Formal Approach to Guard Time Optimization for TDMA*

Oday Jubran
Carl von Ossietzky Universität Oldenburg
jubran@informatik.uni-oldenburg.de

Bernd Westphal
Albert-Ludwigs-Universität Freiburg
westphal@informatik.uni-freiburg.de

ABSTRACT

Treating clock drift to avoid message collision and loss is a major challenge when the Time Division Multiple Access (TDMA) protocol is used to schedule communication over a shared medium in networks. We consider clock drift in synchronized networks with tree topology and limited energy, where the root provides the reference clock. We provide a formal approach for analyzing the effect of the slot assignment on the maximal clock drift, given an upper bound on the drift rates of the clocks. We give exact topological characterizations of the slot assignments with largest and least existing clock drift and provide least upper bounds on the clock drift in both cases. We apply our results to derive an optimal, i.e. minimal, safe guard time which effectively avoids message collision or loss due to clock drift.

Keywords

Clock Drift, TDMA, Slot Assignment, Guard Time

1. INTRODUCTION

The Time Division Multiple Access (TDMA) protocol [1] allows multiple network nodes to share a communication medium without message collision or loss. This is achieved by scheduling the communication of the nodes: Time is divided into periodic intervals called *time frames*, where each frame is divided into *time slots*. Each slot is assigned to a node and a node sends messages only during its assigned slots. TDMA in particular provides an opportunity for saving energy by setting nodes into a sleep (power saving) mode during the time slots in which they do not send or listen. TDMA is used in some network communication protocols, such as TTP [2] and FlexRay [3].

*This work was partially supported by the German Research Foundation (DFG) as part of the Transregional Collaborative Research Center "Automatic Verification and Analysis of Complex Systems" (SFB/TR 14 AVACS, `http://www.avacs.org/`).

RTNS 2013 , October 16 - 18 2013, Sophia Antipolis, France.

http://dx.doi.org/10.1145/2516821.2516849.

As TDMA is based on time slots, network nodes are supposed to have the same time. Otherwise, the network may exhibit two critical behaviors: (1) Two nodes send two messages at the same point in time because that point in time still belongs to the time slot of one sender according to its local clock and it already belongs to the time slot of the other sender according to its local clock, causing message collision. (2) A node sends a message to a target node while the latter is in a sleep mode, causing message loss. Message collision and loss may cause serious problems in time critical systems: if a critical message is lost, the system may respond too late or not at all.

Means to provide the same time to all network nodes depend on the network architecture. In the easiest case, all nodes share the same clock signal at the hardware level. Then there is no need for additional clock synchronization mechanisms. If there is one local clock for each node, temperature, voltage change, noise, etc. [4] leads to a variation in the crystal frequencies and thus *clock drift*. In such cases, clock synchronization mechanisms are required. In case of, e.g., FlexRay [3], each node is equipped with a local clock, but all nodes are connected by a bus and receive a common synchronization signal at the beginning of each frame. In general, for example in wireless sensor networks, we cannot assume that all network nodes have access to a common synchronization signal. General clock synchronization mechanisms may involve communication between multiple nodes.

Synchronization mechanisms cannot guarantee the absence of clock drift for two reasons: (1) Two subsequent clock synchronizations may be far enough from each other that a clock drifts critically large in between. (2) Received clock values may not be correct if they are sent by a node which does not provide the reference clock but in turn received its clock value from a third node. A bounded clock drift can be tolerated by using the notion of guard time [5]. Guard time consists of two time intervals that are added to the beginning and the end of each slot. During the guard time of a slot, the assigned node is not allowed to send messages, but it is supposed to listen. In this setting, a bounded clock drift does not yield message collision or loss.

Since guard time is an addition to the slot length, which consequently extends the frame length, guard time reduces the performance of the system: the time required to deliver messages increases due to the increase of the frame length, and extending the listening duration for any node increases energy consumption.

In this work, we consider clock drift in synchronized networks that use TDMA in a tree network topology where the root provides the reference time, assuming that each node is synchronized from its master (parent) within its slot and once per frame. Regarding communication, we can only assume that master (parent) and slave (child) hear each other, but we can not assume that certain nodes are out of each others range; it may happen that all network nodes hear each other and thus cause message collision.

We present a powerful mathematical model of the system, clock drift, and TDMA schedule which allows us to derive an optimal, i.e. minimal, guard time which ensures that the system does not exhibit message collision or loss given the network size, the assignment, the local clock tolerances, and the slot length. In particular, we observe that the optimal guard time depends on the slot assignment. We provide a topological characterization of the assignments that require the largest and least optimal guard time to avoid message collision and loss.

A real-world example for the systems we consider is a wireless fire alarm system in the sense of EN 54-25 [6]. A wireless fire alarm system is a wireless sensor network consisting of a central unit and sensors as nodes, that communicate with each other via radio signals over a shared frequency channel using TDMA. The central unit is the root in the network and each node is equipped with a hardware clock to distinguish its assigned slot in time. The communication between the network nodes is as follows: Each sensor sends a message to its parent, called master, during its assigned slot. The master is in listening mode during the slots of its slaves. As the sensor clocks may exhibit clock drift, clocks are synchronized as follows: Each sensor sends an alive message to its master within its slot, the master replies with an acknowledgment within the same slot and the acknowledgment is time-stamped with the master's clock value. This propagates the value of the central unit clock to the sensors. Wireless fire alarm systems are safety-critical and have to satisfy response deadlines, so message collision and loss have to be avoided. As the sensor nodes in wireless fire alarm system are typically battery powered, energy efficiency is a prominent issue, and overall energy consumption is dominated by energy consumption during both, sending and listening phases.

This work is structured as follows: The remainder of this section discusses related work. Section 2 provides preliminaries for modeling the network. Section 3 presents a formal model of the networks we consider. In Section 4, we explain the notion of guard time. In Section 5, we present our results: the topological characterizations of the slot assignments with the largest and least maximum clock drift, and the optimal guard time. Section 6 discusses significant aspects of the results. Finally, Section 7 presents the conclusion.

1.1 Related Work

Recent research concerned clock synchronization and slot assignment algorithms, and our approach and results may provide quality measurements for such algorithms as we describe in the following.

Clock synchronization algorithms are surveyed in [7]. Each algorithm is intended to fit an environment or a specific topology. For example, The Network Time Protocol NTP [8] is used for networks with hierarchical structure, with a root representing correct time. In this protocol, messages sent through the hierarchy downwards are timestamped to propagate the clock value of the root to the nodes. In this work, we consider a clock synchronization protocol which controls a network in a hierarchical structure, and our results may support analyzing NTP, or extending it with time intervals of optimal lengths.

In addition to clock synchronization, there exists the problem of slot assignment using TDMA, which reflects the question: Which slot should be assigned to which node such that the slot and frame lengths are optimized, communication is fast, and energy is conserved. In [9], general assignment problems are investigated, where some are related to TDMA slot assignment. Some research focuses on the question: How to optimally assign a slot to multiple nodes whose messages do not collide with each other because they are, e.g., not in each other's range. The quality of slot assignment algorithms is measured according to some properties, such as scalability, efficiency, and fault-tolerance. For example, the algorithm in [10] focuses on having a non-recurring slot assignment, towards having an optimal frame size wrt. the number of slots per frame. Other algorithms, as [11], focus on minimizing slot length. Other algorithms, as [12], are intended to be self-stabilizing: To recover from transient errors and faults in a finite time, where a fault can be message collision.

