Received May 14, 2021, accepted July 6, 2021, date of publication July 26, 2021, date of current version August 10, 2021.
Digital Object Identifier 10.1109/ACCESS.2021.3099637

On the Deadline Miss Probability of Various Routing Policies in Wireless Sensor Networks

MATÍAS SEPÚLVEDA[1], CHRISTIAN OBERLI[1], (Member, IEEE), BENJAMIN BECKER[2], AND PATRICK LIESER[2]
[1]Department of Electrical Engineering, Pontificia Universidad Católica de Chile, Macul, Santiago 7820436, Chile
[2]Multimedia Communications Laboratory, Technische Universität Darmstadt, 64283 Darmstadt, Germany

Corresponding author: Matías Sepúlveda (mesepulveda@uc.cl)

This work was supported in part in Chile by the Research Center for Integrated Disaster Risk Management (CIGIDEN) under Grant ANID/FONDAP/15110017, in part by the German Research Foundation (DFG) within Project B1 of the Collaborative Research Center (CRC) and through the Mercator Fellowship Program under Grant 1053 MAKI, and in part by the LOEWE Initiative, Hessen, Germany, within the emergenCITY Center. The work of Matías Sepúlveda was supported by the Scholarship from Magíster Nacional through the Program of the Chilean National Agency for Research and Development (ANID) under Grant 2020–22201551.

ABSTRACT Moving data across communication networks is often subject to deadline requirements. An example is early warning of disasters of natural origin, where sensor measurements at the disaster location must be communicated across a network within a predefined maximum delay in order for a consequent warning to be timely. In this work, we present a probabilistic model that allows for characterizing the delay experienced by sensor measurements in a wireless sensor network from source to sink depending upon the routing metric used for forwarding the data through the network. Using link delay probability distributions and the probabilities of following different paths to the sink, source-to-sink delay distributions are found for routing policies based on minimum hop-count, minimum mean delay and the Joint Latency (JLAT) protocol. An algorithm for calculating the end-to-end source to sink delay probability density function (PDF) is presented for the general case of networks that use routing tables whose input for routing decisions is the remaining time-to-deadline. The work provides a general tool for routing delay analysis, allowing for comparison of the deadline miss probability between different routing policies. An improved form of JLAT is proposed. Its deadline miss probability is found using the presented algorithm and compared to the ones determined for minimum hop-count, minimum mean delay and JLAT by means of an example.

INDEX TERMS Mesh networks, probability density function, real time systems, routing protocols, wireless sensor networks.

I. INTRODUCTION

Consider a wireless sensor network (WSN) with nodes connected by edges. All nodes may be sources of data, which is relayed from node to node in multihop fashion along the edges toward the desired destination nodes [1]–[5]. Any node may be a destination node, but for simplicity and without loss of generality we will focus on delivering the data to one specific *sink node*. Regardless of its source node, all data must reach the sink by a deadline. We are interested in the probability of missing the deadline.

It is evident that the deadline miss probability (DMP) depends on whichever policy is used for taking routing decisions. These decisions are often based on the value of some *routing metric*. We will focus on the role played by routing metrics on the DMP.

In order to motivate our work, the sequel provides an example from the realm of wireless sensor networks. We will use this case throughout the article in order to frame the work within a concrete context and use case, but the results are applicable to other contexts as well, including logistics and transportation networks and digital communication networks in general [6]–[8].

A time-critical application of WSN is early warning of flash floods. To this end, the authors have been operating a WSN for early warning of flash floods since 2014 [4], [5]. The network is composed of 17 stations located in the Quebrada de Ramón (QR) basin at the foothills of Santiago, Chile, at elevations between 878 and 2962 meters above sea level. The sensor nodes measure various hydro-meteorological

The associate editor coordinating the review of this manuscript and approving it for publication was Zihuai Lin.

variables every 10 minutes. The measurements are relayed over the network in multiple hops using the Sensorscope communication protocol stack [2], [3], [9] to a sink node with cellular access to the Internet, which uploads the data to a server in the Cloud. In the QR network, measurements must reach the sink within 30 minutes to be useful for early warning.

Seeking the goal of minimizing the end-to-end delay from source to sink *on average* does not ensure that the probability of meeting a deadline is maximized. In effect, a path P_1 across the network may have a smaller expected delay than a path P_2, but a larger delay variance (Fig. 1, recreated from [10]). This can cause that the probability of missing the deadline is larger for P_1 than for P_2 even though the expected delay of P_1 is smaller. From this example, it is compelling to consider the probability density functions of paths delays (Fig. 1) for routing decisions, rather than statistical parameters about them, such as the mean value of those distributions.

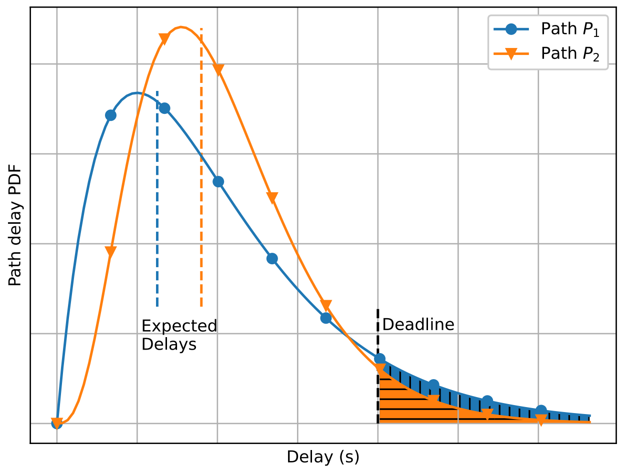

FIGURE 1. The probability of missing the deadline is larger for path P_1 than for path P_2 (shaded areas) even though the expected delay of P_1 is smaller (dashed lines). Recreated from [10].

In this paper we develop a method that allows for determining the probability distributions of the end-to-end delays in WSNs. The method is valid for any routing policy in which the next hop is determined based on the time en route of the data to be forwarded. (Fixed routing tables that ignore the time en route are special cases for which the method is also valid.) The technique builds on a mathematical model we develop for the probability distribution of the time en route of data as it journeys from its source node towards the sink. The model is then used in an iterative algorithm for determining the end-to-end delay probability distribution seen from a node to the sink, when data is forwarded on each hop by minimizing the probability of missing a deadline given the time en route spent already up to the current hop. Further, the deadline miss probabilities of routing by minimum hop-count, minimum mean delay and the Joint Latency (JLAT) protocol are derived for comparison. An example is provided as well in order to illustrate the use of the algorithm and for showing the differences between the deadline miss probabilities of the four approaches.

This paper is organized as follows: Section II presents a general model for calculating the delay probability distribution functions of paths and the DMP from a given node to the sink. In Section III the DMP of the routing metrics min-hop, minimum mean delay and JLAT is analyzed. An algorithm for calculating the delay probability distribution functions of routing approaches based on deadline-aware routing tables is presented in Section IV. Finally, Section V presents the conclusions of our work.

II. GENERAL MATHEMATICAL MODEL

Our aim is to study the probability of achieving or missing a deadline by data moving from sensor nodes to a sink node in a WSN. By *data value* or *measurement* we mean an indivisible data unit composed by a *sensor ID*, a *source node ID*, a *measured value* and a *measurement timestamp*.

Once a measurement is acquired by a sensor node, it begins a multi-hop journey across the network towards the sink node. Along the way it encounters numerous delays, caused by transmission queues, channel access delays, propagation delays, protocol stack processing delays, etc. [11]. These delays repeat for each new hop the data takes along its route toward the sink. One-hop delays add up over time to a *time en route* value and subtract time left until the deadline (*time-to-deadline*). The *deadline* itself is the maximum time that a data value acquired by a sensor node can take to reach the sink in order to be useful for an application (e.g. an early warning application) [12].

