Worst Case Dimensioning of Wireless Sensor Networks under Uncertain Topologies

Jens B. Schmitt[1] and Utz Roedig[2]

[1]Distributed Computer Systems Lab (DISCO), University of Kaiserslautern, Germany
Email: jschmitt@informatik.uni-kl.de
[2]Mobile & Internet Systems Laboratory (MISL), University College Cork, Ireland
Email: utz@cs.ucc.ie

***Abstract* - Many applications in wireless sensor networks require an absolute predictable network behaviour under all possible traffic and routing conditions. This requires network dimensioning with respect to crucial parameters such as energy usage, message delay and buffer requirements. Network calculus provides a method set that can be used to prove that all possible traffic conditions under a given routing-implied topology can be supported. As it is often impossible to fix the routing topology before or even during the operation of a sensor network, an uncertainty arises with which network calculus cannot deal directly. In this paper, we make two contributions: 1) We demonstrate how to apply network calculus in a fixed sensor network topology. 2) We show how to incorporate topology uncertainty into these calculations.**

***Index Terms* - sensor networks, network calculus, network dimensioning, routing topology.**

I. Introduction

A. Motivation

Application areas for wireless sensor networks might be for example production surveillance, traffic management, medical care or military applications. In these areas it is crucial to ensure that the sensor network is functioning even in a worst case scenario. Here, functioning often means that critical information is neither lost nor delayed beyond a certain bound. If a sensor network is used for example for production surveillance it must be ensured that messages indicating a dangerous condition are not dropped or excessively delayed since otherwise serious consequences as for example a machine breakdown may occur. Hence, if functionality in worst case scenarios cannot be proven, people might be in danger and the production system might not be certified by authorities.

As it may be difficult or even impossible to systematically generate the worst case in a real world test scenario or even in a simulation an analytical framework is desirable that allows a worst case analysis in sensor networks. Network calculus [1] is a relatively new tool that allows worst case analysis of packet-switched communication networks and has been applied to model wired IP-based networks [2]. However, network calculus is general enough such that it can also be tailored to be used as a framework to analyze wireless sensor networks with a fixed topology as it is shown later in the text.

For many application areas of sensor networks it is not clear what the routing-implied network topology will look like before or even during the network is deployed. In this paper, the influence of this topology uncertainty during the planning and dimensioning of wireless sensor networks is analyzed. This approach allows to dimension a sensor network and its nodes in a way that correct network operation, even under worst case conditions and uncertainty of the network topology, can be ensured.

B. Problem Scope

As nodes in a sensor network are power constrained, methods to reduce power consumption are essential. One heavily used method is the usage of radio duty cycles [3], [4]. The transceiver system is periodically set into a sleep state in which energy is saved but communication cannot take place. Thus, each node delays messages while forwarding them according to the used duty cycle and its traffic load. This message delay may be a critical factor in a sensor network that has to be controlled. Obviously, it can be traded for energy consumption. Additionally, the message delay has an influence on the buffer/ memory requirements of a node as messages have to be stored before they can be forwarded.

The aforementioned sensor network parameters *duty cycle*, *message transfer delay* and *buffer requirements* can be investigated analytically using a network calculus approach (see next section). The results of the calculation can then be used to dimension the sensor network and its nodes so that support for worst case traffic scenarios can be guaranteed.

However, it is necessary to assume, besides other factors, a specific network topology before network calculus can be applied. This might be difficult in many application scenarios as the exact routing topology often cannot even roughly be known beforehand. As a very obvious example imagine that the sensor nodes are dropped from a plane. Nevertheless, some parameters restricting the resulting topology might be known. The number of nodes in a sensor field or the maximum hop-distance in the field might be examples for such restricting factors. In particular, such restrictions might be enforced by careful topology control of the sensor network, as for example in [5]. While our proposal does not depend on such restrictions it will be discussed how much a worst case dimensioning can benefit from such prerequisites.