In network communication protocols, as [3], clock synchronization is solved by stating upper bound on clock tolerances. In our work, we do not focus on the methodology of assigning slots to nodes, however, we show how the slot assignment order may affect the maximum clock drift in synchronized networks having tree topology. Our results may provide measuring efficiency of such algorithms and protocols, such as measuring the effect of the slot assignment order on the observed clock drift.

2. PRELIMINARIES

For our formal model of network topologies we need the standard notions of rooted tree and its depth and size.

A finite (directed) graph is a pair $T = (N, E)$ consisting of a finite set of *nodes* N and a set of *edges* $E \subseteq N \times N$; edge $(n, n') \in E$ is *directed* from n to n'. The number of nodes in N is called the *size* of T. A sequence of nodes $n_0, \dots, n_q$ is called *path* of T of *length* q iff the pairs (n_i, n_{i+1}), $0 \leq i < q$, are in E. A path with $n_0 = n_q$ and $q > 0$ is called *cycle*, a graph without a cycle is called *acyclic*.

A *rooted tree* is an acyclic graph T with a unique node n such that, for each $n' \in N$, there is a unique path from n to n' in T; n is called *root* of T. The *depth* $depth(n)$ of a node $n \in N$ is defined inductively as follows: The root has depth 0, and for $(n, n') \in E$, n' has depth $depth(n) + 1$. We say a tree T *has depth* $d \in \mathbb{N}_0$ if d is the maximum depth observed for any node in T.

Let $T = (N, E)$ be a tree with root n. A tree $T' = (N', E')$ with $N' \subseteq N \backslash \{n\}$, $E' \subseteq E$, and root n' such that $(n, n') \in E$ is called *subtree* of T. The trivial tree, which consists of the root only, does not have any subtree. A subtree of T is called *maximal* if T does not have any strictly larger subtree.

Our definition of subtree is non-standard as we require that the subtree root is a direct child of the full tree's root.

3. FORMAL MODEL

In this section, we present a formal model of tree-like communication networks with local clocks. The model allows to formalize message collision and loss. Our notion of synchronized communication formalizes the assumption that communication is scheduled by TDMA and that a node's clock is synchronized at least once per frame with the node's master. The model is the basis for our deductive approach to obtain optimal guard time. To the best of our knowledge, this is the first integrated formal model of communication networks with TDMA and clock drift.

3.1 Network Topologies with Clock Drift

The class of network topologies we consider is modelled as follows.

DEFINITION 1 (NETWORK TOPOLOGY). *A network topology is a finite rooted tree $T = (N, E)$. For each edge $(n, n') \in E$, n is called the* master *of n', denoted by $ms(n')$, and n' is called a* slave *of n. The root of T is called* central unit *and denoted by $cu(T)$. Each node other than the root is called* sensor. *The set of sensors in T is denoted by $Sn(T)$.* ♢

Note that in the trivial network topology consisting of only the central unit, there is no inter-node communication at all, and therefore no message collision or loss. In order to avoid the discussion of the trivial case, in the following, we only consider network topologies T with sensors, i.e. with $Sn(T) \neq \emptyset$, and thus with depth $d > 0$.

To define clock drift and message collision and loss, it is sufficient to observe the clock value of each node in the topology, represented by a positive real number ($\mathbb{R}_0^+$), and whether a node is sending a message or listening, represented by two boolean ($\mathbb{B}$) values. Note that our model *does neither* observe any physical activity during the corresponding communication scenario, nor the message content, since we are treating only message collision and loss.

As we assume that the central unit clock provides the reference time, that is, the central unit clock does not drift. Even if the central unit has a bad crystal compared to a reference clock outside of the system, the sensors in the network follow the central unit clock. For simplicity, we assume that all clocks start with value 0. In Section 6.2, we show how our results can be generalized to evolutions where clocks have different initial values at time 0.

DEFINITION 2 (EVOLUTION). *An* evolution *over a network topology $T = (N, E)$ is an interpretation $\mathcal{I}$ of the observables $clk_n : \mathsf{Time}\ (= \mathbb{R}_0^+)$, $send_n : \mathbb{B}$, and $listen_n : \mathbb{B}$ for $n \in N$ such that*

1. *$\mathcal{I}(clk_n)(0) = 0$ for each $n \in N$, and*
2. *$\mathcal{I}(clk_{cu(T)})(t) = t$ for each $t \in \mathsf{Time}$.*

We write $Evo(T)$ to denote the set of all evolutions over T. We write $clk_n^{\mathcal{I}}(t)$, $send_n^{\mathcal{I}}(t)$, and $listen_n^{\mathcal{I}}(t)$ to denote $\mathcal{I}(clk_n)(t)$, $\mathcal{I}(send_n)(t)$, and $\mathcal{I}(listen_n)(t)$, respectively. ♢

The following notion of synchronization point is useful for the following sections.

DEFINITION 3 (SYNCHRONIZATION POINT). *Let $\mathcal{I}$ be an evolution over network topology T. A point in time $t \in \mathsf{Time}$ is called* synchronization point *of sensor $n \in Sn(T)$ in $\mathcal{I}$ iff the clock of n has the same value as the clock of its master, i.e. if $clk_n^{\mathcal{I}}(t) = clk_{ms(n)}^{\mathcal{I}}(t)$.* ♢

Formally, the *clock speed* is simply the derivative of an evolution of clock values with respect to time. The *clock drift* is the difference between a clock and the reference clock, not only the difference to the master's clock, and the *drift rate* is the rate of change of clock drift.

Note that the evolution of clock values is not restricted by Definition 2: clock values may arbitrarily change, in particular non-continously. Thus, clock speed and drift rate are in general only partial functions. However, we later specify assumptions on the change of clock values.

DEFINITION 4 (CLOCK DRIFT). *Let $\mathcal{I}$ be an evolution over network topology $T = (N, E)$.*

1. *The* clock speed *of node $n \in N$ in $\mathcal{I}$, denoted by $\varphi_n^{\mathcal{I}}$, is the first derivative of the interpretation of clk_n with respect to time, i.e.*
$$\varphi_n^{\mathcal{I}} = \frac{\partial}{\partial t}\ clk_n^{\mathcal{I}}(t).$$
2. *The* clock drift *of node $n \in N$ in $\mathcal{I}$ at time $t \in \mathsf{Time}$, denoted by $\varrho_n^{\mathcal{I}}(t) \in \mathbb{R}$, is the difference between the clock values of n and the central unit at time t in $\mathcal{I}$, i.e.*
$$\varrho_n^{\mathcal{I}}(t) = clk_n^{\mathcal{I}}(t) - clk_{cu(T)}^{\mathcal{I}}(t).$$
3. *The* drift rate *of node $n \in N$ in $\mathcal{I}$, denoted by $\Delta_n^{\mathcal{I}}$, is the first derivative of the clock drift of n in $\mathcal{I}$ with respect to time, i.e.*
$$\Delta_n^{\mathcal{I}} = \frac{\partial}{\partial t}\ \varrho_n^{\mathcal{I}}(t).$$

We may omit the superscript $\mathcal{I}$ if the interpretation is clear from the context. ♢

In phases where the evolution of a node's clock value is differentiable, there is the following relation between the current clock value, and an earlier clock value (of the node's master) and the drift rate.

