3}$$

Obviously, the forwarding probability can be adjusted similarly to the GOSSIP() function. Additionally, the message priorities allow for prioritizing messages based on application-depended restrictions. During the development of WPDD, we had the following two mechanisms in mind: (1) in WSN, usually, the normal behavior of a given environment is monitored. In special cases, faults or alarms have to be distributed through the network. Such priority messages must be handled differently. (2) message aggregation and data fusion techniques are used to reduce the number of messages in the network. Admittedly, such aggregated messages have a much higher impact on subsequent analysis than non-aggregated ones. Therefore, a scheme for optimized handling of such messages is needed.

3.2 Algorithm

In the following, the basic algorithm is depicted and discussed that is used on all nodes in the network. The algorithm can be configured to allow multicast and broadcast communication. Additionally, it can be enhanced to adapt to changing network conditions, e.g. in terms of network congestion.

As shown in the pseudo-code, the algorithm works as follows. First, a set of message types $msg_{local}[]$ is initialized that contains information whether a particular message can be processed by the local node or not. The outer loop waits for new messages. After the reception of a message, it is first checked if it can be processed locally. Then (line 6), the priority $P(msg_{prio})$ of the message is calculated. Please refer to section 4 for specific calculation functions. In an inner loop, each neighbor is examined by calculating an according weighting $W(N_n)$ for this neighbor. This can be done using any kind of gossip function. Finally, a message is forwarded if the expression $W(N_n) < P(msg_{prio})$ becomes true, i.e. if the desired distribution range is higher than the estimated node weighting.

```
1)  initialize $msg_{local}[]$;
2)  for each received message msg(type, prio)
3)     if $msg_{type} \in msg_{local}[]$ then
4)        process msg
```

```
   endif
   calculate P($msg_{prio}$)  (describing the distribution range)
   for each neighboring node $N_n$
      calculate node weighting W($N_n$)  (e.g. using the GOSSIP() function)
      if W($N_n$) < P($msg_{prio}$) then
         forward message to $N_n$
      endif
   endfor
endfor
```

Using this algorithm, flooding can be expressed if *P()* and *W()* are set to deliver fixed values corresponding to *P()>W()*. Gossiping can be simulated by setting *P()* to a static value in [0,1]. Finally, different variants of weighted probabilistic data dissemination can be expressed by allowing different distributions / probability functions for *P()* and *W()*.

3.3 Additional Features

Especially in case of self-organizing data communication as achieved using weighted probabilistic forwarding schemes, congestion control must be addressed as a necessary component in order to prevent overload situations and to make sure that the network can respond to high-priority requests at any time [9]. Therefore, local feedback information can be used to adapt the dissemination strategy based on the network behavior by modifying *W()*, i.e. the message forwarding probability, and the behavior of *P()*, i.e. the handling of $prio_{msg}$. The resulting behavior is depicted in Fig 2. The timeliness of received measures at the sink node depends primarily on the message generation rate. If only few messages are created per time interval, the accuracy of the final measurements is being reduced. On the other hand, a low generation rate induces less congestion in the network. To adapt this rate according to the requirements (network congestion, energy savings), global feedback information is used. Again, this process is an optimization issue as depicted in Fig 2.

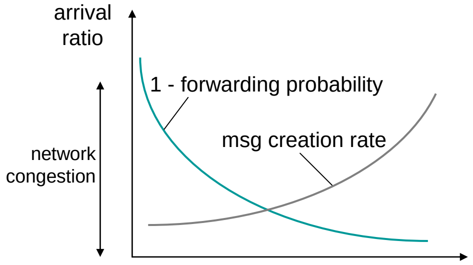

Fig 2. Optimization process for adaptive regulation of the network utilization using two competitive feedback loops

The basic requirements on dynamic self-calibrating congestion control can be summarized as follows. The algorithm must be able (1) to maintain control even if some links get temporarily saturated, (2) to give priority to important messages, and (3) to prevent starvation of particular transmissions.

Primarily, the method is based on the number of successfully received messages *N* during the last time interval *T*. The algorithm works as follows:

```
for each received message msg(type, prio)
   update message counter N(M,T)
   identify importance factor $I_M$
   Calculate probability P(N,$I_M$)
   if exponentialDistribution(P,T) then
      forward msg
   endif
endfor
```

The adaptation as seen from a global point of view is summarized in Fig 2. The mechanism can be used to carefully adapt the network behavior in a typical WSN by employing appropriate feedback information. It fulfills all the mentioned three requirements.