C. Outline

In the following section it is shown how network calculus can be tailored so that a worst case analysis of the relevant quantities in sensor networks is possible under a given fixed topology. In Section III the frequent case of uncertainty about the topology of a sensor network during its planning is investigated and the concept of worst case topologies is incorporated into sensor network calculus. Section IV provides some numerical results and Section V reviews related work before in Section VI the paper is concluded.

II. Sensor Network Calculus

Before the network calculus based system model of wireless sensor networks is presented we provide a short overview on the relevant results from basic network calculus (for an in-depth treatment consult the excellent text by Le Boudec and Thiran [1]).

A. Some Background on Network Calculus

Network calculus can be interpreted as a system theory for *deterministic* queueing systems, based on min-plus algebra. What makes it different from traditional queueing theory is that it is concerned with worst case rather than average case or equilibrium behavior. It thus deals with bounding processes called arrival and service curves rather than arrival and departure processes themselves. Next some basic definitions and notations are provided before some basic results from network calculus are summarized.

Definition: The input function $R(t)$ of an arrival process is the number of bits that arrive in the interval $[0,t]$. In particular, $R(0) = 0$ and R is wide-sense increasing.

Definition: The *output function* $R^o(t)$ of a system S is the number of bits that have left S in the interval $[0,t]$. In particular, $R^o(0) = 0$ and R^o is wide-sense increasing.

Definition: Min-Plus Convolution. Let f and g be wide-sense increasing and $f(0) = g(0) = 0$. Then their convolution under min-plus algebra is defined as

$$(f \oplus g)(t) = inf_{0 \le s \le t}\{f(t-s) + g(s)\}$$

Definition: Min-Plus Deconvolution. Let f and g be wide-sense increasing and $f(0) = g(0) = 0$. Then their deconvolution under min-plus algebra is defined as

$$(f \otimes g)(t) = sup_{s \ge 0}\{f(t+s) - g(s)\}$$

Now, by means of the min-plus convolution, the arrival and service curve are defined.

Definition: Arrival Curve. Let α be a wide-sense increasing function α such that $\alpha(t) = 0$ for $t < 0$. α is an arrival curve for an input function R iff $R \le R \oplus \alpha$. It is also said that R is α-smooth or R is constrained by α.

Definition: Service Curve. Consider a system S and a flow through S with R and R^o. S offers a service curve β to the flow iff β is wide-sense increasing and $R^o \ge R \oplus \beta$.

From these, it is now possible to capture the major worst case properties for data flows: maximum delay and maximum backlog. These are stated in the following theorems.

Theorem 1: Backlog Bound. Let a flow *R(t)*, constrained by an arrival curve α, traverse a system S that offers a service curve β. The backlog *x(t)* for all *t* satisfies:

$$x(t) \le sup_{s \ge 0}\{\alpha(s) - \beta(s)\} = v(\alpha, \beta) \quad (1)$$

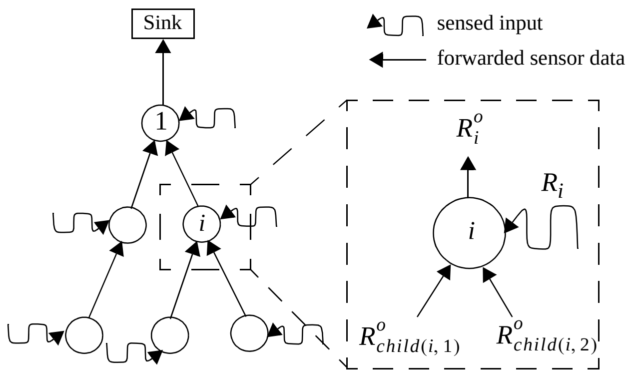

Figure 1 - Sensor Network Model.

Theorem 2: Delay Bound. Assume a flow $R(t)$ constrained by arrival curve α traverses a system S that offers a service curve β. At any time t, the virtual delay $d(t)$ satisfies:

$$d(t) \le sup_{s \ge 0}\{inf\{\tau \ge 0 | \alpha(s) \le \beta(s+\tau)\}\} = h(\alpha, \beta) \quad (2)$$

As a system theory network calculus offers further results, in particular a bound on the output when traversing a single node.