NOTE 1. *Let $\mathcal{I}$ be an evolution over a network topology T and let $clk_n^{\mathcal{I}}$ be differentiable on the interval $(t_1, t_3) \subset \mathbb{R}_0^+$.*

1. *Then*
$$\forall\, t_2 \in [t_1, t_3) \bullet \varrho_n^{\mathcal{I}}(t_2) = \varrho_n^{\mathcal{I}}(t_1) + \int_{t_1}^{t_2} \Delta_n^{\mathcal{I}}(t)\,dt.$$
2. *If t_1 is a synchronization point of n, then*
$$\forall\, t_2 \in [t_1, t_3) \bullet \varrho_n^{\mathcal{I}}(t_2) = \varrho_{ms(n)}^{\mathcal{I}}(t_1) + \int_{t_1}^{t_2} \Delta_n^{\mathcal{I}}(t)\,dt. \diamond$$

PROOF. (1) Def. 4 and fundamental theorem of calculus. (2) Definition 3, Definition 4, and 1. □

With Definition 2, we can precisely characterize the undesired conditions of message collision and loss. Note that we assume that messages sent between a sensor and its master do not collide with each other during the sensor's slot; we only consider message collision or loss due to clock drift, but not defects of the communication protocols employed between master and slave and not flaws in the employed clock synchronization protocol.

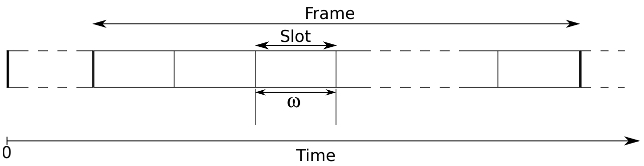

Figure 1: TDMA frames and slots.

Definition 5 (Message Collision/Loss). *An evolution $\mathcal{I}$ over network topology T is said to have*

1. message collision *at time $t \in \mathsf{Time}$ between two different sensors $n_1, n_2 \in Sn(T)$ iff both send at t, i.e. if*

$$send^{\mathcal{I}}_{n_1}(t) \wedge send^{\mathcal{I}}_{n_2}(t),$$

2. message loss *at time $t \in \mathsf{Time}$ for sensor $n \in Sn(T)$ iff n is sending at t while its master is not listening, i.e. if*

$$send^{\mathcal{I}}_{n}(t) \wedge \neg listen^{\mathcal{I}}_{ms(n)}(t).$$

We write $coll^{\mathcal{I}}_{n_1,n_2}(t)$ ($loss^{\mathcal{I}}_n(t)$) iff $\mathcal{I}$ has a message collision (loss) at t between sensors n_1, n_2 (for sensor n). ♢

3.2 Scheduled Communication

The notions of frame and slot employed by TDMA can simply be formalized as a partitioning of the time domain (cf. Figure 1).

Definition 6 (Frame, Slot). *The* TDMA schedule *for $k \in \mathbb{N}^+$ sensors with* slot length $\omega \in \mathbb{R}^+$ *is a pair $(slot, frm)$ of functions*

$$frm : \mathsf{Time} \to \mathbb{N}^+, \quad \textit{and} \quad slot : \mathsf{Time} \to \mathbb{N}^+$$

that are point-wise defined as

$$frm(t) = \left\lfloor \frac{t}{k \cdot \omega} \right\rfloor + 1, \quad slot(t) = \left(\left\lfloor \frac{t}{\omega} \right\rfloor \mod k \right) + 1.$$

A time interval $[t_1, t_2)$ is called the slot *(of $(slot, frm)$) with* slot id $(i, j) \in \mathbb{N}^+ \times \mathbb{N}^+$ *iff*

$$t_2 - t_1 = \omega \wedge \forall\, t \in [t_1, t_2) \bullet slot(t) = i \wedge frm(t) = j. \quad ♢$$

A scheduling or (slot) assignment is then just a mapping of sensors to slots. For simplicity, we consider bijective assignments, i.e. for each sensor, there exists only one assigned slot and vice versa, and we assume that the assignment is fixed during a system evolution. In Section 6, we discuss conditions under which our results generalize to dynamic assignments.

Definition 7 (Scheduled Evolution). *Let T be a network topology and $(slot, frm)$ a TDMA schedule for $k = |Sn(T)|$ sensors with slot length ω. An evolution $\mathcal{I}$ over T is called* scheduled *wrt. $(slot, frm)$ iff there exists an* assignment *of sensors to slots, i.e. a bijection*

$$assign : Sn(T) \to \{1, \ldots, k\}$$

such that:

1. *Each sensor $n \in Sn(T)$ sends messages only during the assigned slot according to its local clock, i.e.*

$$\forall\, t \in \mathsf{Time} \bullet send^{\mathcal{I}}_n(t) \implies slot(clk^{\mathcal{I}}_n(t)) = assign(n).$$

2. *For each sensor $n \in Sn(T)$, its master is listening in the slot assigned to n according to the master's clock, i.e.*

$$\forall\, t \in \mathsf{Time} \bullet slot(clk^{\mathcal{I}}_{ms(n)}(t)) = assign(n) \implies listen^{\mathcal{I}}_{ms(n)}(t).$$

By $Evo(T, \omega, assign) \subseteq Evo(T)$, we denote the set of evolutions scheduled by 'assign' for slot length ω. ♢

3.3 Clock Synchronization

Two main assumptions of our work are formalized by the notion of synchronized evolution in Definition 8 below.

Firstly, for simplicity, we assume that each sensor has at least one synchronization point per frame. This assumption is a simplification because if the employed synchronization mechanism uses the same medium that is managed by TDMA, then message loss or collision during a sensor's slot may inhibit proper synchronization in general; one can imagine our assumption to require a separate reliable channel for synchronization. Note that our reasoning doesn't get circular by this assumption. In the following, we derive a safe guard time under this assumption one synchronization per frame. Safe guard times have the property that message loss or collision is effectively avoided, thus with a safe guard time, the synchronization mechanism may well use the shared medium managed by TDMA without being affected by message loss or collision. If the medium as such is reliable, we obtain one synchronization point per slot as needed for our approach. See Section 6.3 for a discussion of safe guard times for unreliable media. Further note that the notion of synchronized evolution models a wide range of explicit synchronization messages as well as piggy-backed timestamps.

The second assumption is that the evolution of clock values is differentiable except for synchronization points. For simplicity, we assume that there is at most one synchronization point per slot where the local clock value is updated. In order to model systems which use multiple synchronization points per slot, one can e.g. designate the latest synchronization points to be considered. As clock values evolve from synchronization points, we assume that the right-side derivative exists for all points in time.

Definition 8 (Synchronized Evolution). *Let $\mathcal{I}$ be an evolution over T which is scheduled wrt. $(slot, frm)$. $\mathcal{I}$ is called* synchronized *iff the following conditions hold:*

1. *Each sensor has at least one synchronization point in each of its slots, i.e.*

$$\forall\, n \in Sn(T), j \in \mathbb{N}_0\ \exists\, t \in \mathsf{Time} \bullet frm(t) = j \wedge slot(t) = assign(n) \wedge clk^{\mathcal{I}}_n(t) = clk^{\mathcal{I}}_{ms(n)}(t).$$

2. *For each $n \in N$, $clk^{\mathcal{I}}_n$ is differentiable except for 0 and at most one point in each slot of n, i.e. if $\frac{\partial}{\partial t}\ clk^{\mathcal{I}}_n(t)$ does not exist at $t, t' \in \mathsf{Time} \setminus \{0\}$ with $t \neq t'$ then there are two different slots $[t_1, t_2)$ and $[t'_1, t'_2)$ assigned to node n such that that $t \in [t_1, t_2)$ and $t' \in [t'_1, t'_2)$; the right-side derivative of $clk^{\mathcal{I}}_n$ exists for all $t \in \mathsf{Time}$.* ♢