For each hop, individual or aggregated measurements are composed by the network layer into *packets*, which are then passed down to the node's MAC (medium access control) layer and transmitted over the physical medium. We assume that the network layer has, if needed, the capability of inspecting the measurement timestamp of each data unit carried by the packets in order to assist routing decisions for each measurement. Thus, upon downstream reception of a packet, the receiving network layer may extract the measurements carried by the packet and insert them into separate packets that will be forwarded to differing neighbours, depending on the time left for each measurement to fulfill its deadline.

The end-to-end delay realization for each measurement to reach the sink depends on the route followed and hence on the metrics used to make the forwarding decisions. A priori, the delay from node u to the sink is a random variable which we denote $D_N^{(u)}$ (subscript N denotes delays seen from nodes to the sink when the route is unknown a priori). The probability of missing a deadline, henceforth called DMP (Deadline Miss Probability), for a given time-to-deadline δ and source node u is then

$$\mathrm{DMP}_N^{(u)}(\delta) = \mathbb{P}\left\{D_N^{(u)} > \delta\right\} = \int_{\delta}^{\infty} f_N^{(u)}(d)\,\mathrm{d}d, \qquad (1)$$

where $f_N^{(u)}(d)$ is the probability density function (PDF) of $D_N^{(u)}$.

It is to be noted that (1) is related to the metric Deadline Miss Ratio (DMR) for node u discussed e.g. in [13]–[15]. DMR is an estimator of the DMP. It is also worth noting that DMR is related to the Measurement Delivery Ratio (MDR) studied in [4]. MDR and DMR are statistical complements, although DMR is measured at the network layer, whereas MDR is empyrical and includes statistics about sensors and hardware failure.

The random properties of $D_N^{(u)}$ depend on a number of aspects. In order to study $D_N^{(u)}$, we turn our attention to a few concepts from graph theory in the sequel.

We define a *path* as a sequence of edges that connect a node u with the sink node s. We assume that every node has at least one path to the sink and that all nodes on a path are distinct. Paths with cycles are therefore ruled out. Furthermore, we assume that edges may be bi-directional. Therefore, some paths may follow links in opposite direction than other paths. By $\mathcal{P}_u$ we denote the set of all paths from node u to the sink. $P^{(k,u)} \in \mathcal{P}_u$ is the k-th path from u to the sink, enumerated by k in no particular order.

Paths are sequences of links. Hence, each path $P^{(k,u)}$ can be said to have an associated set $\mathcal{E}^{(k,u)}$ of edges between pairs of nodes (v_1, v_2) that compose the path. Traffic across link (u, v) experiences a random delay $D_L^{(u,v)}$, whose PDF is denoted by $f_L^{(u,v)}(d)$ (the subscript L denotes delays across single links). We define $D_L^{(u,v)}$ as the time elapsed from the moment of reception of a measurement within a packet by node u, until the time at which that measurement, possibly carried by a different packet, is acknowledged to have been successfully received by node v one hop downstreams.

Several works have been done to capture the link statistics, either by measurement [11], [16], [17], simulation [18]–[20] or analytical models [11], [20]–[26].

We assume that the network topology and link PDFs are known and static. We also assume that link delays are independent random variables.

The end-to-end delay for path $P^{(k,u)}$ is given by

$$D_P^{(k,u)} = \sum_{(v_1,v_2)\in\mathcal{E}^{(k,u)}} D_L^{(v_1,v_2)}. \tag{2}$$

(Keeping with the notation, subscript P denotes delays related to paths.) Because link delays are assumed independent, the path delay PDFs can be obtained as [12], [27]

$$f_P^{(k,u)}(d) = \left(f_L^{(u,v_1)} * f_L^{(v_1,v_2)} * \cdots * f_L^{(v_n,s)}\right)(d), \tag{3}$$

where $*$ denotes the convolution operation of functions. We can see that the delay PDF of a path can be determined from all link PDFs that form the path.

Node PDFs introduced in (1) and path PDFs introduced in (3) are related but not the same. For a source node u, the former represents the combined a priori node-to-sink delay considering that any path in $\mathcal{P}_n$ can be taken, while the latter is the specific delay PDF of path k. However, it is to be pointed out that we may speak of a DMP of a path just as much it was defined in (1) for nodes, as follows:

$$\mathrm{DMP}_P^{(k,u)}(\delta) = \int_\delta^\infty f_P^{(k,u)}(d)\,\mathrm{d}d, \tag{4}$$

with $f_P^{(k,u)}(d)$ given by (3).

The notation introduced above is summarized in Table 1 (some notations in the table will be introduced later).

TABLE 1. Notation used throughout this article.

u, v, w	Denote nodes.
s	Denotes the sink node.
n, m	Integer index that enumerates nodes.
k	Integer index that enumerates paths.
d	A delay in seconds.
$\mathcal{P}_u$	Set of all paths between a node u and the sink.
$P^{(k,u)}$	k-th path from node u to sink.
$\mathcal{E}^{(k,u)}$	Set of edges that form path $P^{(k,u)}$.
$P^{(\hat{k},u)}$	Chosen path from node u to sink (by some criterion).
$(u,\ldots,v_n)$	Route starting at u and ending at v_n.
$D_L^{(u,v)}$	Delay of link (u,v), with PDF $f_L^{(u,v)}(d)$.
$D_P^{(k,u)}$	Delay of path $P^{(k,u)}$, with PDF $f_P^{(k,u)}(d)$.
$D_N^{(u)}$	Delay of node u to sink, with PDF $f_N^{(u)}(d)$.
$D_R^{(u,\ldots,v_n)}$	Time en route of route $(u,\ldots,v_n)$, with PDF $f_R^{(u,\ldots,v_n)}(d)$.
$f_L^{(u,v)}(d)$	PDF of the link delay $D_L^{(u,v)}$.
$f_P^{(k,u)}(d)$	PDF of the path delay $D_P^{(k,u)}$.
$f_N^{(u)}(d)$	PDF of the node delay $D_N^{(u)}$.
$f_R^{(u,\ldots,v_n)}(d)$	PDF of time en route $D_R^{(u,\ldots,v_n)}$.
$f_{R\|E}^{(u,\ldots,v_n)}(d\|E)$	Conditional PDF of time en route given event E.
$\mathrm{DMP}_N^{(u)}(d)$	Deadline Miss Probability of node u.
$\mathrm{DMP}_P^{(k,u)}$	Deadline Miss Probability of path $P^{(k,u)}$.
δ	Time-to-deadline.
$\mathrm{NH}_{v_n}(\delta)$	Next hop from node v_n for time-to-deadline δ.
M_{v_n}	Number of entries in the routing table of node v_n.
Δ_{v_n}	Random variable of the time-to-deadline at node v_n.
δ_{app}	Application deadline that measurements must meet.

The concepts introduced above are used in the next section for analysing the deadline miss probability of various routing metrics used in communications.

III. DEADLINE MISS PROBABILITY OF KNOWN ROUTING PROTOCOLS

Routing protocols for WSNs have been proposed in quite vast number and variety [28]–[30]. Routing aims at determining to which node data should be forwarded next and which path to the sink shall be followed. Routing decisions are generally taken based on some metric, such as min-hop [2], [3], [9], minimum mean delay and Joint Latency (JLAT) [12]. We next analyze the DMP performance of these routing approaches using the model introduced in the previous section.