Theorem 3: Output Bound. Assume a flow $R(t)$ constrained by arrival curve α traverses a system S that offers a service curve β. Then the output function $R^o(t)$ is constrained by

$$\alpha^o = \alpha \otimes \beta \quad (3)$$

Theorem 4: Concatenation of Nodes. Assume a flow *R(t)* traverses systems S_1 and S_2 in sequence where S_1 offers service curve β_1 and S_2 offers β_2. Then the resulting system S, defined by the concatenation of the two systems offers the following service curve to the flow:

$$\beta = \beta_1 \oplus \beta_2 \quad (4)$$

B. Sensor Network System Model

Within the traffic that is modeled only the sensor reports are taken into account. Traffic generated from the base station towards the nodes (e.g. interests [6] to set up the network structure and configure the nodes) is explicitly not taken into account. This is considered feasible based on the typical situation that the traffic flowing towards the sensors is magnitudes lower than traffic caused by the sensing events. Furthermore, it is assumed that the routing protocol being used forms a sink tree towards a base station (see Fig. 1).

Each sensor node i senses its environment and thus is exposed to an input function R_i corresponding to its sensed input traffic. If sensor node i is not a leaf node of the tree then it also receives sensed data from all of its child nodes $child(i,1), \ldots, child(i,n_i)$, where n_i is the number of child nodes of sensor node i. Sensor node i forwards/processes its input which results in an output function R_i^o from node i towards its parent node (see Fig. 1).

Now the basic network calculus components, arrival and service curve, have to be incorporated. First the arrival curve $\bar{\alpha}_i$ of each sensor node in the field has to be derived. The input of each sensor node in the field, taking into account its sensed input and its children's input, is given by:

$$\bar{R}_i = R_i + \sum_{j=1}^{n_i} R^o_{child(i,j)} \quad (5)$$

Thus, the arrival curve for the total input function for sensor node *i* is given by:

$$\bar{\alpha}_i = \alpha_i + \sum_{j=1}^{n_i} \alpha^o_{child(i,j)} \quad (6)$$

Second, the service curve has to be specified. The service curve depends on the way packets are scheduled in a sensor node which mainly depends on link layer characteristics. More specific, the service curve depends on how the duty cycle and therefore the energy-efficiency goals are set.

Finally, the output of sensor node *i*, i.e. the traffic which it forwards to its parent in the tree, is constrained by the following arrival curve:

$$\alpha^o_i = \bar{\alpha}_i \otimes \beta_i = \left(\alpha_i + \sum_{j=1}^{n_i} \alpha^o_{child(i,j)}\right) \otimes \beta_i \quad (7)$$

In order to calculate network-wide characteristics like the maximum message transfer delay or local buffer requirements especially at the most challenged sensor node just below the sink (which is called node 1 from now on) an iterative procedure to calculate the network internal flows is required:

1) Let us assume that arrival curves for the sensed input α_i and service curves β_i for sensor node *i*, *i* = 1,...,*N*, are given.
2) For all leaf nodes the output bound α^o_i can be calculated according to (3). Each leaf node is marked "calculated".
3) Now, for all nodes only having children which are marked "calculated" the output bound α^o_i can be calculated according to (7) and they can again be marked "calculated".
4) If node 1 is "calculated" terminate, otherwise go to step 3.

After the network internal flows are computed according to this procedure, the local worst case buffer requirements B_i and per node delay bounds D_i for each sensor node *i* can be calculated according to Theorem 1 and 2:

$$B_i = v(\bar{\alpha}_i, \beta_i) = sup_{s \ge 0}\{\bar{\alpha}_i(s) - \beta_i(s)\} \quad (8)$$

$$D_i = h(\bar{\alpha}_i, \beta_i) = sup_{s \ge 0}\{inf\{\tau \ge 0 | \bar{\alpha}_i(s) \le \beta_i(s+\tau)\}\} \quad (9)$$