Lemma 1. *Let $\mathcal{I}$ be a synchronized evolution over topology T with k sensors. The* distance *between two*

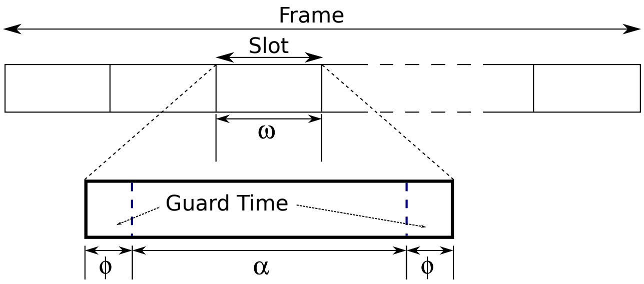

Figure 2: Guard Time.

synchronization points of a sensor $n \in Sn(T)$ is at most $(k+1) \cdot \omega$, i.e.

$$\forall t \in \mathsf{Time}\ \exists t' \in \mathsf{Time} \bullet$$
$$0 < t' - t < (k+1) \cdot \omega \wedge clk_n(t') = clk_{ms(n)}(t').$$

Proof Sketch (cf. Appendix). Let $n \in Sn(T)$ be a sensor and $t \in \mathsf{Time}$. By Definition 8, there is a synchronization point in the next slot of n following t. In the worst case, t is the lower boundary of a slot of n and a synchronization point, then there is another synchronization point in the subsequent slot, at the upper boundary the latest. The claimed distance follows from Definitions 6 and 7. □

Note 2. *Let $\mathcal{I}$ be a synchronized evolution over network topology T. Then $\varphi_n^{\mathcal{I}}$ and $\Delta_n^{\mathcal{I}}$ are defined on* Time *except for at most one point in each slot of n.* ◇

Proof. By Definition 8, $clk_n^{\mathcal{I}}(t)$ is differentiable. □

4. GUARD TIME

In this section, we formally define the notion of guard time. A sensor obeys a guard time $\phi \in \mathbb{R}_0^+$ if it does not send for a duration of ϕ at the beginning and end of its assigned slot (cf. Figure 2). Note that obeying guard time is defined in terms of the local clock of the sensor: the sensor does not send if its local clock points to a value within the guard time of the sensor's slot.

Definition 9 (Guard Time). *An evolution $\mathcal{I}$ over topology T has* guard time *$\phi \in \mathbb{R}_0^+$ iff (1) $\mathcal{I}$ is scheduled with a slot length $\omega \geq 2\phi$ and (2) sensors don't send for a duration of ϕ at the beginning and end of their slot, i.e.*

$$\forall n \in Sn(T), t \in \mathsf{Time} \bullet send_n^{\mathcal{I}}(t) \implies$$
$$slot(clk_n^{\mathcal{I}}(t) - \phi) = slot(clk_n^{\mathcal{I}}(t) + \phi) = slot(clk_n^{\mathcal{I}}(t)).$$

For each slot $[t_1, t_2)$ of $\mathcal{I}$ with guard time ϕ, the time intervals $[t_1, t_1 + \phi)$ and $[t_2 - \phi, t_2)$ are called the (left and right) guard intervals *of the slot. The time interval $[t_1 + \phi, t_2 - \phi)$ is called* α-interval *of the slot.* ◇

The following theorem derives, given a guard time ϕ, sufficient conditions on the clock drift in order to effectively avoid message collision and loss.

Theorem 1. *Let $\mathcal{I}$ be a scheduled evolution over topology T with k sensors and guard time ϕ.*

1. $\mathcal{I}$ does not have any message collision if

$$\forall n \in Sn(T), t \in \mathsf{Time} \bullet |\varrho_n^{\mathcal{I}}(t)| \leq \phi.$$

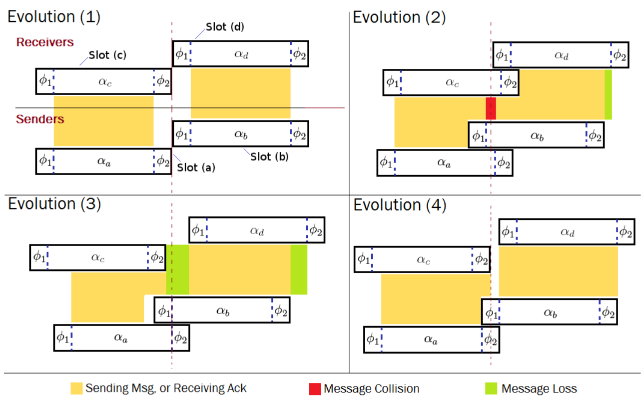

Figure 3: Safe Guard Time

2. $\mathcal{I}$ does not have any message loss if

$$\forall n \in Sn(T), t \in \mathsf{Time} \bullet |\varrho_n^{\mathcal{I}}(t)| \leq \frac{\phi}{2}.$$

Proof Sketch (cf. Appendix).
1. By premise, sensors obey the guard time and, because clock drift is bounded by the guard time, they send in their slot wrt. the central unit clock. Any message collision would yield a contradiction to the assignment being bijective.

2. By premise, sensors obey the guard time and, because clock drift is bounded by $\frac{\phi}{2}$, they send well inside their slots wrt. the central unit clock. Masters listen throughout the slots of their slaves. This phase comprises, by the bound on clock drift, all sending points of the slaves, so there is no message loss. □

To illustrate Theorem 1, consider the example in Figure 3. It assumes a synchronized system over a network topology T with 4 nodes $n_a, n_b, n_c, n_d \in N$ where n_c is master of n_a and n_d is master of n_b. Slots a and b are assigned to n_a and n_b, respectively. Given a guard time ϕ, we show different evolutions with different clock drift values for the given nodes and the possibility of message collision or loss. Note that by Definition 7, masters n_c and n_d listen during slots a and b, respectively. In Evolution (1), all clocks run at the same speed, and therefore there is no message collision or loss. In Evolution (2), there is clock drift with an absolute value that is greater than ϕ, therefore message collision between n_a, n_b, and n_c *and* loss for n_b occur. In Evolution (3), the absolute values of clock drift do not exceed ϕ but only $\frac{\phi}{2}$. There is no message collision, but there exists message loss for n_b. In Evolution (4), the absolute values of clock drift do not exceed $\frac{\phi}{2}$, thus neither message collision nor loss are exhibited.

The following corollary summarizes that the condition for avoiding message loss is stronger than the one avoiding message collision alone.

Corollary 1. *Let $\mathcal{I}$ be a scheduled evolution over topology T with guard time ϕ. If $\frac{\phi}{2}$ is an upper bound on the absolute value of clock drift in $\mathcal{I}$, i.e. if*

$$\forall n \in Sn(T), t \in \mathsf{Time} \bullet |\varrho_n^{\mathcal{I}}(t)| \leq \frac{\phi}{2}$$

holds, then $\mathcal{I}$ has neither message collision nor message loss.

PROOF. Theorem 1. $\square$

Note that the absence of message collision and loss does not necessarily imply a violation of these conditions. For example, a sensor clock may have a high speed in the first half of a frame and a low speed in the second half of the frame in a way such that it has a big maximal clock drift but still is perfectly on time for its next slot.

In Lemma 2, we observe, for completeness, sufficient conditions for message collision and loss. It states that the bounds given by Theorem 1 are minimal in the sense that we can construct evolutions with guard time strictly smaller than ϕ (or $\frac{\phi}{2}$) which have message collision (or loss).