A. MINIMUM HOP-COUNT (MIN-HOP)

This metric seeks to route by minimizing the number of hops used to reach the sink. The optimal path $P^{(\hat{k},u)}$ for a node u is typically chosen at random (or by any other suitable decision criterion) among the set of all values of k that correspond to paths in $\mathcal{P}_u$ with minimum hop-count, as follows:

$$P^{(\hat{k},u)} : \hat{k} \in \underset{k}{\operatorname{argmin}} \left| \mathcal{E}^{(k,u)} \right|, \tag{5}$$

where $|\cdot|$ denotes the cardinality of a set.

For this routing metric, the chosen path depends only on the topology, and remains the same for as long as the topology does not change. Therefore, the delay PDF for a node u is:

$$f_N^{(u)}(d) = f_P^{(\hat{k},u)}(d), \tag{6}$$

where $f_P^{(\hat{k},u)}(d)$ is the PDF of the delay that corresponds to the optimal path $P^{(\hat{k},u)}$ chosen by (5). The DMP is given by (1) and (6).

B. MINIMUM MEAN DELAY

Another interesting metric to analyze is the minimum mean delay (MMD). Routing by this metric seeks to minimize the mean delay of the measurements and corresponds to the example given in Fig. 1. Mathematically, the path chosen by node u using MMD is given by:

$$\begin{aligned} P^{(\hat{k},u)} : \hat{k} &= \underset{k}{\operatorname{argmin}} \, \mathbb{E}\left\{D_P^{(k,u)}\right\} \\ &= \underset{k}{\operatorname{argmin}} \sum_{(v_1,v_2)\in\mathcal{E}^{(k,u)}} \mathbb{E}\left\{D_L^{(v_1,v_2)}\right\}, \end{aligned} \tag{7}$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation operator.

As with min-hop, the chosen path by MMD is fully defined by link statistics and the topology. Therefore, the DMP of node u for a deadline δ is again given by (1) and (6).

MMD may be impractical for implementation. It is worth mentioning to this end that a practical routing metric with similarities to the MMD metric is the *expected transmission count* metric (ETX) [31]. ETX is a statistic kept by every node about the average number of transmission attempts (including re-transmissions) over every link to its neighbours until a reception acknowledgment is received. For routing, the ETX metrics are accumulated link-by-link by flooding from the sink node into the network. Each node compares the cumulative ETX metrics received from its neighbours and determines the path to the sink by choosing the neighbour with smallest cumulative ETX.

C. JOINT LATENCY (JLAT) PROBABILITY ROUTING

The JLAT algorithm [12] uses discrete versions of the $f_P^{(k,u)}(d)$ densities and accumulates their probability up to an *application deadline* (system parameter δ_{app}) in order to calculate, for every path, the probability of meeting the application deadline, as follows (JLAT metric):

$$C_P^{(k,u)} = \sum_{d=1}^{\delta_{\text{app}}} f_P^{(k,u)}[d], \tag{8}$$

where δ_{app} and d in [12] are specified in discrete time. Each node u then routes its measurements along the path with maximum $C_P^{(k,u)}$. In terms of our continuous-time notation, the path chosen by a node u given an application deadline δ_{app} in seconds is determined as

$$P^{(\hat{k},u)}(\delta_{\text{app}}) : \hat{k} = \underset{k}{\operatorname{argmin}} \, \mathrm{DMP}_P^{(k,u)}(\delta_{\text{app}}), \tag{9}$$

where $\mathrm{DMP}_P^{(k,u)}(\delta_{\text{app}})$ is given by (4).

The path chosen using the JLAT metric is established at the source node and stays fixed from there on. Therefore, the node PDF (cf. (1)) can again be obtained using (6). We point out, however, that in contrast to routing by min-hop or MMD, it is apparent from (9) that the optimal path $\hat{k}$ now depends on the specified application deadline δ_{app}. As the optimal path changes with δ_{app}, so does the corresponding path delay PDF $f_P^{(\hat{k},u)}(d)$ from (3). The DMP is, however, still given by (1), which in this case may also be expressed as:

$$\mathrm{DMP}_N^{(u)}(\delta_{\text{app}}) = \min_k \mathrm{DMP}_P^{(k,u)}(\delta_{\text{app}}). \tag{10}$$

One challenging aspect of routing with JLAT is that every node must know the path PDFs $f_P^{(k,u)}(d)$ of all possible paths to the sink, which by (3) implies also knowing many —if not all $f_L^{(u,v)}(d)$— densities of the network. This is challenging in practice and possibly unfeasible for networks with more than a few nodes.

D. JLAT WITH UPDATES

In practice, as a measurement hops across a network and time goes by, the time-to-deadline shortens and the optimal JLAT path that each relay node would pick may be different than the path chosen at the beginning of the journey by the source node. It is evident that updating the optimum path at each hop with a new JLAT calculation based on the remaining time-to-deadline instead of δ_{app} is an idea that should be considered. It is clear that the DMP of this *JLAT with updates* approach must be smaller or equal than that of the standard JLAT method.

In JLAT with updates, the path that a measurement eventually follows is unknown a priori and the route finally followed (a posteriori) may actually contain cycles. This is so because the optimum next hop chosen by each node visited is based on the remaining time-to-deadline and it may, under given circumstances, be most likely to meet the deadline if the data is sent back to an already visited node. As a consequence, a node's delay statistics $f_N^{(u)}(d)$ do not correspond anymore to those of a specific path, as was the case so far with the metrics considered. Expression (6) therefore no longer applies. A methodology for calculating $f_N^{(u)}(d)$ under these conditions is presented in the following section.

IV. DEADLINE MISS PROBABILITY OF PROTOCOLS WITH DEADLINE-BASED ROUTING TABLES

A. ROUTING TABLES AND ROUTES

Consider a network in which each node makes forwarding decisions based on a deadline-aware routing table. Input to the table is the time-to-deadline of the measurement that is to be forwarded and the table output is the next-hop neighbour.

Concretely, the routing table of a node v is a collection of entries as follows:

$$\mathrm{NH}_v(\delta) = \begin{cases} w_1, & \text{if } \delta \in [\delta^1_{v,w_1}, \delta^2_{v,w_1}) \\ \vdots & \vdots \\ w_m, & \text{if } \delta \in [\delta^1_{v,w_m}, \delta^2_{v,w_m}) \\ \vdots & \vdots \\ w_{M_v}, & \text{if } \delta \in [\delta^1_{v,w_{M_v}}, \delta^2_{v,w_{M_v}}). \end{cases} \quad (11)$$

Above, when the time-to-deadline δ is in the range $[\delta^1_{v,w_m}, \delta^2_{v,w_m})$, then node w_m will be chosen by node v as the next-hop (NH). M_v is the number of entries in the routing table of node v.

The routing tables (11) may be constructed by any suitable means, for instance, using (9) and varying δ_{app} over the entire range of possible time-to-deadline values, thus finding the optimum path and next hop (routing table (11)) for every time-to-deadline δ.

By the above routing policy, each new hop of any given measurement depends on the link delays it experienced on all previous hops. Every measurement therefore follows an individual trajectory through the network. We call these trajectories *routes* and they grow with each new hop. Each time a hop is added to the tail of a route, then the original route is defined as the parent route and the new one as the child route. For example, route (u, v_1) is the parent route of route (u, v_1, v_2).

The *time en route* is the time elapsed from the moment a measurement was taken at its source node until the time the data packet carrying the measurement reaches the end node of a given route.

The time en route is a random variable. For a route $(u, v_1, \ldots, v_n)$, we represent it by $D_R^{(u,v_1,\ldots,v_n)}$ and its PDF is $f_R^{(u,v_1,\ldots,v_n)}(d)$ (subindex R denotes delays related to routes).

The *time-to-deadline* for a data unit to reach the desired destination (typically the sink) when the data is at the last node of a route, v_n, is given by:

$$\Delta_{v_n} = \delta_{\text{app}} - D_R^{(u,v_1,\ldots,v_n)}. \quad (12)$$

The time-to-deadline is also a random variable. At any given node v_n, the realizations δ_{v_n} of Δ_{v_n}, observed by node v_n each time a new measurement passes it, are used as input to the node's routing table in order to determine the next-hop node.

