

# Statistical Response Time Bounds in Randomly Deployed Wireless Sensor Networks

Steffen Bondorf

disco | Distributed Computer Systems Lab  
University of Kaiserslautern, Germany  
bondorf@informatik.uni-kl.de

Jens B. Schmitt

disco | Distributed Computer Systems Lab  
University of Kaiserslautern, Germany  
jschmitt@informatik.uni-kl.de

**Abstract**—Response time bounds are important for many application scenarios of wireless sensor networks (WSN). Often, during the planning phase of a WSN its topology is not known. It rather results from a deployment process. This makes the provision of deterministic response time bounds difficult. In this paper, we strive for statistical response time bounds in WSNs that take the stochastic nature of the deployment process into account. Based on a Monte Carlo method we derive estimates for quantiles of the maximum response time distribution under uncertainty about the topology. In numerical experiments we show that the long but light tail of this distribution causes considerably lower bounds compared to the deterministic one even under small violation probabilities and, yet, on the other hand compare favourably with the median of the distribution.

**Index Terms**—Performance Bounds, Wireless Sensor Networks, Quantile Estimation.

## I. INTRODUCTION

There are many applications of wireless sensor networks (WSN) that require bounds on the response time of any sensor node to report on events. Response times depend on a number of factors, most importantly: the traffic intensity and its pattern, the forwarding capabilities of the nodes, their lifetime goals and thus their duty cycle, and the *topology* of the WSN. Sensor network calculus [1], [2] is a method to compute response time bounds based on these factors. Usually such calculations are carried out during the planning phase of a WSN in order to support decisions like how many nodes and sinks are required, etc. In fact, most of the above factors can be assumed to be known or controllable. Yet, the topology of the WSN is often not perfectly known in advance. So, the sensor network calculus cannot be directly applied. To the best of our knowledge, the only work that tries to factor in the uncertainty about the topology of a WSN due to random deployment is [3]. In that work, worst-case topologies have been proven in the framework of deterministic sensor network calculus, thus arriving at deterministic response time bounds.

In this paper, we show that under uncertainty about the topology worst-case response time bounds can be far off from realistic response times. We show that the discrepancy mainly depends on the degree of uncertainty about the topology. Most importantly, we demonstrate that by departing from deterministic worst-case bounds to statistical bounds we can improve the bounds considerably without losing too much assurance. This statement is substantiated by a Monte Carlo

based method to compute statistical response time bounds with a controllable precision.

## II. SENSOR NETWORK CALCULUS

This is a basic overview of the Sensor Network Calculus (SNC). Detailed explanations can be found in [1], [2].

### A. Basic Sensor Network Calculus

To apply the basic SNC, the network topology has to be known. For example, a tree-structured network topology with a sink at the root and  $n$  sensor nodes can be used. Next, the network traffic has to be described in terms of the so-called arrival curves for each node. An arrival curve defines an upper bound for the input traffic of a node. Leaf nodes in the network must handle traffic according to the sensing function they perform; for example, a node might sense an event and create a data packet at the maximum rate of one packet every second. This sensing pattern can be expressed as an arrival curve  $\alpha_i$ . Non-leaf nodes handle traffic according to their own sensing pattern and the traffic they receive from other nodes. To calculate the output, the so-called service curve  $\beta_i$  is used. The service curve specifies the worst-case forwarding capabilities of a node. The necessary forwarding latencies are defined by the nodes' forwarding characteristics. From the arrival and service curves, it is possible to calculate the output bounds for each node. Using those bounds, it is possible to compute the effective input  $\bar{\alpha}_i$  for each node. After that, the local per-node delay bounds  $D_i$  for each sensor node  $i$  can be calculated according to basic network calculus results [4]:

$$D_i = h(\bar{\alpha}_i, \beta_i) = \sup_{s \geq 0} \{ \inf \{ \tau \geq 0 : \bar{\alpha}_i(s) \leq \beta_i(s + \tau) \} \}.$$

To compute the total information 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:

$$\bar{D}_i = \sum_{j \in P(i)} D_j$$

Clearly, a bound on the maximum information transfer delay in the sensor network can then be calculated as  $D = \max_{i=1, \dots, N} \bar{D}_i$ . The whole procedure is called *total flow analysis* (TFA) because the entire traffic arriving at a given node is treated in an aggregate fashion. Examples for the use of this calculus can be found, e.g., in [5]–[7].