LEMMA 2. *Let $\mathcal{I}$ be a scheduled evolution over network topology T with slot length ω.*

1. *If there are two sensors $n_1, n_2 \in Sn(T)$ and a point in time $t \in \mathsf{Time}$ such that*
   - *$assign(n_2) = assign(n_1) + 1$,*
   - *$t = t_2$ for a slot $[t_1, t_2)$ of n_1,*
   - *both nodes send continuously during their α-interval, and*
   - *$\varrho^{\mathcal{I}}_{n_1}(t) = \phi + \varepsilon$ and $\varrho^{\mathcal{I}}_{n_2}(t) = -\phi$ for $0 < \varepsilon < \omega$,*

   then there is message collision between n_1 and n_2 at t.
2. *If there is a sensor $n \in Sn(T)$ and a point in time $t \in \mathsf{Time}$ such that*
   - *$t = t_2 - \frac{\phi}{2}$ for a slot $[t_1, t_2)$ of n,*
   - *n sends continuously during its α-interval, and*
   - *$\varrho^{\mathcal{I}}_{n}(t) = \frac{\phi}{2} + \varepsilon$ and $\varrho^{\mathcal{I}}_{ms(n)}(t) = -\frac{\phi}{2}$ for $0 < \varepsilon < \omega$,*

   then there is message loss at t.

PROOF SKETCH (CF. APPENDIX). By construction. There exists an evolution which satisfies the premises and, by sending throughout the α-interval, exhibit collision between nodes n_1 and n_2 and loss of a message from n, respectively. $\square$

5. OPTIMAL SAFE GUARD TIME

Recall that the goal of this work is a *minimal* guard time, which guarantees the absence of message collision and loss, in order to obtain a frame length as short as possible. Thereby (1) the response time of sensors is minimized, i.e., the maximum duration between the slots of a sensor in two consecutive frames, and (2) energy consumption, which is for instance in wireless fire alarm systems dominated by energy consumption in transmission phases, can be reduced because masters need only listen for a shorter time.

The local clocks in a system are typically determined by crystal devices. There are bounds on the quality of those devices and they are sensitive to environmental conditions such as temperature. Manufacturers of those devices often guarantee bounds on the drift rate for certain environmental conditions.

DEFINITION 10 (BOUNDED DRIFT RATE). *Let $\mathcal{I}$ be a synchronized evolution over topology T. $\Delta^{max} \in \mathbb{R}^+_0$ is a least upper bound on the magnitude of drift rate in $\mathcal{I}$ iff Δ^{max} is the smallest number such that*

$$\forall\, n \in Sn(T), t \in \mathsf{Time} \bullet |\Delta^{\mathcal{I}}_n(t)| \le \Delta^{max}.$$

We use $Evo_{sync}(T, \omega, assign, \Delta^{max})$ to denote the set of all synchronized evolutions over T with slot length ω which are scheduled by assign and for which $\Delta^{max} \in \mathbb{R}^+_0$ is a least upper bound on the magnitude of drift rate. $\Diamond$

Assuming a bound Δ^{max} on the drift rate, we use Theorem 1 to characterize the existence of a safe guard time wrt. Δ^{max} and, if a safe guard time exists, we derive the optimal safe guard time.

DEFINITION 11 (SAFE GUARD TIME). *A guard time $\phi \in \mathbb{R}^+_0$ is said to be a* safe for a network topology T, slot length ω, schedule *assign*, and least upper bound $\Delta^{max} \in \mathbb{R}$ on the drift rates *iff no synchronized evolution*

$$\mathcal{I} \in Evo_{sync}(T, \omega, assign, \Delta^{max})$$

exhibits message collision or message loss. $\Diamond$

Note that a safe guard time need not exist for a given topology, slot length, assignment, and clock drift bound. If local clocks drift with a large amount during a frame, collision or loss may happen before a re-synchronization of clocks is possible.

In the following, we in particular study the effect of the assignment on the maximum clock drift, so we define the maximum clock drift of an assignment in a given topology.

DEFINITION 12 (MAXIMUM CLOCK DRIFT). *Let T be a network topology. We call $\varrho^{max}_{\omega,\Delta^{max}}(assign, n) \in \mathbb{R}^+_0$ the* maximum clock drift of node $n \in Sn(T)$ under assignment '*assign*' *iff it is the least upper bound of the clock drift of node n in any synchronized evolution with slot length ω and least upper bound $\Delta^{max} \in \mathbb{R}^+_0$ on the clock drift rates, i.e. if*

$$\varrho^{max}_{\omega,\Delta^{max}}(assign, n) = \sup\{\varrho^{\mathcal{I}}_n(t) \mid \mathcal{I} \in Evo_{sync}(T, \omega, assign, \Delta^{max}), t \in \mathsf{Time}\}.$$

The maximum clock drift of assignment '*assign*' in T *is*

$$\varrho^{max}_{\omega,\Delta^{max}}(assign) = \sup\{\varrho^{max}_{\omega,\Delta^{max}}(assign, n) \mid n \in Sn(T)\}. \Diamond$$

In Theorem 2, we characterize the existence and value of an optimal safe guard time in terms of a new notion of *forward distance*. The forward distance from a sensor n_1 to a sensor n_2 is simply the number of slots between any slot assigned to n_1 and the next slot assigned to n_2, which may lie in the same frame or in the subsequent frame. With the forward distance notion, we can sum up the forward distances between sensors along a path, to compute the time required to deliver a clock value from the first sensor to the last sensor in the path, given that each sensor is synchronized within its assigned slot. We use D_{assign} to denote the maximum sum of forward distances of sensors along any path in the network topology. With D_{assign}, we can compute the maximum clock drift for any given assignment, and subsequently, an optimal guard time.

DEFINITION 13 (FORWARD DISTANCE). *Given an assignment assign of slots to nodes for network topology T of size k, the* forward distance *between two sensors $n, n' \in Sn(T)$ is defined by the function*

$$fdist : Sn(T) \times Sn(T) \to \mathbb{N}^+$$

which is defined point-wise as follows:

$$fdist_{assign}(n, n') =$$
$$\begin{cases} assign(n') - assign(n) & \text{if } assign(n') > assign(n) \\ assign(n') + k - assign(n) & \text{if } assign(n') \leq assign(n). \end{cases}$$

We use D_{assign} to denote the maximum of the sums of the forward distances between nodes on a path in T, i.e.

$$D_{assign} =$$
$$max \left\{ \sum_{i=1}^{d-1} fdist_{assign}(n_i, n_{i+1}) \;\middle|\; n_0, n_1, \ldots, n_d \text{ path in } T \right\}. \diamond$$

LEMMA 3. *Let $n \in Sn(T)$ be a sensor of a network topology T with k sensors. Then*

$$\varrho^{max}_{\omega,\Delta^{max}}(assign, n) =$$
$$\left(\sum_{i=1}^{d-1} fdist_{assign}(n_i, n_{i+1}) + k + 1 \right) \cdot \omega \cdot \Delta^{max} \quad (1)$$

where $n_0, n_1, \ldots, n_d$ is the path from the central unit to node $n = n_d$ in T.