The probability of following route $(u, v_1, \ldots, v_n, w)$ can be expressed in terms of the probability of following its parent route as follows:

$$\begin{aligned} &\mathbb{P}\{(u, \ldots, v_n, w)\} \\ &\quad = \mathbb{P}\{(u, \ldots, v_n, w) \cap (u, \ldots, v_n)\} \\ &\quad = \mathbb{P}\{(u, \ldots, v_n, w) | (u, \ldots, v_n)\} \cdot \mathbb{P}\{(u, \ldots, v_n)\} \\ &\quad = \mathbb{P}\{\mathrm{NH} = w | (u, \ldots, v_n)\} \cdot \mathbb{P}\{(u, \ldots, v_n)\}, \end{aligned} \quad (13)$$

where the symbol | denotes conditional probability and

$$\begin{aligned} &\mathbb{P}\{\mathrm{NH} = w | (u, \ldots, v_n)\} \\ &\quad = \mathbb{P}\left\{D_R^{(u,\ldots,v_n)} \in (\delta_{\text{app}} - \delta^2_{v_n,w}, \delta_{\text{app}} - \delta^1_{v_n,w}]\right\} \\ &\quad = \int_{\delta_{\text{app}} - \delta^2_{v_n,w}}^{\delta_{\text{app}} - \delta^1_{v_n,w}} f_R^{(u,\ldots,v_n)}(d)\, \mathrm{d}d, \end{aligned} \quad (14)$$

is the probability of selecting w as next-hop (NH) given that route $(u, v_1, \ldots, v_n)$ has been followed.

It is to be noted that min-hop, MMD and (standard) JLAT can be modeled as particular cases of this routing mechanism. In effect, since all of these routing methods determine and fix the path at the source node, the routing table at each node is simply such that the same fixed next-hop neighbour is returned regardless of the remaining time-to-deadline (in the case of JLAT, however, the routing tables vary with δ_{app}).

B. PDF OF THE TIME EN ROUTE

Consider starting at the source node u. Because the time en route at the source node is zero, the remaining time to deadline is equal to the application deadline (δ_{app}). The source node forwards its measurements according to its routing table (11) evaluated with $\delta = \delta_{\text{app}}$. Therefore, routing at the source node is always the same. The time en route at the first hop v_1, $D_R^{(u,v_1)}$, is therefore

$$D_R^{(u,v_1)} = D_L^{(u,v_1)}, \quad (15)$$

and has a PDF given by:

$$f_R^{(u,v_1)}(d) = f_L^{(u,v_1)}(d). \quad (16)$$

Consider now the second hop. Node v_1 inspects the time en route and calculates the realization δ_{v_1} of Δ_{v_1} using (12). Then, entering the routing table (11) with δ_{v_1}, the next hop node w is determined and the data is forwarded accordingly.

Using the law of total probability, $f_R^{(u,v_1)}(d)$ can be expressed in terms of conditional probabilities as:

$$\begin{aligned} f_R^{(u,v_1)}(d) = \sum_{m=1}^{M_{v_1}} [&f_{R|\mathrm{NH}}^{(u,v_1)}(d | \mathrm{NH} = w_m) \\ &\cdot \mathbb{P}\{\mathrm{NH} = w_m | (u, v_1)\}], \end{aligned} \quad (17)$$

where $f_{R|\mathrm{NH}}^{(u,v_1)}(d|\mathrm{NH} = w)$ is the conditional PDF of $D_R^{(u,v_1)}$ given that the next hop shall be w, that is, given that $\Delta_{v_1} \in [\delta^1_{v_1,w}, \delta^2_{v_1,w})$; and $\mathbb{P}\{\mathrm{NH} = w | (u, v_1)\}$ is given by (14).

The choice of a given next-hop node w using (11) implies that the time en route up to the current hop v_1 is, by (12), in the range

$$D_R^{(u,v_1)} \in (\delta_{\text{app}} - \delta^2_{v_1,w}, \delta_{\text{app}} - \delta^1_{v_1,w}]. \quad (18)$$

Furthermore, it is to be noted in (17) that because of (11) all conditions $\mathrm{NH} = w_m$ are disjoint events in m. For this reason, each addend in (17) provides the definition of $f_R^{(u,v_1)}(d)$ for the corresponding range given by (18). We may therefore express (17) in simpler fashion as

$$f_R^{(u,v_1)}(d) = f_{R|\mathrm{NH}}^{(u,v_1)}(d | \mathrm{NH} = w) \cdot \mathbb{P}\{\mathrm{NH} = w | (u, v_1)\}, \quad (19)$$

keeping in mind, however, that w varies with d. Focusing on hopping next to a specific node w and therefore on the range $d \in (\delta_{\text{app}} - \delta^2_{v_1,w}, \delta_{\text{app}} - \delta^1_{v_1,w}]$, we may solve for the conditional PDF in (19) and use (14) to obtain:

$$f^{(u,v_1)}_{R|\text{NH}}(d|\text{NH}=w) = \frac{f^{(u,v_1)}_R(d)}{\int_{\delta_{\text{app}}-\delta^2_{v_1,w}}^{\delta_{\text{app}}-\delta^1_{v_1,w}} f^{(u,v_1)}_R(d)\,\mathrm{d}d}. \quad (20)$$

The above PDF is non-zero only for the range of d indicated above, and 0 otherwise.

Using (20), the PDF of the time en route from source node u to a given node w is then obtained as

$$f^{(u,v_1,w)}_R(d) = f^{(u,v_1)}_{R|\text{NH}}(d|\text{NH}=w) * f^{(v_1,w)}_L(d). \quad (21)$$

We shall also point out that by (20), $f^{(u,v_1,w)(d)}_R$ in (21) depends solely on $f^{(u,v_1)}_R(d)$ and on $f^{(v_1,w)}_L(d)$.

The calculation of $f^{(u,v_1,w)}_R(d)$ in (21) must be performed for all neighbours w of v_1 to which v_1 could route with time-to-deadline values in the range $\Delta_{v_1} \in [0, \delta_{\text{app}}]$, according to the routing table (11).

In general, for the n-th hop node v_n of a parent route $(u, v_1, \ldots, v_n)$, the conditional PDF of time en route is

$$f^{(u,\ldots,v_n)}_{R|\text{NH}}(d|\text{NH}=w) = \frac{f^{(u,\ldots,v_n)}_R(d)}{\int_{\delta_{\text{app}}-\delta^2_{v_n,w}}^{\delta_{\text{app}}-\delta^1_{v_n,w}} f^{(u,\ldots,v_n)}_R(d)\,\mathrm{d}d}, \quad (22)$$

for $d \in (\delta_{\text{app}} - \delta^2_{v_n,w}, \delta_{\text{app}} - \delta^1_{v_n,w}]$ and 0 otherwise, and the time en route to a given next hop node w is:

$$f^{(u,\ldots,v_n,w)}_R(d) = f^{(u,\ldots,v_n)}_{R|\text{NH}}(d|\text{NH}=w) * f^{(v_n,w)}_L(d). \quad (23)$$

Once $f^{(u,\ldots,v_n,w)}_R(d)$ has been calculated, the child route $(u, v_1, \ldots, v_n, w)$ is turned into a new parent route by assigning $n \leftarrow n+1$ and $v_n \leftarrow w$ and the PDFs of the time en route for all possible hops following w can be determined using again (22) and (23).