## B. Advanced Sensor Network Calculus

Although, TFA is a straightforward method for applying network calculus in the domain of wireless sensor networks, there is room for improvement with respect to the quality of the calculated performance bounds. This is because of the fact that a concatenation result for consecutive nodes offering service curves is not exploited by TFA. In particular, we can exploit and even extend the concatenation result towards the so-called Pay Multiplexing Only Once analysis (PMOO) described in [8], to compute an end-to-end service curve for the specific flow of interest from one sensor node to the sink. Due to the sink-tree structure of the network, all flows that join the flow of interest remain multiplexed until the sink, making it possible to calculate the total information transfer delay  $\bar{D}_i$  for a given sensor node  $i$  by using a flow-specific end-to-end service curve. PMOO can be shown to deliver a tight bound for sink-trees of homogeneous nodes [9]. When compared to the addition of the nodal delay bounds, as done by TFA, this results in considerably less pessimistic bounds as each interfering flow's burst has to be taken into consideration only once.

## C. Worst-Case Topologies

The most direct usage of the sensor network calculus assumes a known topology. Yet, this is exactly where we want to part from in this paper. Previous work of ours established for certain (but typical) WSN scenarios the concept of worst-case topologies [3] with respect to the maximum response time. Under no knowledge at all about the topology this may result in very conservative bounds. Therefore the work in [3] also reasoned about the worst-case topology when at least some restrictions could be made. Here we provide the basic notions and the central result from [3]:

**Definition 1.** ( $(o, d)$ -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 2.** (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 3.** In a homogeneous sensor network of  $n$  nodes each with an token bucket arrival curve  $\alpha_{r,b}$ , a rate-latency service curve  $\beta_{R,T}$  and an  $(o, d)$ -constraint on the topology, a sensor network topology which consists of as many nodes as possible below the node next to the sink and has a maximally deep  $(o, d)$ -constrained tree below this node constitutes a worst case topology with respect to the maximum response time.

An important contribution of this paper is to investigate how conservative these *deterministic* response time bounds are in terms of random deployment processes and to devise an approach to provide for *statistical* response time bounds.

## III. MODELING THE DEPLOYMENT PROCESS

In this section, we discuss how to model deployment processes that involve a certain degree of uncertainty. We abstract the actual deployment process to the determination of the sensor nodes' locations. Together with routing this directly results in uncertainty about the topology of the WSN.

The conceivable deployment processes, e.g. by airplane or human operators, are usually modeled by spatial point processes. In the following two subsections we introduce deployment models with different assumptions on the degree of structure and randomness of this point process.

### A. Uniform Random Deployment

Deployment processes with a high amount of uncertainty can be most easily modeled by a uniform random placement of the sensor nodes over the sensor field. This is essentially equivalent to a spatial Poisson process, where however the number of random points in the plane is set deterministically.

Specifically, for each sensor node both coordinates are generated independently by a random number generator delivering uniformly distributed values.

### B. Statistically Disturbed Grid Deployment

Uniform random deployment models a completely random deployment process, but there may often be some *structure*. We want to reflect this by a *partial* random deployment model. In our case, the deployment process strives to achieve a grid layout of the sensor nodes as this results in a number of desirable characteristics, e.g., with respect to coverage [10]. However, we assume a perfect grid layout not to be achievable or resulting in a too costly deployment process and thus model a grid deployment approximation called *statistically disturbed grid* which is similar to the approach used in [11].

**Definition 4.** (Statistically Disturbed Grid) Given a set  $\mathcal{L}$  of locations generated by overlaying a regular  $n * n$  grid on the sensor field  $\mathbb{F}$ , we define for each  $l \in \mathcal{L}$  an area  $D_l$  of possible locations for sensor node  $l$  by

$$D_l := \{l' \in \mathbb{F} : d(l, l') < r_{disturbance}\},$$

where  $d(a, b)$  denotes the Euclidean distance between positions  $a$  and  $b$  and  $r_{disturbance}$  denotes the so-called disturbance radius. A *statistically disturbed grid* is now generated by choosing for each sensor node  $l$  a position from  $D_l$  according to a uniform random distribution.

With the disturbance radius, we can control how accurate a regular grid layout ( $r_{disturbance} = 0$ ) is approximated.