To compute the total message transfer delay $\bar{D}_i$ for a given sensor node *i* the per node delay bounds on the path *P(i)* to the sink need to be added:

$$\bar{D}_i = \sum_{i \in P(i)} D_i \quad (10)$$

The maximum message transfer delay in the sensor network can straightforwardly be calculated as

$$\bar{D} = max_{i = 1, \ldots, N}\{\bar{D}_i\} \quad (11)$$

Discussion. Readers knowledgeable in network calculus may wonder about the hop-by-hop calculation of the total delay as specified in (11) and whether it would not be possible to derive a network-wide service curve based on the concatenation result of Theorem 4. While due to the traffic aggregation inside the network the concatenation result cannot be applied directly, there is in fact a way to still derive a network-wide service curve based on modified service curves that take into account the effects of cross-traffic on a data flow [1]. However, the bounds achieved in this way are not necessarily lower (whether they are or are not depends on the actual parameters of arrival and service curves). Furthermore, we believe that the hop-by-hop calculation will lend itself better towards integrating in-network processing into future, more elaborate extensions of the model.

Often, sensor network applications may regard message transfer delay only as a constraint and primarily care about maximizing their lifetime. The length of the duty cycle, and thus the energy consumption properties of the sensor nodes, are incorporated into the service curve as will be discussed in Section II.D. Hence, instead of calculating delay bounds and buffer requirements as described above, the calculations could also start with a given delay/buffer requirement and work out the length of the duty cycle and thus the power consumption level.

In summary, the sensor network calculus framework allows from a worst case perspective to relate the local characteristics *sensing activity* and *buffer requirements* to the global characteristics *message transfer delay* and *network lifetime*.

C. Instantiation of the Arrival Curve

Now a specific arrival curve for the sensing input at each of the sensor nodes has to be selected. The simplest option in bounding the sensing input at a given sensor node is based on its *maximum sensing rate* which is either due to the way the sensing unit is designed or limited to a certain value by the sensor network application's task in observing a certain phenomenon. The arrival curve for a sensor node *i* corresponding to simply putting a bound on the maximum sensing rate is given by

$$\alpha_i(t) = p_i t = \gamma_{p_i, 0}(t) \quad (12)$$

Here $\gamma_{p,b}$ denotes an affine function with slope *p* and *b* as y-axis intercept.

The set of sensible arrival curve candidates is large. The more knowledge on the sensing operation and its characteristics is incorporated into the arrival curve for the sensing input the better the worst case bounds become. We consider it a strength of the sensor network calculus framework that it is open with respect to arbitrary arrival curves.

D. Instantiation of the Service Curve

The service curve captures the characteristics with which sensor data is forwarded by the sensor nodes towards the sink. It abstracts from the specifics of the link layer and makes a statement on the minimum service that can be assumed.

A typical and well known example of a service curve from traditional traffic control in a packet-switched network is given by the *Rate-Latency Service Curve*

$$\beta_{R,T}(t) = R(t-T)^+ \quad (13)$$

where the notation $(x)^+$ denotes *x* if $x \ge 0$ and 0 otherwise. This service curve results from the use of many popular packet schedulers (for example Weighted Fair Queueing (WFQ) [7]).

The latency term nicely captures the worst case characteristics induced by the usage of a duty cycle concept. Whenever the duty cycle approach is applied there is the chance that sensed data or data to be forwarded arrives after the last duty cycle of the next hop is just over and thus a fixed latency occurs until the forwarding capacity is available again. For the forwarding capacity it is assumed that it can be lower bounded by a fixed rate which depends on transceiver speed, the chosen link layer protocol and again the duty cycle. So, with some new parameters the following service curve at sensor node *i* is obtained:

$$\beta_i(t) = \beta_{f_i, l_i}(t) = f_i(t - l_i)^+ \tag{14}$$

Here f_i and l_i denote the forwarding rate and latency for node *i*. The latency l_i corresponds to the length of the sleep period for the chosen duty cycle and the forwarding rate f_i to the link capacity times the duty cycle.