The above procedure can be performed in tree-like fashion, following all possible routes from a source node u and determining the route delay PDF for each route. This notion is the base of an iterative algorithm for finding the end-to-end delay PDF $f^{(u)}_N(d)$ from any given source node u to the sink, described next.

C. ITERATIVE ALGORITHM FOR FINDING THE END-TO-END DELAY PDF $f^{(u)}_N(d)$

Starting from any node u, the steps described in the previous subsection can be followed to every neighbor of u (the first hop however is always the same, as noted), and again from each one of them to their neighbours. The corresponding route PDFs are calculated each time using (22) and (23). Hops back to a previously visited node can occur. This flooding procedure shall continue into the network, following all possible routes. Each time a hop is added to the tail of a route, the resulting new route is tested for its probability to reach the sink by the deadline. This is done by evaluating the DMP of the route, as follows:

$$\begin{aligned}\text{DMP}^{(u,\ldots,v_n)}_R(\delta_{\text{app}}) &= \mathbb{P}\left\{D^{(u,\ldots,v_n)}_R > \delta_{\text{app}}\right\} \\ &= \int_{\delta_{\text{app}}}^{\infty} f^{(u,\ldots,v_n)}_R(d)\,\mathrm{d}d, \end{aligned} \quad (24)$$

The test may yield two outcomes, namely:

1) The DMP of the route is 1, thus indicating that the sink cannot be reached anymore by the deadline along this route. In this case, this route is unviable and shall be discarded.
2) The sink is reached (with $\text{DMP}^{(u,\ldots,v_n)}_R(\delta_{\text{app}}) < 1$). In this case, the route followed is a viable route for reaching the sink by the deadline.

Once no more routes are left to be explored, the accumulated statistics of each route that reached the sink are weighted by the probability of taking that route, thus obtaining the end-to-end delay PDF as follows:

$$f^{(u)}_N(d) = \sum_{\forall (u,\ldots,s)} \mathbb{P}\{(u,\ldots,s)\} \cdot f^{(u,\ldots,s)}_R(d), \quad (25)$$

where $\mathbb{P}\{(u,\ldots,s)\}$ is given by (13).

It is to be noted that reaching condition 1 above may be impractical because the number of routes to be explored might be very large. Discarding routes with a criterion $\text{DMP}^{(u,\ldots,v_n)}_R(\delta) \geq 1-\epsilon$, with ϵ a small probability is a practical alternative. Larger values of ϵ prune routes sooner and shorten the search tree, but also tend to limit the accuracy of the result. This is so because (13) is not exact when $\epsilon \neq 0$, and looses accuracy when ϵ grows.

Algorithms 1 and 2 detail the procedure for calculating the end-to-end delay PDF $f^{(u)}_N(d)$, as described.

Algorithm 1 Calculation of End-to-End Delay PDF $f^{(u)}_N(d)$

Input: u: Source node,
δ_{app}: Application deadline.
Output: $f^{(u)}_N(d)$: End-to-end delay PDF of node u.
1: $v_1 = \text{NH}_u(\delta_{\text{app}}) \leftarrow$ From routing table in (11).
2: $f^{(u,v_1)}_R(d) = f^{(u,v_1)}_L(d) \leftarrow$ From (16).
3: VRL = [ ] $\leftarrow$ Create an empty Valid Routes List.
4: Call the Recursive Calculation (RC) of routes (Algorithm 2),
VRL = RC(v_1, $f^{(u,v_1)}_R(d)$, 1, VRL).
5: $f^{(u)}_N(d) \leftarrow$ From (25) with the entries of VRL.
6: **return** $f^{(u)}_N(d)$.

Once the end-to-end delay PDF $f^{(u)}_N(d)$ has been obtained with Algorithm 1, the DMP can be calculated using (1), with $\delta = \delta_{\text{app}}$. It is to be noted, however, that $f^{(u)}_N(d)$ returned by Algorithm 1 is only valid for the specific value of δ_{app} given to the algorithm as input. Therefore, calculating the DMP with (1) for other values of δ_{app} requires each time to determine the corresponding $f^{(u)}_N(d)$ with Algorithm 1 first.

Algorithm 2 RC: Recursive Calculation (RC) of the Valid Routes List (VRL)

Input: v_n: Last node of the route,
$f_R^{(u,\ldots,v_n)}(d)$: Time en route PDF,
$\mathbb{P}\{(u,\ldots,v_n)\}$: Probability of following the route,
VRL: Valid Routes List.
Output: VRL: Updated Valid Routes List.

```
if the probability of reaching the sink along the followed route by the deadline becomes negligible, i.e. ∫_0^{δ_app} f_R^{(u,...,v_n)}(d) dd ≤ ε, then
    return VRL.
else if v_n is the sink, i.e. v_n = s, then
    Add {P{(u,...,s)}, f_R^{(u,...,s)}(d)} to VRL.
    return VRL.
end if
{If none of the stopping conditions is met, Algorithm 2 is called for every possible next-hop node.}
for all w in routing table (11) for node v_n do
    Find δ^1_{v_n,w}, δ^2_{v_n,w}.
    P{NH = w|(u,...,v_n)} ← From (14).
    P{(u,...,v_n,w)} ← From (13).
    f_{R|NH}^{(u,...,v_n)}(d|NH = w) ← From (22).
    f_R^{(u,...,v_n,w)}(d) ← From (23).
    RC is called with node w,
        VRL = RC(w, f_R^{(u,...,v_n,w)}(d), P{(u,...,v_n,w)}, VRL).
end for
return VRL.
```

A numerical example is presented in the Appendix in order to illustrate the model and algorithm presented in this work.

V. CONCLUSION

In this work we analyze and model the probability distribution of the end-to-end delay in wireless sensor networks for data moving from source-to-sink with delay deadline requirements. Our work contributes an algorithm that allows for determining the end-to-end delay probability distribution for any routing policy that is based on routing tables whose input is the time-to-deadline of the transported data. With this delay distribution, the deadline miss probability of routing protocols can be calculated for deadline-limited networking applications.

APPENDIX. EXAMPLE

In the sequel we illustrate the use of the model and algorithm presented in this work by means of the example network shown in Fig. 2. The calculations and implementations of Algorithms 1 and 2 were performed using the Python programming language with its Math and Numpy libraries.

All nodes have four paths without cycles to reach the sink node s. Focusing arbitrarily on node 1, for each of its four paths, the path delays (2) are given by:

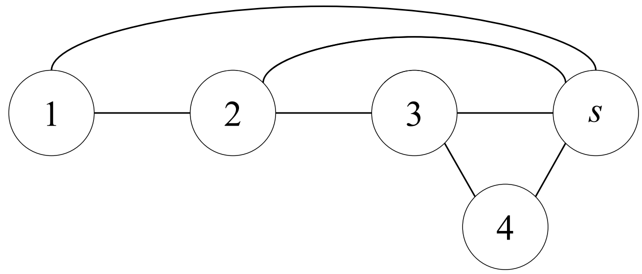

FIGURE 2. Example network.