## IV. COMPUTING STATISTICAL RESPONSE TIME BOUNDS

In this section, we introduce our approach to compute statistical bounds on maximum response times in WSNs with random deployment. Basically, we propose a Monte Carlo method for the estimation of quantiles of the maximum response time. The maximum response time of a given topology can simply be calculated using SNC as detailed in Section II. However, the actual topology is the result of a random process

which lets the deployment become a random variable itself. Relating the known distribution function for the random vector of node positions (e.g., a uniform random distribution) to a distribution function for the topology is next to impossible. Thus, we resort to randomly sampling the probability space of possible topologies by implementing a Monte Carlo simulation [12]:

- 1) *Define a domain of possible inputs.*  
In our case, the domain is the set of possible sink-trees.
- 2) *Generate inputs randomly from the domain using a certain specified probability distribution.*  
The probability distribution of the sink-tree topologies depends on the deployment process. In any case, generating node distributions randomly and applying a routing scheme leads to a set of random sink-tree topologies.
- 3) *Perform a deterministic computation using the inputs.*  
In this step, we have to work out the maximum response time for each topology generated in Step 2. This can be done based on the deterministic calculations using SNC.
- 4) *Aggregate the results of the individual computations into the final result.*

Using order statistics we estimate different quantiles of the maximum response time distribution over the probability space of random topologies as induced by the respective random node distribution. The precision of this estimation is calculated based on a Chernoff bound approach as presented in [13].

Results from Monte Carlo methods can always be improved by generating more random samples in Step 2. Yet, this has to be weighted against the additional computational effort (mainly incurred in Step 3). Based on a prescribed precision for the quantile estimator (Step 4) we can work out the required number of random topologies to be generated beforehand. Some of the mathematical details are given in the next subsection.

## V. EXPERIMENTS

### A. Experimental Setup

In this section, we specify the assumptions and actual parameter settings for the numerical experiments we perform to investigate statistical response time bounds. We try to choose the parameters based on realistic values for sensor nodes, traffic demands, and deployment model parameters. Since some of the parameters may have an important effect on our bounds we treat them as primary factors and evaluate them for several levels. As a benchmark for our statistical bounds we select the worst-case topology based approach from [3].

1) *Sensor Node and Traffic Characteristics:* The sensors are modeled to be identical and consisting of an IEEE 802.15.4-compliant transceiver as well as a sensing unit that periodically generates some measurement data.

To obtain quantifiable ratios between worst-case response times of independently generated topologies we set the parameters in such a way that sensor network calculus' PMOO analysis is still able to compute *finite* bounds. This is a non-trivial task as there is uncertainty about the topology and

therefore the load can only be estimated in advance. The main factors are the number of sensors  $n$ , the sensors' service curve  $\beta_{R,T}$  and the arrival curve  $\alpha_{r,b}$ . Let  $n'$  denote the number of flows crossing a node, then the PMOO delay bound is finite iff  $R \geq n' * r$ , i.e., every node in one hop distance to the sink is able to forward all data flows from the subtree below it.

The arrival rate  $r$  can be kept small by efficient encoding. Accordingly we chose  $\alpha_{r,b} = \alpha_{280[\frac{b}{s}], 280[b]}$ , leaving at least 4 bytes of the IEEE 802.15.4 frame for a sensor's measurements.

The transmission rate  $t_{tx}$  of IEEE 802.15.4 is  $r_{tx} = 250 \frac{kb}{s}$ . Yet, we want to take duty cycling, i.e., the ratio  $\rho$  of periods of operation divided by all periods, into account. The service rate is thus reduced to  $R = r_{tx} \times \rho$ . The investigated levels of  $\rho$  in our experiments are  $\rho = 0.25$  and  $\rho = 1$ , i.e., there is no duty cycling. In the former case  $R = 62.5 \frac{kb}{s}$ , which yields a maximal subtree size of  $\lfloor \frac{62500}{280} \rfloor = 223$  nodes.

2) *Network Characteristics:* The deployment area is modeled as a unit square. In order to enforce finite response time bounds with high probability, we chose the uniform random deployment and statistically disturbed grid deployments with  $r_{disturbance} \in \{0.01, 0.03, 0.05\}$ . These deployment models, greedy perimeter stateless routing, and a transmission range of 0.15 result in maximum distances of 5 to 6 hops to the central sink. We chose to deploy  $n = 30^2$  sensors as three models are grid approximations and a deployment of more than 1000 nodes leads to subtrees exceeding the maximal size.