III. Incorporating Uncertainty on the Sensor Network Routing Topology

In sensor network dimensioning an uncertainty exists, not only due to traffic and service variation but potentially also with respect to the topology. The topology may not be known at the time of dimensioning the network and can only be assumed to be random. Nevertheless, the influence of the topology needs to be factored in due to the burstiness increase phenomenon resulting from Theorem 3 and the per-hop delay calculation.

One straightforward idea would be to enumerate all possible topologies and analyze them to find the worst case topology that may be encountered and on which the dimensioning should be based to be on the safe side. If no constraints on the topology besides being a tree are given, a large number of possible topologies is obtained. In particular it is known that the number of non-isomorphic rooted trees T_N can be calculated from the recurrence relation [8]:

$$T_N = \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{d|i} T_d d\right) T_{n-i+1} \text{ with } T_0 = T_1 = 1 \tag{15}$$

Here the notation $d|i$ means all *d* which divide *i*. As shown by Otter [9] the ratio between two subsequent elements of the series T_N is constant in the limiting case

$$\lim_{n \to \infty} \frac{T_N}{T_{N-1}} = \alpha = 2.955765... \tag{16}$$

with α known as Otter constant. That means in effect that roughly $T_N = O(3^N)$, i.e. the number of non-isomorphic trees consisting of *N* nodes increases exponentially. To enumerate and evaluate all possible topologies is therefore computationally very expensive for larger numbers of sensor nodes without putting further constraints on the routing topology.

In fact, often in practical cases restrictions on possible topologies can be made, in particular the depth of the resulting tree and the number of children per node may be limited, which, of course, decreases the search space. Placing a constraint on the number of children a node may have, is not very effective in reducing the search space, though, as again Otter [9] determined the number of binary trees to be

$$B_N \approx 0.7916 \times 2.483^N \times N^{-3/2} \approx O(2.5^N) \tag{17}$$

which still exhibits an exponential growth with larger number of sensor nodes. Also restricting the depth of the tree is not too promising if it cannot be kept at a very low number compared to the total number of sensor nodes *N*, since most of the possible trees can be expected at medium depths.

A more promising line of thought for large sensor networks is to put some effort into which topologies are candidates to exhibit the worst case behavior. In the context of this paper a topology is a worst case topology if it has either

- the highest buffer requirements at node 1,
- the highest message transfer delay, or
- the shortest network lifetime.

As is shown below (for a limited case, though) these three criteria often coincide. It is pretty intuitive that a line topology (the tree with *N*-1 edges) is often a candidate as worst case topology for all of the above metrics, since it maximizes the effect of burstiness increase as the sensed data travels towards the sink. However, in general this cannot be assumed, but depends on the specifics of the examined sensor network scenario, in particular on the arrival and service curves being chosen. For example, matters can become complicated if homogeneity of the sensor nodes is not assumed. Yet, in many practical cases, a worst case topology might be deduced for given arrival and service curves by inspection.

As a worst case topology might again lead to extremely conservative bounds, it may be very helpful to incorporate realistic constraints on maximum depth and outdegree as mentioned above. Under this assumption and under the selection of arrival and service curves from Section II a fairly general observation on worst case topologies can be made.

Definition: (o,d)-Constrained Tree. A tree is (*o*,*d*)-constrained if all of its nodes have an outdegree of less than *o* and none is more than *d* edges away from the root.

Definition: Maximally Deep (o,d)-Constrained Tree. A tree with *N* nodes is a maximally deep (*o*,*d*)-constrained tree if it is (*o*,*d*)-constrained and the sum of distances from each node to the sink $\sum_{i=1}^{N} d_i$ is maximal (d_i denotes the number of edges from node *i* to the root), i.e. there is no other (*o*,*d*)-constrained tree with a larger sum of distances.

Theorem 5: In a homogeneous sensor network of *N* nodes with a maximum sensing rate arrival curve $\gamma_{p,0}$ and a rate-latency service curve $\beta_{f,l}$ and an (*o*,*d*)-constraint on the network topology a sensor network topology which puts as many nodes as possible below node 1 and has a maximally deep (*o*,*d*)-constrained tree below node 1 constitutes a worst case topology.