4. Simulation Model and Results

In order to evaluate the proposed method, we created a simulation model to analyze overhead, performance, and reliability of weighted probabilistic data dissemination in comparison to the approaches such as flooding and gossiping. The preliminary results demonstrate the capabilities of our proposal. Depending on the application scenario, the parameters must be set and adapted accordingly in order to achieve optimal a network behavior.

4.1 Model and Parameters

The simulation model was implemented using the discrete event simulation tool AnyLogic[1]. We created the topology shown in Fig 3. To compare the performance of the discussed dissemination schemes, multiple setups were created to reflect the network behavior in different scenarios.

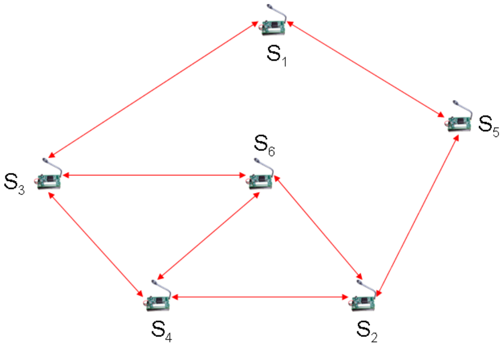

Fig 3. Simulation model consisting of partially meshed nodes

Several globally defined parameters are used in every node. These values allow an easier configuration of the simulation model in order to achieve comprehensive results. The following list shows all global parameters that influence the performance of the node communication or control the used communication scheme, respectively:

- MinDelay: minimum transmission delay for each node (2ms)
- MaxDelay: maximum transmission delay for each node (8ms)
- LossRatio: probability for packet losses on each communication channel
- MsgPrio: priority of each packet specifying its importance; the following values can be defined:
    - [0, 1]: each node uses the same priority for created messages
    - -1: a uniform distribution in [0, 1] is used to specify the priority for each packet
    - -2: the message priority is specified for each node separately
- CommType: used communication type; in general, a packet will be forwarded if the following condition holds: $P(i) > W(k)$, where
    - i is the priority of a given message
    - $P(i)$ is a predefined function based on the message priority i
    - k is the weighting of a node
    - $W(k)$ is a predefined function based on the node weighting

---

[1] http://www.xjtek.com/

The node behavior can be configured in order to switch to a different dissemination scheme. In the simulation model, we distinguish five communication types (depicted as CTi in the following) using different functions for $P()$ and $W()$:

1. Flooding (CT1): $P(i) = x > W(k)$ (node weighting and message priority have no influence)
2. Gossiping (CT2): $P(i) = i$; $W(k) = Random(0, 1)$
3. Weighted Probabilistic Data Dissemination (Types 3, 4, and 5)
   CT3: $P(i) = i$; $W(k) = 1 – k$
   CT4: $P(i) = i$; $W(k) = 1 - k *Random(0, 1)$
   CT5: $P(i) = i*Random(0, 1)$; $W(k) = 1 - k *Random(0, 1)$

Network behavior	**Loss ratio**	**Node weighting**	**Message priority**
1-1 (unicast, node 1 transmits to node 4)	0	0.3	0.8
	0.3	0.8	0.3
1-2 (multicast, node 1 sends to nodes 2 and 4)	0.3	0.3	0.8
		0.8	0.3
2-1 (unicast, nodes 1 and 2 send to node 4)	0.3	0.3	50%=0.3 & 50%=0.8
		0.8	
loop (every node sends to another on: 1-2-3-4-5-6-1)	0.3	50%=0.3 & 50%=0.8	50%=0.3 & 50%=0.8
		0.3	
		0.8	
all-1 (unicast, all nodes send to node 4)	0	50%=0.3 & 50%=0.8	50%=0.3 & 50%=0.8
	0.3	0.3	one node=0.8 & other nodes=0.3
		0.8	

Table 1. Evaluated combinations of the parameter settings (node weighting / message priority "50%=0.3 & 50%=0.8" means that 50% out of all nodes (1, 3, 5) are configured to a weighting / priority of 0.3, while all others are set to 0.8 (2, 4, 6)

In addition, several node specific parameters are used to control the local activities, i.e. the node behavior:

- No: each node is identified by an unambiguous identification
- Destination: all generated packets are sent to this destination; the destination varies with different scenarios, i.e. simulation setups, but cannot be changed within a single simulation experiment
- PacketRate: frequency for generating and transmitting packets at each node (set to 1s)
- PacketPrio: message priority used to modify the importance of a particular message; this value in the range of [0, 1] is used only of the globally configured MsgPrio is set to -2
- NodePrio: weighting of the node in [0, 1] to calculate the forwarding probability
- MsgInterval_1: to evaluate the network behavior, two intervals can be configured in which statistical measures are generated; in our experiments, MsgInterval_1 was set to 10s
- MsgInterval_2: similar to MsgInterval_1, set to 60s

In Table 1, the different measurement setups are summarized. Basically, they all differ in the characterization of the loss ratio, node weighting, and message priority. Additionally, different network behaviors are modeled.