3) *Precision of the Statistical Bounds:* The number of topologies to generate is defined by the desired precision for the statistical response time bounds. As we move closer to the actual worst-case topology, a high precision becomes more costly. So, the maximum number of topologies to generate is determined by the maximum quantile. We decided to estimate a maximum quantile of  $p = 0.99$ . Referring to the Chernoff bound approach in [13], we set the values of  $\delta = 0.01$  and  $\epsilon = 0.01$ , such that for all quantile estimates of the maximum response time bounds  $\hat{r}_p$  it applies that

$$P(\hat{r}_p \in [r_{0.99p}, r_{1.01p}]) \geq 0.99,$$

or, in particular, for  $p = 0.99$

$$P(\hat{r}_{0.99} \in [r_{0.9801}, r_{0.9999}]) \geq 0.99.$$

This results in the necessity of

$$N \geq \left\lceil \frac{2(1 + 2 \times 0.01)}{0.99 \times 0.01^2} \ln \left( \frac{2}{0.01} \right) \right\rceil = 109178$$

generated topologies for each configuration of parameters. Thus, we chose to set  $N = 110,000$  in all our experiments.

4) *Worst-Case Response Time Bounds:* To benchmark the results from our method of computing statistical bounds, we compare them against worst-case bounds derived based on [3]. To that end, we require an  $(o, d)$ -constraint on the maximum subtree of the WSN. Under our previous assumptions, there is in fact no better bound for  $d$  than the number of sensor nodes  $n$ . As this would lead to infinite delay bounds for all our investigated scenarios, we needed a more competitive alternative. Therefore, we extracted the maximum subtree size

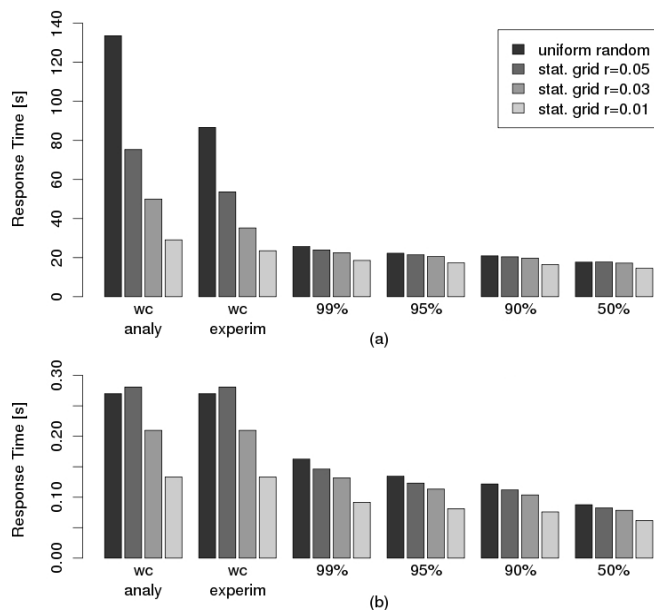


Figure 1. Response time for  $30^2$  sensors and cycle  $\rho = 0.25$  (a),  $\rho = 1$  (b)

$n'$  and the resulting depth  $d$  from each generated topology set and used these values together with  $o = \infty$  to invoke Definition 2 from [3] (see Section II) in order to calculate the worst-case response time bound.

## B. Numerical Results

In this section, we present the numerical results of our experiments. In particular, we compare our statistical response time bounds with deterministic worst-case bounds. Thereby, we analyze the effect of some parameters on the achievable bounds like the degree of the deployment process randomness.

1) *Quality of the Bounds:* In Figure 1, we show the analytical worst-case bound from [3] (*wc analy*), the worst response time encountered in the Monte Carlo simulation (*wc experim*), and several quantile estimates computed based on our Monte Carlo method. Besides the usually interesting high quantiles we also provided the 0.5-quantile, i.e., the median.

For a duty cycle of  $\rho = 0.25$ , the analytical worst-case, while not being totally unrealistic in terms of overshooting on observed response time bounds in simulations, can be considerably improved by statistical bounds. The actual gain depends on the deployment model; this is to be analyzed in the next subsection in more detail. Further, it can be observed that the 99%-quantile is only slightly higher than the median, which makes the statistical approach even more attractive as it indicates that the maximum response time distribution has a rather long, but light tail. By releasing some assurance on the response time bound we can bring the quantile pretty close to the average behaviour of the system (assuming that the median characterizes average behavior).