Proof: The proof is based on the observation that the worst case topology is the one that *maximizes the arrival curve at node 1* of the sensor network, as this results in the worst case buffer requirements (and maximum message transfer delay as well as the shortest network lifetime). Each sensor node *i* below node 1 contributes to the arrival curve at node 1 in the following way

$$\Delta_i \bar{\alpha}_{\text{node 1}} = \gamma_{p,\,(d_i - 1)pl} \tag{18}$$

This can be easily checked by calculating the influence of sensor node *i* hop by hop towards the sink:

$$\begin{aligned} \alpha_i^o &= (\gamma_{ap,\,bpl} + \gamma_{p,\,0}) \otimes \beta_{f,\,l} = \gamma_{(a+1)p,\,(a+b+1)pl} \\ \alpha_{parent(i)}^o &= (\alpha_i^o + \gamma_{cp,\,dpl} + \gamma_{p,\,0}) \otimes \beta_{f,\,l} \\ &= \gamma_{(a+c+2)p,\,(2a+b+c+d+3)pl} \\ &\dots \end{aligned} \tag{19}$$

whereas without sensor node *i* injecting traffic (but still forwarding) the following would be obtained

$$\begin{aligned} \alpha_i^o &= \gamma_{ap,\,bpl} \otimes \beta_{f,\,l} = \gamma_{ap,\,(a+b)pl} \\ \alpha_{parent(i)}^o &= (\alpha_i^o + \gamma_{cp,\,dpl} + \gamma_{p,\,0}) \otimes \beta_{f,\,l} \\ &= \gamma_{(a+c+1)p,\,(2a+b+c+d+1)pl} \\ &\dots \end{aligned} \tag{20}$$

Here, the constants *a*, *b* and *c*, *d* represent the influence of further traffic converging at node *i* and its parent, respectively. If it is assumed that the maximum possible number of nodes that can be put below node 1 is denoted by *N'* (which may often be equal to *N*-1 depending on the constraint *d*) the arrival curve at node 1 is given by (assuming without loss of generality that the sensor nodes are numbered adequately)

$$\bar{\alpha}_{\text{node 1}} = \gamma_{p,\,0} + \sum_{i=1}^{N'} \Delta_i \bar{\alpha}_{\text{node 1}} = \gamma_{N'p,\,pl\sum_{i=1}^{N'}(d_i - 1)} \tag{21}$$

which is clearly maximized if the tree below node is a maximally deep *(o,d)*-constrained tree.

■

Note that maximally deep (*o*,*d*)-constrained trees are easy to construct by recursively putting nodes as deep as possible in the tree before creating new branching points.

Certainly, Theorem 5 is only a special case and to obtain a general result would be much more complex but 1) we believe it to be of some value as a typical scenario and 2) it demonstrates what is meant by finding the worst case topology by inspection.

IV. Numerical Results

The numerical results in this section shall illustrate the abstract discussion on worst case topologies of Section III.

A. *Experimental Design*

The measures of interest when making dimensioning decisions for a sensor network and thus the response variable of the investigation are the message transfer delay, the required buffer size at node 1 (and in homogeneous sensor networks thus of all nodes), and the sensor network lifetime.

The major factors affecting the response variables are the topological constraints, the forwarding characteristics governed by the chosen duty cycle, the number of nodes in the field and the sensing activity. The latter two factors are considered secondary in the following investigation and thus are fixed: the number of nodes for all experiments is assumed to be *N = 1000* nodes and the maximum sensing rate is chosen as *P = 0.258 bit/s* which roughly corresponds to sending a packet every 18 minutes, which for some of the examined scenarios is the highest possible sensing rate. The topological constraints and the forwarding characteristics are considered primary factors and their influence on the response variables shall be examined. In particular, with respect to the forwarding characteristics Mica-2 nodes [4] under TinyOS are assumed as state-of-the-art sensors. Mica-2 supports a nominal link speed of 19.2 kbit/s and is assumed to be operated with duty cycles of 1% and 11.5%[1] which implies forwarding rates/latencies of *f = 258 bit/s* / *l = 1.096 ms* and *f = 2488 bit/s* / *l = 0.096 ms*, respectively.