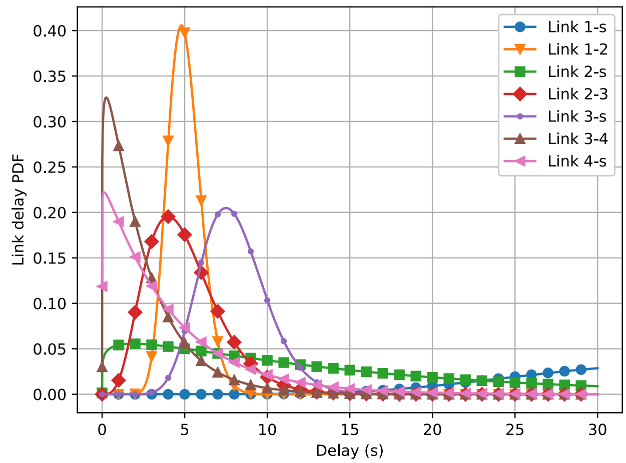

FIGURE 3. Link delay PDFs, $f_L^{(v_1,v_2)}(d)$, used in the example network.

$$D_P^{(1,1)} = D_L^{(1,s)} \tag{26}$$
$$D_P^{(2,1)} = D_L^{(1,2)} + D_L^{(2,s)} \tag{27}$$
$$D_P^{(3,1)} = D_L^{(1,2)} + D_L^{(2,3)} + D_L^{(3,s)} \tag{28}$$
$$D_P^{(4,1)} = D_L^{(1,2)} + D_L^{(2,3)} + D_L^{(3,4)} + D_L^{(4,s)}. \tag{29}$$

Knowing the PDF of each of the links, we can calculate the delay PDF of each path using (3). These are:

$$f_P^{(1,1)}(d) = f_L^{(1,s)}(d) \tag{30}$$
$$f_P^{(2,1)}(d) = \left(f_L^{(1,2)} * f_L^{(2,s)}\right)(d) \tag{31}$$
$$f_P^{(3,1)}(d) = \left(f_L^{(1,2)} * f_L^{(2,3)} * f_L^{(3,s)}\right)(d) \tag{32}$$
$$f_P^{(4,1)}(d) = \left(f_L^{(1,2)} * f_L^{(2,3)} * f_L^{(3,4)} * f_L^{(4,s)}\right)(d). \tag{33}$$

In [17], the authors made measurements of the link delays in a wireless network. They concluded that 90% of the times the best fit for the link PDF was obtained with a Gamma or Logistic distribution. For this reason, we will assume for our example that the links have delays $D_L^{(u,v)} \sim \text{Gamma}(\alpha_{uv}, \beta_{uv})$, where α_{uv} is a shape parameter and β_{uv} is a rate parameter (or inverse scale parameter), although our method can take PDFs of any nature. The values chosen arbitrarily for these parameters in this example are shown in Table 2. The corresponding PDFs are illustrated in Fig. 3.

The convolutions (31), (32) and (33) were evaluated numerically because there are no closed-form expressions for

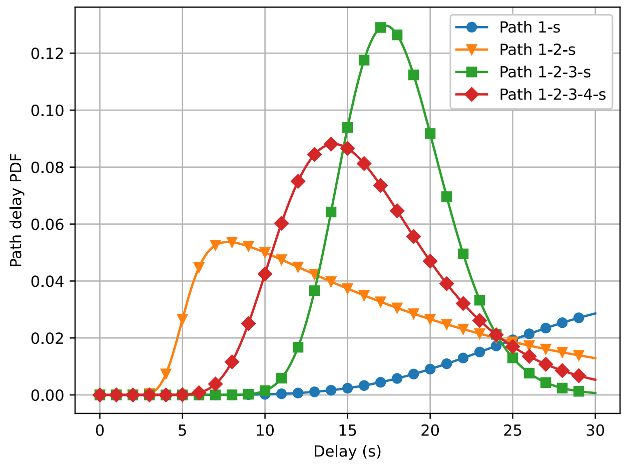

FIGURE 4. Path delay PDFs $f_P^{(k,u)}(d)$, given by (3), of the paths from node 1 to the sink.

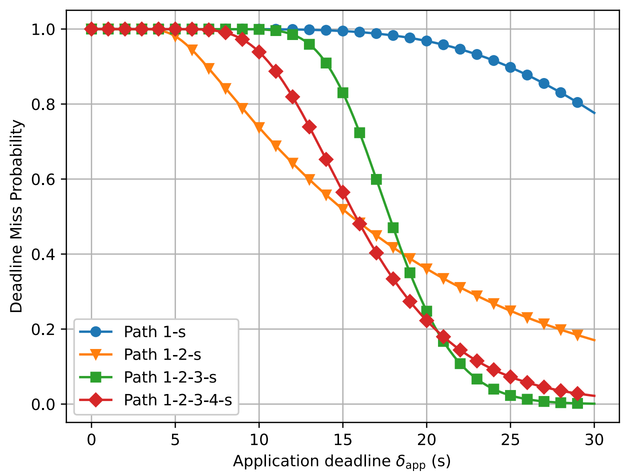

FIGURE 5. Deadline miss probability of the paths from node 1 to the sink, given by (4).

them. The resulting delay PDFs for all paths from node 1 to the sink obtained by (3) are shown in Fig. 4. The corresponding path DMPs obtained by (4) are shown in Fig. 5.

TABLE 2. Parameters used for the example network.

Parameter	α_{1s}	α_{12}	α_{2s}	α_{23}	α_{3s}	α_{34}	α_{4s}
Value	10	25	1.15	5	16	1.11	1.03

Parameter	β_{1s}	β_{12}	β_{2s}	β_{23}	β_{3s}	β_{34}	β_{4s}
Value	0.25	5	0.08	1	2	0.44	0.25

The path chosen by each node when JLAT routing is used depends on the application deadline δ_{app} and is given by (9). Figs. 6 through 9 present the optimum paths obtained for nodes 1 through 4 in the example, respectively, for values of δ_{app} between 0 s and 30 s. All path statistics were determined as described above for node 1.

In order to determine the delay PDF and DMP of the JLAT with updates routing method, the routing tables (11) must be known. For this example we determined them as explained in Section IV-A. Concretely, the next hop toward the sink at any particular node is specified by the first hop of the JLAT path that is optimum for the remaining time-to-deadline of the measurement to be forwarded

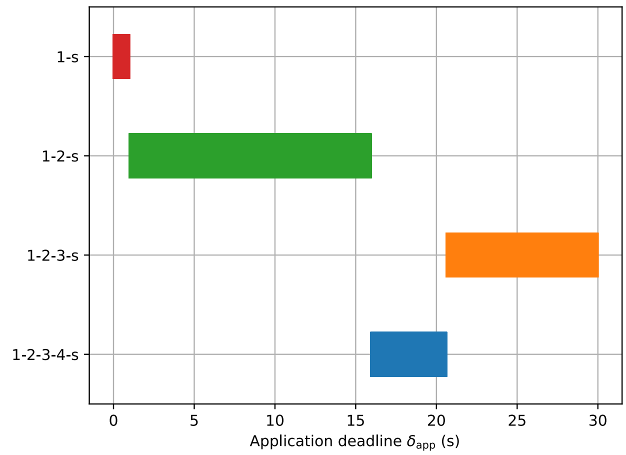

FIGURE 6. Paths used from node 1 towards the sink by the JLAT metric, obtained with (9) for various application deadlines.

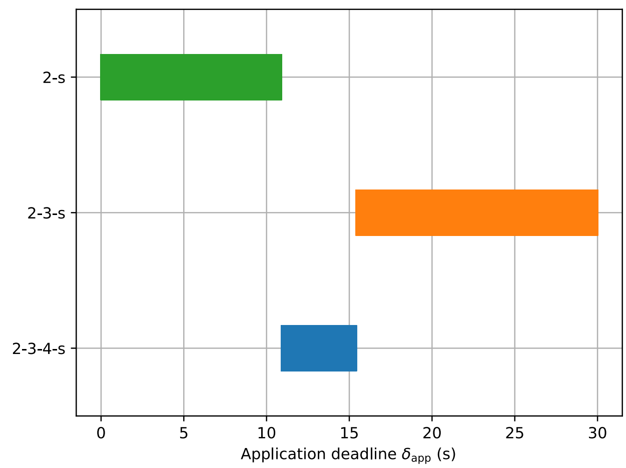

FIGURE 7. Paths used from node 2 towards the sink by the JLAT metric, obtained with (9) for various application deadlines.