PROOF SKETCH (CF. APPENDIX). By induction over the depth of nodes d. For the base case and the step, first show "$\leq$" using Note 1 and then "$\geq$" by construction. $\square$

COROLLARY 2. *Let T be a network topology with k sensors and let assign be an assignment of slots to nodes for T. Then*

$$\varrho^{max}_{\omega,\Delta^{max}}(assign) = (D_{assign} + k + 1) \cdot \omega \cdot \Delta^{max}.$$

PROOF. Lemma 3. $\square$

THEOREM 2. *Let T be a network topology with k sensors and let $\alpha \in \mathbb{R}^+$ be the length of the α-intervals.*

There exists an optimal, i.e. smallest, safe guard time for T wrt. the least upper bound $\Delta^{max} \in \mathbb{R}^+_0$ on the clock drift rates iff

$$\Delta^{max} < \frac{1}{4 \cdot (D_{assign} + k + 1)}.$$

The optimal safe guard time for T wrt. Δ^{max} is given by

$$\phi_{opt} = \alpha \cdot \frac{2 \cdot (D_{assign} + k + 1) \cdot \Delta^{max}}{1 - 4 \cdot (D_{assign} + k + 1) \cdot \Delta^{max}}.$$

PROOF SKETCH (CF. APPENDIX). Study the solutions of the equation system induced by the sufficient and necessary criterion for $\phi \in \mathsf{Time}$ being a safe guard time provided by Theorem 1 and Lemma 2, and the value of $\varrho^{max}_{\omega,\Delta^{max}}(assign)$ given by Corollary 2. $\square$

By Definition 13, the forward distances and thus the maximum sum D_{assign} depend on the slot assignment. In the following, we derive two optimal guard times by studying the relation between the shape of a network topology and the slot assignment:

In Section 5.1, we derive a safe guard time that is optimal if we cannot make any assumptions on the assignment. We topologically characterize those assignments which may have collision or message loss if the guard time is chosen smaller than the optimal one.

In Section 5.2, we derive an optimal safe guard time for the assignments with the least possible optimal guard time, and we show the topological characterizations of such assignments. We consider such assignments as the best case. Surprising about our results is that, in the best case, the depth of a network topology as such is not the limiting factor for clock precision; it is the size of subtrees.

5.1 Worst Case Assignment

The worst case reflects the case where the slot assignment order yields the largest possible sum of forward distances along any path in the topology, which consequently yields the largest clock drift. The largest sum of forward distances along a path is obtained as follows: Let the number of slots in a frame be k. The forward distance from a sensor n_1 to a sensor n_2 can reach up to $k - 1$ slots, only if any slot assigned to n_2 has a following slot assigned to n_1, given the assumption of bijective assignment. For a path $n_1, ..., n_s$ with $s \in \mathbb{N}$ nodes, the largest sum of forward distances along the path can reach up to $(s - 1)(k - 1)$ slots, which is maximized by choosing the largest s, namely the tree depth.

LEMMA 4. *Let assign be an assignment of slots to nodes for network topology T of depth d with k sensors.*

1. *$D_{assign} \leq (d - 1)(k - 1)$.*
2. *$D_{assign} = (d - 1)(k - 1)$ iff there exists a path*

$$n_0, n_1, ..., n_d$$

in T such that $n_0 = cu(T)$, and the sensors on the path are assigned reverse adjacent slots by assign, i.e.

$$assign(n_i) = (assign(n_{i+1}) + 1) \mod k \quad (2)$$

for $1 \leq i < d$ (called (2)-path for short).

PROOF SKETCH (CF. APPENDIX). For networks of depth 1, both claims hold trivially so let T be a network topology of depth $d \geq 2$. Point (1) follows from *assign* being bijective and Definition 13. For Point (2), show both directions of the bi-implication separately. $\square$

COROLLARY 3. *Let T be a network topology of depth d with k sensors. Let $\Delta^{max} \in \mathbb{R}^+_0$ be a least upper bound on the clock drift rates, $\omega \in \mathbb{R}^+_0$ a slot length, and 'assign' an assignment of nodes to slots.*

1. *$\varrho^{max}_{\omega,\Delta^{max}}(assign) \leq (d(k - 1) + 2) \cdot \omega \cdot \Delta^{max}$.*
2. *$\varrho^{max}_{\omega,\Delta^{max}}(assign) = (d(k - 1) + 2) \cdot \omega \cdot \Delta^{max}$ iff 'assign' has a (2)-path.*
3. *Let $\alpha \in \mathbb{R}^+$ be the length of the α-intervals. There exists a safe guard time for T wrt. Δ^{max} iff*

$$\Delta^{max} < \frac{1}{4(d(k - 1) + 2)}.$$

A safe guard time for T wrt. Δ^{max} is given by

$$\phi_{opt} = \alpha \cdot \frac{2(d(k - 1) + 2) \cdot \Delta^{max}}{1 - 4(d(k - 1) + 2) \cdot \Delta^{max}}.$$

For assignments assign with a (2)-path, ϕ_{opt} is the optimal, i.e. smallest safe guard time.

PROOF. Points (1) and (2) follow from Lemma 4,

$$(d - 1)(k - 1) + k + 1 = d(k - 1) + 2, \quad (3)$$

and Lemma 3. Point (3) follows from Points (1) and (2), and Lemma 4, Equation (3), and Theorem 2. $\square$

5.2 Best Case Assignment

In the best case, the slot assignment order yields the least possible maximum sum of forward distances along any path in the topology, which consequently yields the least maximum clock drift of any sensor. The maximum sum of forward distances along a path $n_1, ..., n_s$ is minimized when the forward distance between n_i and n_{i+1}, for $1 \leq i < s$, is minimized. The minimum possible forward distance between n_i and n_{i+1} is 1 when the slot assigned to n_{i+1} is the next slot adjacent to the slot assigned to n_i. However, this distance cannot be achieved for all paths; if n_i has another slave n', and the forward distance from n_i to n_{i+1} is 1, then the distance from n_i to n' is greater than 1. To minimize the maximum sum of forward distances along any path in the topology, the sensors belonging to each subtree have to be assigned adjacent slots, and within the adjacent slots, each slave is assigned a slot that is after the slot assigned to its master.

Note that in this case, the least possible maximum sum of forward distances along any path in the topology depends on the size of the largest subtree, not on the tree depth as such. Yet the tree depth is a lower bound on the size of the largest subtree.

LEMMA 5. *Let assign be an assignment of slots to nodes for network topology $T = (N, E)$ of depth d with k sensors and maximal subtree(s) of size $K \in \mathbb{N}^+$.*

1. $D_{assign} \geq K - 1$.
2. $D_{assign} = K - 1$ *iff*
   (a) *For each path $n_0, n_1, ..., n_m$, where $m \in \mathbb{N}^+$ and $n_0 = cu(T)$, the forward distance between n_1 and each n_i, for $1 < i \leq m$, is at most $K - 1$, i.e.*
   $$\forall\, 1 < i \leq m \bullet \mathit{fdist}_{assign}(n_1, n_i) \leq (K - 1)$$
   Note that, given that the maximal tree size is K, the slots of any subtree of size K are adjacent by this condition.
   (b) *Each slave is assigned a slot that is after the slot assigned to its master, i.e.*
   $$\forall\, n, n', n'' \in Sn(T) \bullet (n, n'), (n', n'') \in E$$
   $$\implies \mathit{fdist}_{assign}(n, n') < \mathit{fdist}_{assign}(n, n'')$$

PROOF SKETCH (CF. APPENDIX). For network topologies of depth 1, the two claims hold trivially. For network topologies of depth $d \geq 2$, firstly show that conditions (2a) and (2b) establish forward distances which imply $D_{assign} = K - 1$. Secondly, show that if any of (2a) and (2b) is violated, then $D_{assign} > K - 1$ follows from Definition 13. ☐