4.2 Simulation Results and Discussion

SumRcv / SumSend – in the optimal case, the quotient SumRcv/SunSend is equal to one, i.e. every packet is sent once and it was properly received. In all experiments, we only consider unreliable data communication. Therefore, the probability to receive each packet increases with each copy that was sent. Admittedly, this also increases the network overload. The most important aspect is a *possible adaptation* to the current network behavior. The following experiments (Fig 4 and Fig 5) show the performance of flooding, gossiping, and weighted probabilistic data dissemination.

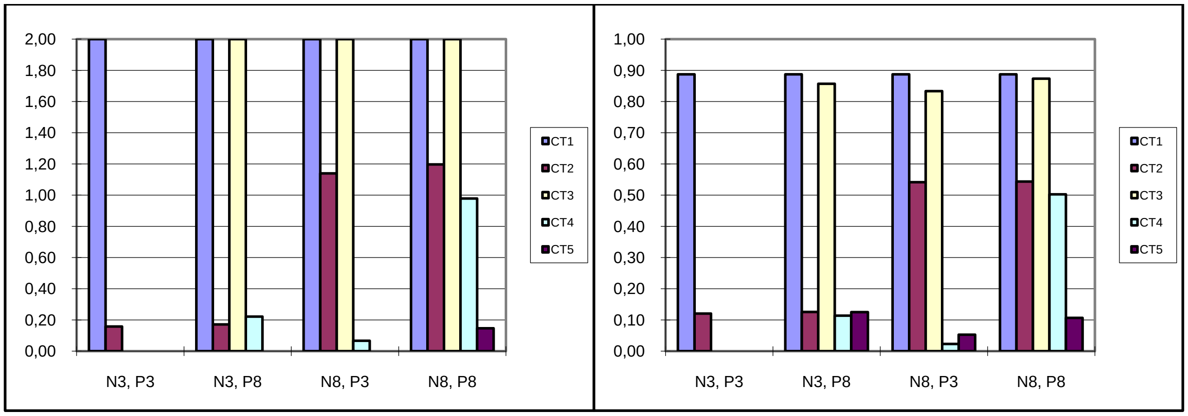

Fig 4. Evaluation of the successful receive rate of the different communication types (left: node 1 to node 4, loss ratio 0; right: node 1 to node 4, loss ratio 0.3)

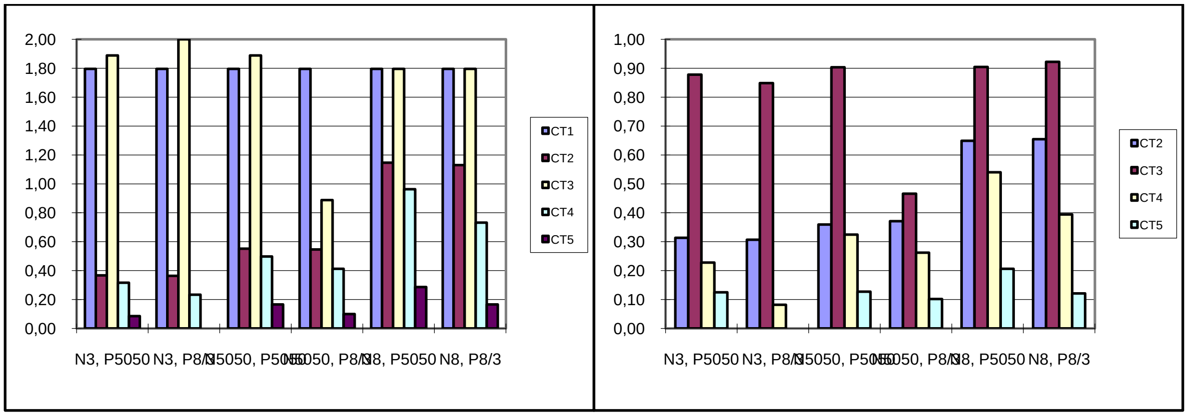

Fig 5. Evaluation of the successful receive rate of the different communication types (left: all nodes to node 4, loss ratio 0; right: all nodes to node 4, loss ratio 0.3)