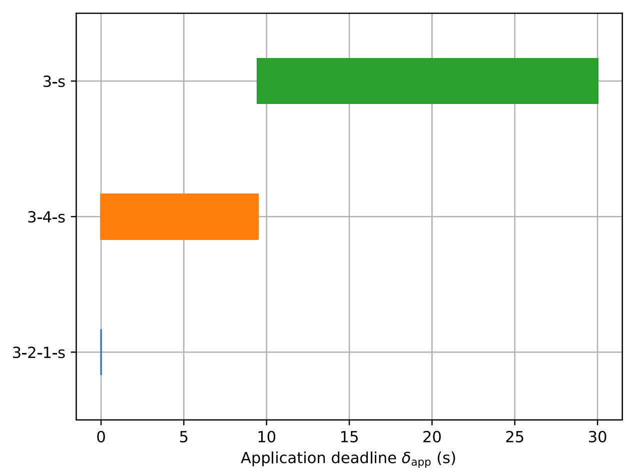

FIGURE 8. Paths used from node 3 towards the sink by the JLAT metric, obtained with (9) for various application deadlines.

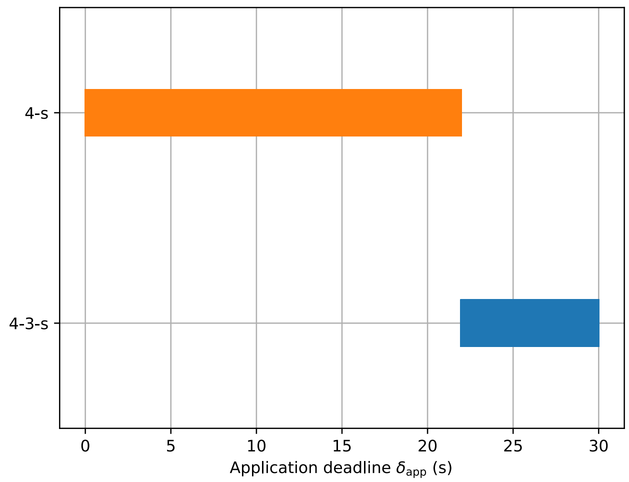

FIGURE 9. **Paths used from node 4 towards the sink by the JLAT metric, obtained with (9) for various application deadlines.**

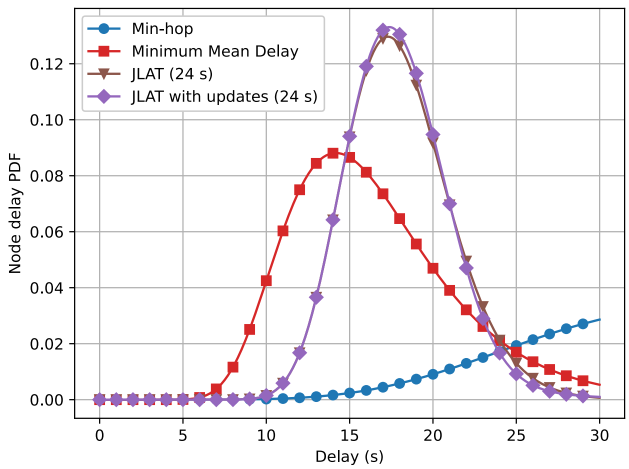

FIGURE 10. **Delay PDF of node 1, $f_N^{(1)}(d)$, for different routing metrics with δ_{app} = 24 s. Obtained with (6) for min-hop, minimum mean delay and JLAT; and with Algorithm 1 for JLAT with updates.**

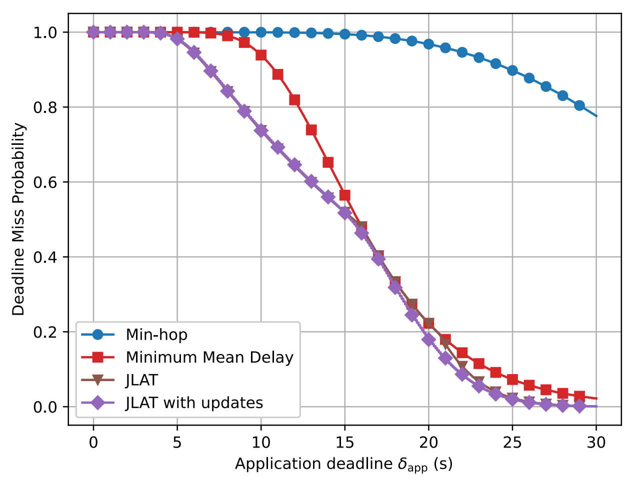

FIGURE 11. **Deadline miss probability (1) from node 1 to the sink for each routing metric studied.**

(i.e. by equation (9) evaluated with the measurement's time-to-deadline instead of δ_{app}). Therefore, the routing tables can be directly extracted from Figs. 6 through 9, taking the remaining time-to-deadline for δ_{app}. Given the routing tables, Algorithm 1 run for node 1 yields its end-to-end PDF (25) for that node.

The delay PDFs of node 1 for min-hop, MMD, standard JLAT and JLAT with updates are presented in Fig. 10. An application deadline $\delta_{app} = 24$ seconds was used. Specifying a different application deadline may result in different delay PDFs for both JLAT cases, because different first-hop neighbours may therefore be chosen (cf. Fig. 6).

Finally, the DMP (1) for node 1 and all four studied routing metrics is shown in Fig. 11. It is to be noted that the abscissa shows application deadline (δ_{app}), which takes values from 0 to 30 s. The path used by min-hop is always 1-*s*, as expected, and the path used by MMD is always 1-2-3-4-*s* (compare Figs. 5 and 11). JLAT uses either path 1-*s*, 1-2-*s*, 1-2-3-*s* or 1-2-3-4-*s* depending on the specified value of δ_{app} (cf. Fig. 6). Lastly, JLAT with updates has equal or better performance than (standard) JLAT, as expected.

REFERENCES

[1] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*. Hoboken, NJ, USA: Wiley, 2007.

[2] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: Application-specific sensor network for environmental monitoring," *ACM Trans. Sen. Netw.*, vol. 6, no. 2, pp. 17:1–17:32, Mar. 2010, doi: 10.1145/1689239.1689247.

[3] S. Barros, "Diseño de protocolos para redes inalámbricas de sensores sobre LatinOS: Metodología y ejemplos," M.S. thesis, Dept. Elect. Eng., Pontificia Universidad Católica de Chile, Santiago, Chile, Jan. 2013.

[4] R. Guerrero, "Field evaluation of sensor to cloud communication metrics for assessing the robustness of a wireless sensor network used in early warning systems," M.S. thesis, Dept. Elect. Eng., Pontificia Universidad Católica de Chile, Santiago, Chile, May 2020.

[5] C. Oberli, R. Guerrero, J. Gironás, C. Escauriaza, S. Barros, and J. P. de Villers Grandchamps, "Experimental assessment of the robustness of a wireless sensor network used for early warning of flash floods," in *Proc. Joint Int. Resilience Conf.*, Nov. 2020, pp. 105–110.

[6] Y. Fan, R. Kalaba, and J. Moore, "Arriving on time," *J. Optim. Theory Appl.*, vol. 127, no. 3, pp. 497–513, 2005.

[7] K. Bérczi, A. Jüttner, M. Laumanns, and J. Szabó, "Arrival time dependent routing policies in public transport," *Discrete Appl. Math.*, vol. 251, pp. 93–102, Dec. 2018.

[8] A. Agafonov and V. Myasnikov, "Stochastic on-time arrival problem with Levy stable distributions," in *Proc. 4th Int. Conf. Intell. Transp. Eng. (ICITE)*, Sep. 2019, pp. 227–231.