2) *Factor Analysis:* In this section, we further analyze the bounds' dependencies on some factors. While we performed an extensive analysis, we limit our presentation on the deployment model being used and the duty cycle of the nodes.

**Deployment Models.** By investigating different deployment models (see Section III) we analyze the effect of the degree of uncertainty about the topology. The results are depicted in Figure 1(a) for a duty cycle of 0.25. Clearly, the higher the degree of uncertainty, the higher the difference between worst-case and statistical response time bounds. In fact, only the worst-case bound is affected by the increasing uncertainty. As mentioned before, the statistical response time bound represented by the 99% quantile stays very close to the median of the maximum response time distribution under all deployment models, which is actually the best one could hope for.

**Duty Cycle.** To investigate the effect of duty cycling on the response time bounds we performed a set of experiments with no duty cycling, i.e.,  $\rho = 1$ . The results are shown in Figure 1(b). In contrast to Figure 1(a) which contains the results for equal settings except for  $\rho$  the delays are much lower. More interestingly, we can observe that the differences between worst-case and statistical response time bounds are no longer as pronounced as before. Apart from the almost deterministic deployment (stat. grid  $r = 0.01$ ), the deployments achieve more or less the same results. It is also interesting to note that the analytical worst case bounds are experienced under all deployment models. The reason for this behavior lies in the fact that without duty cycling the topological influence is mainly reduced to the size of the maximum subtree of the WSN, which provides for less variability of the respective maximum response time bound distribution.

## REFERENCES

- [1] J. B. Schmitt, F. A. Zdarsky, and L. Thiele, "A Comprehensive Worst-Case Calculus for Wireless Sensor Networks with In-Network Processing," in *IEEE Real-Time Systems Symposium (RTSS)*, Dec. 2007.
- [2] J. Schmitt and U. Roedig, "Sensor network calculus - a framework for worst case analysis," in *Proc. Distributed Computing on Sensor Systems (DCOSS)*, pp. 141–154, June 2005.
- [3] J. Schmitt and U. Roedig, "Worst Case Dimensioning of Wireless Sensor Networks under Uncertain Topologies," in *Proc. of WiOpt, Workshop on Resource Allocation in Wireless Networks*, 2005.
- [4] J.-Y. Le Boudec and P. Thiran, *Network Calculus A Theory of Deterministic Queuing Systems for the Internet*. No. 2050 in Lecture Notes in Computer Science, Berlin, Germany: Springer-Verlag, 2001.
- [5] A. Koubaa, M. Alves, and E. Tovar, "Modeling and Worst-Case Dimensioning of Cluster-Tree Wireless Sensor Networks," in *IEEE International Real-Time Systems Symposium (RTSS)*, 2006.
- [6] H. She, Z. Lu, A. Jantsch, L.-R. Zheng, and D. Zhou, "Traffic splitting with network calculus for mesh sensor networks," in *Proceedings of the Future Generation Communication and Networking (FGCN'07)*, 2007.
- [7] P. Suriyachai, U. Roedig, and A. Scott, "Implementation of a Deterministic Wireless Sensor Network," in *Proc of the 5th IEEE European Workshop on Wireless Sensor Networks (EWSN)*, 2008.
- [8] J. B. Schmitt, F. A. Zdarsky, and I. Martinovic, "Performance Bounds in Feed-Forward Networks under Blind Multiplexing," Technical Report 349/06, University of Kaiserslautern, Germany, Apr. 2006.
- [9] J. B. Schmitt, F. A. Zdarsky, and M. Fidler, "Delay bounds under arbitrary aggregate multiplexing: When network calculus leaves you in the lurch..." in *Proceedings of the IEEE INFOCOM*, Apr. 2008.
- [10] S. Kumar, T. H. Lai, and J. Balogh, "On k-coverage in a mostly sleeping sensor network," in *ACM MobiCom*, (New York, NY, USA), 2004.
- [11] J. Robinson and E. W. Knightly, "A performance study of deployment factors in wireless mesh networks," in *IEEE INFOCOM*, 2007.
- [12] N. Metropolis and S. Ulam, "The monte carlo method," in *Journal of the American Statistical Association* 44 (247): 335-341, 1949.
- [13] M. Huber, "Notes on median and quantile estimation with applications," 1998.