Discussion:

- <u>Flooding</u>: in case of a loss ratio SumRcv/SumSend is equal to the number of generated packets multiplied by the number of possible paths though the network (in our case 2); quite efficient (about 0.9) if the loss ratio increases
- <u>Gossiping</u>: strongly depending on the node priority; in all experiments, results less than 0.4 have been achieved. In general, the number of successfully received packets is reduced in gossiping because of unnecessarily long paths without any possible adaptation

- Weighted probabilistic data dissemination: the rate depends on the current configuration; if the packet loss ratio is small, communication type 4 should be preferred while type 3 is more appropriate for higher loss rates. The configuration can easily be adapted during network operation

Hop count – preferably, the necessary number of hops though the network is minimized by the routing protocol. Nevertheless, without global state information, optimal paths cannot be calculated. The next experiment was used to evaluate possible differences between the different communication types. The results as shown in Fig 6 are obvious. Flooding always includes the best path while gossiping and WPDD tend to choose random paths through the network.

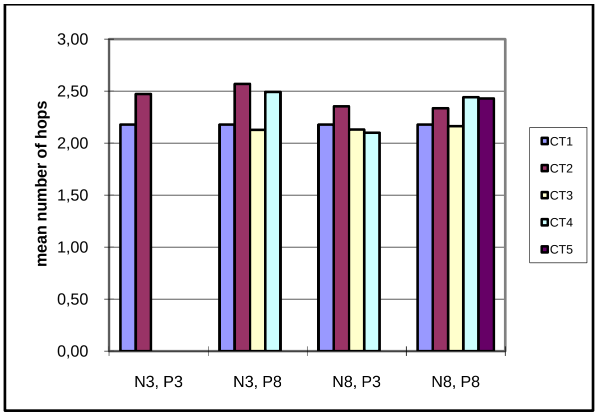

Fig 6. Comparison of the path length (number of hops) for all communication types

Msg/MsgAll – The efficiency of the communication method can be described by the quotient of successfully received messages and the number of received copies for each packet. The closer to one, the more efficient is the algorithm because there was no unnecessary copy transported though the network leading to higher network congestion.

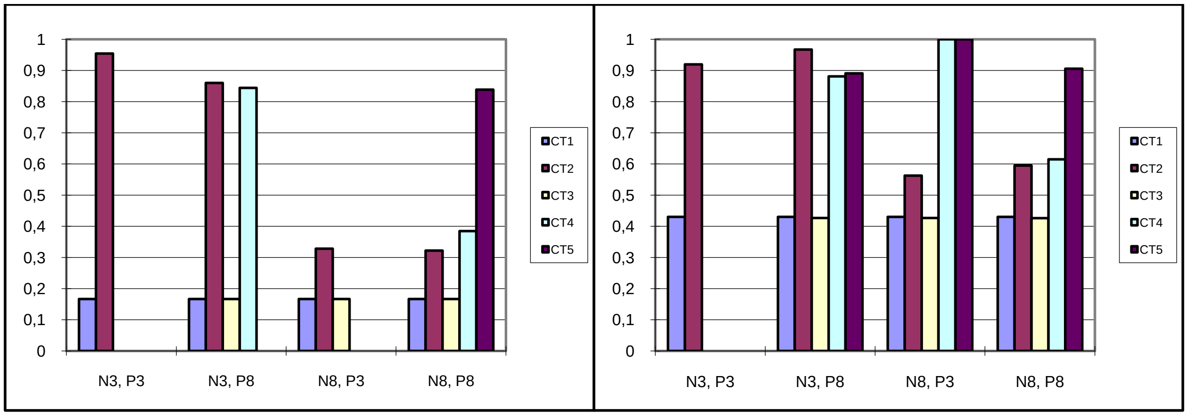

Fig 7. Evaluation of the overhead (duplicated messages) of the different communication types (left: node 1 to node 4, loss ratio 0; right: node 1 to node 4, loss ratio 0.3)

Discussion:

It can be seen that flooding and weighted probabilistic data dissemination (type 3) are the least efficient communication methods. Many copies of each packet are generated and transmitted towards the destination. This is especially the case if the network loss ratio is negligible. On the other hand, type 4 and 5 seem to be most appropriate because the efficiency measure is optimal. Nevertheless, this value is achieved only because so few messages arrive at the destination that the probability to see duplicated messages is about zero.

In summary, it has become obvious that there is no “best” communication scheme. Depending on the requirements of the particular application scenario (minimal network load vs. small end-to-end loss), “best” results can be achieved by adapting the algorithm to the current needs. Only weighted probabilistic data dissemination allows such a free configuration based on message priorities and node weightings.