B. *Results and Discussion*

The results of the experiments are presented in Fig. 2 and Fig. 3. Fig. 2 shows the effect of the primary factors topological constraints and forwarding characteristics on the message transfer delay. In particular, the constraint on the depth of the tree topology is varied from *d = 5* to *d = 50* (on the x-axis) and the constraint on the outdegree *o* is chosen from the set *{2,5,10}* (different curves). Furthermore, the different forwarding characteristics resulting from the two different duty cycles of 1% and 11.5% are provided by the two groups of curves. For the buffer requirements at node 1 essentially the same type of graph is provided in Fig. 3. Note that for both graphs a logarithmic scale on the y-axis is chosen in order to be able to show both groups of curves corresponding to the different duty cycles in one graph.

From Fig. 3 some observations on the effect of the primary factors on the maximum buffer requirements can be made. It is obvious that the possible depth *d* of the topology has a strong effect on the maximum buffer requirements (note again that the y-axis uses a logarithmic scale). This effect is particularly strong for very low *d* and decreases as *d* increases. Hence to achieve low bounds on the required buffer space in sensor nodes it is very important to be able to make a fairly strict restriction on the possible depth of the tree topology.

The other topological constraint on the possible outdegree *o* has less influence than *d* but is considerable if *d* is fairly low

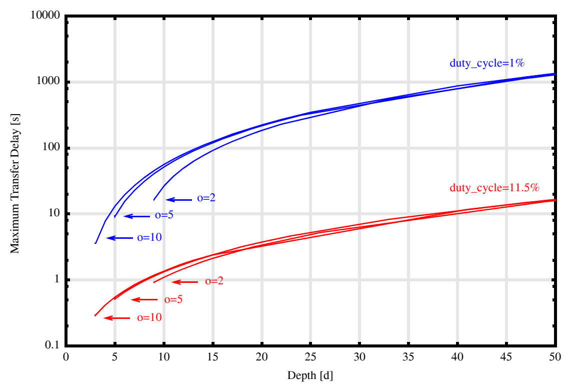

Figure 2 - Maximum Transfer Delay

1. Values are taken from the TinyOS code. The packet length is 36byte, the preamble length for 1% duty cycle is 2654Byte.

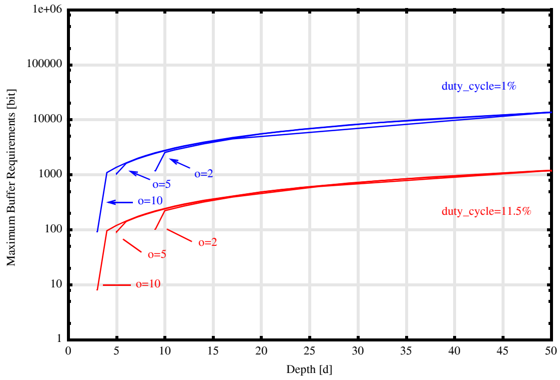

Figure 3 - Maximum Buffer Requirements

whereas for larger *d* it hardly influences the maximum buffer requirements. Therefore, only if a good constraint on the depth of the topology is possible a sensor network design must take care of the outdegree constraint.

The duty cycle obviously has a very strong influence. The lower duty cycle of 1% achieves a maximum buffer requirement of one order of magnitude lower than for a duty cycle of 11.5% for almost all *d*. Of course, on the other hand the network lifetime is decreased by roughly one order of magnitude for the higher duty cycle as well. So, from a sensor network design perspective a trade-off between memory equipment and network lifetime has to be made here.