[9] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: Out-of-the-box environmental monitoring," in *Proc. Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, St. Louis, MO, USA, Apr. 2008, pp. 332–343.

[10] B.-H. Jang, S. Son, and K.-J. Park, "Deadline-aware routing with probabilistic delay guarantee in cyber-physical systems," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2017, pp. 51–53.

[11] F. Despaux, "Modeling and evaluation of the end-to-end delay in wireless sensor networks," M.S. thesis, Laboratoire Lorrain de Recherche en Informatique et Ses Applications, Université de Lorraine, Nancy, France, Sep. 2015. [Online]. Available: https://hal.univ-lorraine.fr/tel-01751796

[12] H. M. Gursu and W. Kellerer, "Deadline-aware wireless sensor network routing: The JLAT metric," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2017, pp. 612–617.

[13] K. D. Kang, S. H. Son, and J. A. Stankovic, "Managing deadline miss ratio and sensor data freshness in real-time databases," *IEEE Trans. Knowl. Data Eng.*, vol. 16, no. 10, pp. 1200–1216, Oct. 2004.

[14] P. T. Shiva, K. B. Raja, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, "Link-reliability based two-hop routing for QoS guarantee in wireless sensor networks," in *Proc. 16th Int. Symp. Wireless Pers. Multimedia Commun. (WPMC)*, Jun. 2013, pp. 1–6.

[15] Y. Li, C. S. Chen, Y.-Q. Song, Z. Wang, and Y. Sun, "Enhancing real-time delivery in wireless sensor networks with two-hop information," *IEEE Trans. Ind. Informat.*, vol. 5, no. 2, pp. 113–122, May 2009.
[16] B.-R. Chen, G. Peterson, G. Mainland, and M. Welsh, "LiveNet: Using passive monitoring to reconstruct sensor network dynamics," in *Distributed Computing in Sensor Systems*, S. E. Nikoletseas, B. S. Chlebus, D. B. Johnson, and B. Krishnamachari, Eds. Berlin, Germany: Springer, 2008, pp. 79–98.
[17] N. Abu Ali, E. Ekram, A. Eljasmy, and K. Shuaib, "Measured delay distribution in a wireless mesh network test-bed," in *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl.*, Mar. 2008, pp. 236–240.
[18] P. Jurcík, A. Koubâa, M. Alves, E. Tovar, and Z. Hanzálek, "A simulation model for the IEEE 802.15.4 protocol: Delay/throughput evaluation of the GTS mechanism," in *Proc. 15th Int. Symp. Modeling, Anal., Simulation Comput. Telecommun. Syst.*, Oct. 2007, pp. 109–116.
[19] H. Xie, G. Zhang, D. Su, P. Wang, and F. Zeng, "Performance evaluation of RPL routing protocol in 6lowpan," in *Proc. IEEE 5th Int. Conf. Softw. Eng. Service Sci.*, Jun. 2014, pp. 625–628.
[20] D. D. Chaudhary and L. M. Waghmare, "Quality of service analysis in wireless sensor network by controlling end-to-end delay," in *Proc. 7th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, Jul. 2012, pp. 703–708.
[21] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.
[22] P. Raptis, A. Banchs, and K. Paparrizos, "A-simple-and-effective-delay-distribution-analysis-for-IEEE-802.11," in *Proc. IEEE 17th Int. Symp. Pers., Indoor Mobile Radio Commun.*, Sep. 2006, pp. 1–5.
[23] P. Park, P. Di Marco, P. Soldati, C. Fischione, and K. H. Johansson, "A generalized Markov chain model for effective analysis of slotted IEEE 802.15.4," in *Proc. IEEE 6th Int. Conf. Mobile Adhoc Sensor Syst.*, Oct. 2009, pp. 130–139.
[24] W. He, X. Liu, L. Zheng, and H. Yang, "Reliability calculus: A theoretical framework to analyze communication reliability," in *Proc. IEEE 30th Int. Conf. Distrib. Comput. Syst.*, 2010, pp. 159–168.
[25] Y. Ou and W. Heinzelman, "Modeling and performance analysis for duty-cycled MAC protocols with applications to S-MAC and X-MAC," *IEEE Trans. Mobile Comput.*, vol. 11, no. 6, pp. 905–921, Jun. 2012.
[26] Y. Wang, M. Vuran, and S. Goddard, "Cross-layer analysis of the end-to-end delay distribution in wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 1, pp. 305–318, Feb. 2012.
[27] S. M. Ross, "Random variables," in *Introduction to Probability Models*, 10th ed., S. M. Ross, Ed. Boston, MA, USA: Academic, 2010, ch. 2, pp. 21–95. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780123756862000108
[28] H. Singh and D. Singh, "Taxonomy of routing protocols in wireless sensor networks: A survey," in *Proc. 2nd Int. Conf. Contemp. Comput. Informat. (IC3I)*, Dec. 2016, pp. 822–830.
[29] A. Sarkar and T. S. Murugan, "Routing protocols for wireless sensor networks: What the literature says?" *Alexandria Eng. J.*, vol. 55, no. 4, pp. 3173–3183, Dec. 2016.
[30] H. Echoukairi, K. Bourgba, and M. Ouzzif, "A survey on flat routing protocols in wireless sensor networks," in *Advances in Ubiquitous Networking*, E. Sabir, H. Medromi, and M. Sadik, Eds. Singapore: Springer, 2016, pp. 311–324.
[31] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proc. 9th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*. New York, NY, USA: Association for Computing Machinery, 2003, pp. 134–146, doi: 10.1145/938985.939000.

MATÍAS SEPÚLVEDA received the B.S. degree in electrical engineering and the M.S. degree in telecommunications from the Pontificia Universidad Católica de Chile (PUC), Santiago, Chile, in 2018 and 2021, respectively.

His research interests include wireless sensor networks, early warning systems, routing protocols, and the Internet of Things.

CHRISTIAN OBERLI (Member, IEEE) received the B.S. degree (Hons.) in electrical engineering and the M.S. degree in automatic control applied to biomedical engineering from the Pontificia Universidad Católica de Chile (PUC), Santiago, Chile, and the Ph.D. degree in electrical engineering from the University of California at Los Angeles (UCLA), Los Angeles, CA, USA, in 1994, 1996, and 2004, respectively.

Since 2005, he has been with the faculty of the Department of Electrical Engineering, PUC, where he conducts research and education in wireless communications. He is currently the Head of the Wireless Technologies Laboratory and is also an Associate Researcher with the National Research Center for Integrated Natural Disasters Management. His current research interests include wireless sensor networks and resilience of communications systems, including theoretical aspects as well as prototype development, and testbed implementations.

BENJAMIN BECKER received the B.Ed. degree in mathematics and computer science, the B.Sc. degree in computer science, and the M.Ed. degree in mathematics and computer science from Universität Koblenz-Landau, Koblenz, Germany, in 2017, 2019, and 2019, respectively. He is currently pursuing the Ph.D. degree with the Technical University of Darmstadt. He is also a Research Assistant with the Multimedia Communications Laboratory (KOM), Department of Electrical Engineering and Information Technology, Technical University of Darmstadt. His research interests include centered around stochastic time guarantees in wireless sensor networks and resilient digital cities.

PATRICK LIESER received the B.Sc. and M.Sc. degrees in electrical engineering and information technology and the Ph.D. degree from the Technical University of Darmstadt, Darmstadt, Germany, in 2012, 2014, and 2019, respectively.

Since 2019, he has been a Postdoctoral Researcher and the Head of the Multimedia Communications Laboratory (KOM), Research Group Adaptive Communication Systems, Department of Electrical Engineering and Information Technology, Technical University of Darmstadt. His research interests include centered around complex networked systems and infrastructure-independent communication systems.