5. Conclusions

In this paper, we presented a new algorithm for data-centric routing in wireless sensor networks named WPDD (weighted probabilistic data dissemination). WPDD is based on existing gossiping approaches and, therefore, benefits from previously developed solutions to increase the probability for given packets to travel over long transmission paths (many hops) and through sparsely deployed networks (e.g., as described in [10]). The main characteristics of WPDD are its flexibility and the free adaptation to changing network conditions. Additionally, it inherently supports message prioritization for aggregation and data fusion techniques. WPDD works on locally available information only. Therefore, there is no overhead through control and maintenance of topology or other state information. Controllability is provided by means of variable node behavior (node weighting) and message handling (message priorities). Congestion control can be added as shown in [9]. We created a simulation model in order to evaluate the algorithm. We compared WPDD to pure flooding and to gossiping (we left out mechanisms discussed in [10] because they are applicable to gossiping as well as to WPDD). The results show that, depending on the network behavior, always a particular configuration of WPDD succeeded. Future work includes to improve the algorithm with aggregation features and to work out schemes for adjusting weightings and priorities for larger subnetworks.

References

[1] K. Akkaya and M. Younis, "A Survey of Routing Protocols in Wireless Sensor Networks," *Elsevier Ad Hoc Network Journal*, vol. 3 (3), pp. 325-349, 2005.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *IEEE Communications Magazine*, vol. 40 (8), pp. 102-116, August 2002.

[3] I. F. Akyildiz and I. H. Kasimoglu, "Wireless Sensor and Actor Networks: Research Challenges," *Elsevier Ad Hoc Network Journal*, vol. 2, pp. 351-367, October 2004.

[4] C. L. Barrett, S. J. Eidenbenz, and L. Kroc, "Parametric Probabilistic Sensor Network Routing," Proceedings of International Conference on Mobile Computing and Networking, San Diego, CA, USA, 2003.

[5] D. Braginsky and D. Estrin, "Rumor Routing Algorithm For Sensor Networks," Proceedings of First Workshop on Sensor Networks and Applications (WSNA), Atlanta, Georgia, USA, September 2002.

[6] I. Chakeres, E. Belding-Royer, and C. Perkins, "Dynamic MANET On-Demand (DYMO) Routing," Internet-Draft, draft-ietf-manet-dymo-03.txt, October 2005.

[7] A. Choksi, R. P. Martin, B. Nath, and R. Pupala, "Mobility Support for Diffusion-based Ad-Hoc Sensor Networks," Rutgers University, Department of Computer Science, Technical Report DCS-TR-463, April 2002.

[8] D. Culler, D. Estrin, and M. B. Srivastava, "Overview of Sensor Networks," *Computer*, vol. 37 (8), pp. 41-49, August 2004.

[9] F. Dressler, "Locality Driven Congestion Control in Self-Organizing Wireless Sensor Networks," Proceedings of 3rd International Conference on Pervasive Computing (Pervasive 2005): International Workshop on Software Architectures for Self-Organization, and Software Techniques for Embedded and Pervasive Systems (SASO+STEPS 2005), Munich, Germany, May 2005.

[10] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-Based Ad Hoc Routing," Proceedings of IEEE INFOCOM 2002, June 2002, pp. 1707-1716.

[11] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," Proceedings of 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCOM'00), Boston, MA, USA, August 2000, pp. 56-67.

[12] D. Kempe, J. Kleinberg, and A. Demers, "Spatial Gossip and Resource Location Protocols," *Journal of the ACM (JACM)*, vol. 51 (6), pp. 943-967, November 2001.

[13] T. J. Kwon and M. Gerla, "Efficient Flooding with Passive Clustering (PC) in Ad Hoc Networks," *ACM SIGCOMM Computer Communication Review*, 2002.

[14] J. Luo, P. T. Eugster, and J.-P. Hubaux, "Pilot: Probabilistic Lightweight Group Communication System for Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 3 (2), pp. 164-179, April 2004.

[15] C. Margi, "A Survey on Networking, Sensor Processing and System Aspects of Sensor Networks," University of California, Santa Cruz, Report, February 2003.

[16] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, July 2003. (http://www.ietf.org/rfc/rfc3561.txt)

[17] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Computer Communications Review*, pp. 234-244, 1994.

[18] E. Royer and C. Perkins, "An Implementation Study of the AODV Routing Protocol," Proceedings of IEEE Wireless Communications and Networking Conference, Chicago, IL, September 2000.