The effects of the primary factors on the message transfer delay are illustrated in Fig. 2. Essentially, the observations that can be made are very much the same as for the maximum buffer requirements. The only thing that is remarkable in addition, is the effect of the duty cycle on the message transfer delay. Here the lower duty cycle results in a roughly two orders of magnitude higher delay than for the larger duty cycle. Interestingly, a duty cycle one order of magnitude larger thus results in a two orders of magnitude reduction of the message transfer delay. So, from the perspective of message transfer delay there is a strong incentive to have a higher duty cycle.

V. Related Work

Much research in wireless sensor networks deals with the problem of assessing characteristic parameters (e.g. message transfer delay, buffer requirements, network lifetime) of a wireless sensor network with a given, or enforced routing topology (e.g [10], [11]). However, to our knowlede none of these analyze the theoretical worst case as we are addressing it with our network calculus based approach. Yet, for a safe network dimensioning we consider the worst case to have a high significance. Furthermore, we incorporate uncertainty on the routing topology in a generic way such that we can leverage a tightly controlled routing topology but do not depend on it in general as the aformentioned research work.

To our knowledge only one publication [12] exists in the context of wireless sensor networks that uses network calculus as analytical tool. The latter paper deals with the very different problem of developing a theoretically sound congestion control in distributed sensor networks. The authors make some basic observations on their flow controller using network calculus but do not consider actually modelling the sensor network itself using network calculus which has been the goal of the paper at hand.

VI. Conclusion and Outlook

An analytical framework based on network calculus to dimension sensor networks has been presented. This sensor network calculus approach allows dimensioning of sensor networks in a way such that correct network operation, even under worst case conditions and uncertainty of the routing topology can be ensured. It has been demonstrated how the various trade-offs and interdependencies between node power consumption, node buffer requirements and message transfer delay can be described using the sensor network calculus framework. Especially it has been shown how the sensor network calculus can be used if the topology is unknown at the time when the network is dimensioned.

Sensor networks will be used in the future for critical applications. In this case the sensor network must be properly dimensioned for all, even worst case, scenarios to ensure continuous and safe operation. We believe the sensor network calculus framework has the potential to become such a proper dimensioning methodology for wireless sensor networks.

References

[1] J.-Y. Le Boudec and P. Thiran. Network Calculus - A Theory of Deterministic Queueing Systems for the Internet. Springer, Lecture Notes in Computer Science, LNCS 2050, 2001.

[2] R. Braden, D. Clark, and S. Shenker. Integrated Services in the Internet Architecture: an Overview. RFC 1633, June 1994.

[3] A. Barroso, J. Benson, T. Murphy, U. Roedig, C. Sreenan, S. Bellis, K. Delaney, B. O'Flynn. The DSYS25 Sensor Platform. Demo at SenSys2004.

[4] Crossbow Technology INC. Mica-2 Data Sheet. http://www.xbow.com.

[5] A. Cerpa and D. Estrin. ASCENT: Adaptive Self-Configuring Sensor Networks Topologies. INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. Volume: 3, 23-27, June 2002.

[6] C. Intanagonwiwat, R. Govindan and D. Estrin. Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. In Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM 2000), August 2000.

[7] A. Demers, S. Keshav, and S. Shenker. Analysis and Simulation of a Fair Queueing Algorithm. In Proceedings of ACM SIGCOMM, pp. 3-12, August 1989.

[8] S.R. Finch. Mathematical Constants. Cambridge, England: Cambridge University Press, 2003.

[9] R. Otter. The Number of Trees. Ann. Math. 49, 583-599, 1948.

[10] A. Salhieh, J. Weinmann, M. Kochhal, and L. Schwiebert. Power Efficient Topologies for Wireless Sensor Networks. In Proceedings of the 2001 International Conference on Parallel Processing (ICPP), pp. 156-163, September 2001.

[11] L. Li and P. Sinha. Throughput and Energy Efficiency in Topology-controlled Multi-hop Wireless Sensor Networks. In Proceedings of the WSNA'03, September 2003.

[12] J. Zhang, K. Premaratne and Peter H. Bauer. Resource Allocation and Congestion Control in Distributed Sensor Networks - a Network Calculus Approach. In Proceedings of Fifteenth International Symposium on Mathematical Theory of Networks and Systems, August 2002.