COROLLARY 4. *Let T be a network topology of depth d with k sensors and maximal subtree(s) of size $K \in \mathbb{N}^+$. Let $\Delta^{max} \in \mathbb{R}_0^+$ be a least upper bound on the clock drift rates, $\omega \in \mathbb{R}_0^+$ a slot length, and assign an assignment of nodes to slots.*

1. $\varrho^{max}_{\omega,\Delta^{max}}(assign) \geq (K + k) \cdot \omega \cdot \Delta^{max}$.
2. $\varrho^{max}_{\omega,\Delta^{max}}(assign) = (K + k) \cdot \omega \cdot \Delta^{max}$ *iff assign satisfies the conditions of Lemma 5.2b.*
3. *Let $\alpha \in \mathbb{R}^+$ be the length of the α-intervals. There exists a safe guard time for T wrt. Δ^{max} iff*
   $$\Delta^{max} < \frac{1}{4(K + k)}.$$
   The optimal safe guard time for T wrt. Δ^{max} is given by
   $$\phi_{opt} = \alpha \cdot \frac{2(K + k) \cdot \Delta^{max}}{1 - 4(K + k) \cdot \Delta^{max}}.$$
   It is safe for exactly those assignments of nodes to slots which satisfy the conditions of Lemma 5.

PROOF. Points (1) and (2) follow from Lemma 3, Lemma 5, and

$$(K - 1) + k + 1 = K + k. \tag{4}$$

Point (3) follows from points (1) and (2), and Lemma 4, equation (4), and Theorem 2. ☐

For network topologies of depth 1, the optimal guard time does not depend on the assignment.

COROLLARY 5. *Let T be a network topology of depth 1 with k sensors. Let $\Delta^{max} \in \mathbb{R}_0^+$ be a least upper bound on the clock drift rates, $\omega \in \mathbb{R}_0^+$ a slot length, and assign an assignment of nodes to slots.*

1. $\varrho^{max}_{\omega,\Delta^{max}}(assign) = (k + 1) \cdot \omega \cdot \Delta^{max}$.
2. *Let $\alpha \in \mathbb{R}^+$ be the length of the α-intervals. There exists a safe guard time for T wrt. Δ^{max} iff*
   $$\Delta^{max} < \frac{1}{4(k + 1)},$$
   the optimal safe guard time for T wrt. Δ^{max} is given by
   $$\phi_{opt} = \alpha \cdot \frac{2(k + 1) \cdot \Delta^{max}}{1 - 4(k + 1) \cdot \Delta^{max}}.$$

PROOF. Corollaries 3 and 4. ☐

6. DISCUSSION

For a scheduled and synchronized system over a network topology, the results indicate that the slot assignment affects the maximum clock drift in the network. For the worst case assignment, the maximum clock drift depends on the tree depth, where for the best case assignment, the maximum clock drift depends on the size of the maximal subtree. In the following, we show how our approach can be used and generalized.

6.1 Usage

With our results, one can compute a guard time given a network topology, α-interval, and a maximum drift rate Δ^{max}. If the slot assignment order can be controlled, e.g. because there is a proper procedure to set up the system, one may create an assignment order with small forward distances along paths to have a small optimal guard time. Otherwise, one needs to use the optimal guard time of the worst case assignment to guarantee safety for any assignment.

The results can also be used the other way round: Instead of optimizing guard time for a given assignment, one may choose a fixed guard time and restrict the assignment accordingly. For example, one can follow the best case assignment and restrict the size of the subtrees, e.g., by splitting big ones, to have a safe guard time as small as possible.

6.2 Arbitrary Initial Clock Values

The guard time computed in Theorem 2 is safe and optimal under the assumption that at time 0, all clocks have the value 0 for any evolution (cf. Definition 2). In order to generalize Theorem 2 for the cases where there exist bounded different values of clocks at time 0, the following approach can be used: By looking into the proof of Theorem 2, it is observed that the value of $\varrho^{max}_{\omega,\Delta^{max}}(assign)$ given by Corollary 2 is used to induce an optimal guard time. This value has to be modified by adding the the maximum clock drift that can be observed at time 0, and then, the equation of Theorem 2 is modified accordingly.

However, although the computed guard time in this case is safe, it is not optimal after synchronizing clocks. Therefore, in such cases, one should distinguish between the guard time length during system initialization (or setup), and the guard time length during system runtime. For example, during initialization, energy-consuming protocols can be run until the clocks are synchronized, and after initialization the optimal guard time is employed.

6.3 Message Loss

By definition, we assumed that each sensor is synchronized once per frame, within its assigned slot. In a real-world system, synchronization messages (or timestamps) may be lost due to, e.g., signal weakness. Consequently, a sensor's clock, that missed the synchronization, may continue to drift for a longer time than usual. To treat such cases using guard time, one has to specify the number of possible subsequent missed synchronization points for any sensor, and compute the additional clock drift that might be observed similar to the case in Section 6.2. In particular, considering the worst case assignment, if the number of subsequent missed synchronization points is m, the optimal guard time can be computed by considering a value of $d+m$ substituting d in Corollary 3.

6.4 Dynamic Slot Assignment

In our approach, we assume that the slot assignment is static. To treat a dynamic change of slot assignments during a system evolution, one has to consider the following: In general, a safe guard time need not exist if the assignment may change continuously. For instance, imagine a sensor whose assigned slot changes to be always re-positioned behind the slot as indicated by the reference time. Then the assigned sensor will never get its turn. On the other hand, if the assignment may change only once every long period, one can consider the additionally observed clock drift while computing guard time (similar to the previous section), and compute an optimal guard time accordingly.

7. CONCLUSION

We presented a powerful formal model of TDMA protocols used to schedule communication between network nodes which share a single communication medium, given synchronized networks with a tree topology and with a root which provides the reference time. We have shown that the slot assignment order affects the upper bound on clock drift values. We presented, based on exact topological characterizations of the assignments that yield the largest and least maximum clock drift, an approach to compute an optimal safe guard time which effectively avoids message collision and loss.

References

[1] T.S. Rappaport. *Wireless Communications: Principles and Practice*, volume 2. Prentice Hall, 2002.

[2] H Kopetz and G. Grünsteidl. TTP - A Time-Triggered Protocol for Fault-Tolerant Real-Time Systems. In *FTCS*, pages 524–533. IEEE Computer Society, 1993.

[3] FlexRay Consortium et al. Flexray Communications System. *Protocol Specification Version*, 2, 2005.

[4] J. Phillips and K. Kundert. An Introduction to Cyclostationary Noise. 2006.

[5] E. Perahia and R. Stacey. *Next Generation Wireless LANs: Throughput, Robustness, and Reliability in 802.11 n*, volume 28. Cambridge University Press, 2008.

[6] DIN e.V. Fire detection and fire alarm systems - Part 25: Components using radio links and system requirements, German version EN 54-25:2005, 2005.

[7] S.M. Lasassmeh and J.M. Conrad. Time Synchronization in Wireless Sensor Networks: A Survey. In *Proceedings of the IEEE SoutheastCon 2010*, pages 242–245. IEEE, 2010.

[8] D. L. Mills. Internet Time Synchronization: The Network Time Protocol. *IEEE Transactions on Communications*, 39(10):1482–1493, 1991.

[9] R. E. Burkard. Selected Topics on Assignment Problems. *Discrete Applied Mathematics*, 123(1-3):257–302, 2002.

[10] C.W. Commander and P.M. Pardalos. A Combinatorial Algorithm for the TDMA Message Scheduling Problem. *Computational Optimization and Applications*, 43(3):449–463, 2009.

[11] E. Wandeler and L. Thiele. Optimal TDMA Time Slot and Cycle Length Allocation for Hard Real-Time Systems. In *ASP-DAC*, pages 479–484. IEEE, 2006.

[12] S.S. Kulkarni and M. Arumugam. Transformations for Write-All-with-Collision Model. *Computer Communications*, 29(2):183–199, 2006.

APPENDIX

Proof of Theorem 1. The case where T has only one sensor is trivial. Thus in the following we assume that T has at least two sensors. Let $\mathcal{I}$ be a scheduled evolution over T with guard time ϕ. We consider each of the two points separately.

Point (1): Let $n \in Sn(T)$ be a sensor and let $t \in \mathsf{Time}$ be a point in time where n sends, i.e. where $send^{\mathcal{I}}_n(t) = 1$. Because $\mathcal{I}$ is scheduled and has guard time ϕ, there is a slot $[t_1, t_2)$ of n and points in time $t'_1, t'_2 \in \mathsf{Time}$ such that

$clk^{\mathcal{I}}_n(t'_1) = t_1 + \phi$ and $clk^{\mathcal{I}}_n(t'_2) = t_2 - \phi$ and $t'_1 \le t < t'_2$ because by Definition 9, n sends a message only if the clock of n denote a point within the α-interval, i.e., between $t_1 + \phi, t_2 - \phi$. By Definition 4, $clk^{\mathcal{I}}_n(t'_1) = clk^{\mathcal{I}}_{cu(T)}(t'_1) + \varrho^{\mathcal{I}}_n(t'_1)$ thus $t_1 + \phi = clk^{\mathcal{I}}_{cu(T)}(t'_1) + \varrho^{\mathcal{I}}_n(t'_1)$. Because $clk^{\mathcal{I}}_{cu(T)}(t'_1) = t'_1$ and using the premise we obtain $t_1 \le t_1 + \phi - \varrho^{\mathcal{I}}_n(t'_1) = t'_1$. Analogously we obtain $t'_2 \le t_2$, thus $t \in [t_1, t_2)$, which is the slot assigned to n by *assign*, and thus

$$\forall n \in N, t \in \mathsf{Time} \bullet send^{\mathcal{I}}_n(t) \implies slot(t) = assign(n). \quad (5)$$

We show that $\mathcal{I}$ doesn't have a message collision by contradiction. Assume $\mathcal{I}$ has a message collision, i.e. there is $t \in \mathsf{Time}$ where two different nodes $n_1, n_2 \in N$ send, i.e. where $send^{\mathcal{I}}_{n_1}(t) \wedge send^{\mathcal{I}}_{n_2}(t)$. By (5), we have $slot(t) = assign(n_1) \wedge slot(t) = assign(n_2)$ and thus $assign(n_1) = assign(n_2)$ which is a contradiction to *assign* being bijective.

Point (2): Let $n \in Sn(T)$ be a sensor with master n' and let $t \in \mathsf{Time}$ be a point in time where n sends, i.e. where $send^{\mathcal{I}}_n(t) = 1$. Applying the same reasoning as in the proof of point (1) above, we obtain $t \in \left[t_1 + \frac{\phi}{2}, t_2 - \frac{\phi}{2}\right)$ (6) where $[t_1, t_2)$ is an assigned slot to n. The master of n is listening throughout the slot of n. Because $\mathcal{I}$ is scheduled, there are points in time $t'_1, t'_2 \in \mathsf{Time}$ such that $clk^{\mathcal{I}}_{n'}(t'_1) = t_1$ and $clk^{\mathcal{I}}_{n'}(t'_2) = t_2$ and $\forall t' \in [t'_1, t'_2) \bullet listen^{\mathcal{I}}_{n'}(t')$ (7). By Definition 4, $clk^{\mathcal{I}}_{n'}(t'_1) = clk^{\mathcal{I}}_{cu(T)}(t'_1) + \varrho^{\mathcal{I}}_{n'}(t'_1)$ thus $t_1 = clk^{\mathcal{I}}_{cu(T)}(t'_1) + \varrho^{\mathcal{I}}_{n'}(t'_1)$. Because $clk^{\mathcal{I}}_{cu(T)}(t'_1) = t'_1$ and using the premise we obtain $t'_1 = t_1 - \varrho^{\mathcal{I}}_n(t'_1) \le t_1 + \frac{\phi}{2}$. Analogously we obtain $t'_2 \ge t_2 - \frac{\phi}{2}$. Thus, using (7), we have $\forall t' \in \left[t_1 + \frac{\phi}{2}, t_2 - \frac{\phi}{2}\right) \bullet listen^{\mathcal{I}}_{n'}(t')$. Thus, using (6), $listen^{\mathcal{I}}_{n'}(t)$. Thus there is no message loss at t.

Point (3): Let $n \in Sn(T)$ be a node and let $t_1 \in \mathsf{Time}$. Because $\mathcal{I}$ is scheduled, there is a latest slot $[t_1, t_2)$ with $t_1 \le t$ and identity (i, j). By (1) and (2), there is no message collision or loss in $\mathcal{I}$, thus, by Definition 8, there is a synchronization point t_0 in $[t_1, t_2)$. Distinguish two cases:

- $t < t_0$: Choose $t_0 = t'$. Then $t' - t \le \omega < (k+1) \cdot \omega$.
- $t \ge t_0$: Let t'_0 be the synchronization point in the slot $[t'_1, t'_2)$ with identity $(i, j+1)$ which is also assigned to n. t'_0 exists due to Definition 8. Choose t'_0 as t'. By Definition 6, $t'_1 = t_1 + k \cdot \omega$ and $t'_2 = t'_1 + \omega = (t_1 + k \cdot \omega) + \omega = t_1 + (k+1) \cdot \omega$. Thus $t' - t = t'_0 - t < t'_2 - t \le t'_2 - t_0 \le t'_2 - t_1 = (k+1) \cdot \omega$. $\square$

PROOF OF LEMMA 2. Let $T = (N, E)$ be a network topology with at least two sensors.

Point (1): Let $\mathcal{I}$ be a scheduled evolution $\mathcal{I}$ over T with slot length ω. Let $0 < \varepsilon < \omega$, n_1, n_2 be two nodes such that $assign(n_2) = assign(n_1) + 1$, $t \in \mathsf{Time}$ such that $t = t_2$ (8). $[t_1, t_2)$ is the slot of n_1, $\varrho^{\mathcal{I}}_{n_1}(t) = -(\phi + \varepsilon) \wedge \varrho^{\mathcal{I}}_{n_2}(t) = \phi$ (9), and both nodes send continuously during their α-interval. By Definition 4, $\varrho^{\mathcal{I}}_{n_1}(t) = clk^{\mathcal{I}}_{n_1}(t) - clk^{\mathcal{I}}_{cu(T)}(t)$. Thus, using premise (9) and (8), $clk^{\mathcal{I}}_{n_1}(t) = \varrho^{\mathcal{I}}_{n_1}(t) + clk^{\mathcal{I}}_{cu(T)}(t) = t_2 - (\phi + \varepsilon)$ which is in the α-interval of n_2. Similarly, we obtain $clk^{\mathcal{I}}_{n_2}(t) = \varrho^{\mathcal{I}}_{n_2}(t) + clk^{\mathcal{I}}_{cu(T)}(t) = t_2 + \phi$ which is in the α-interval of n_2. Thus, using the premise that n_1 and n_2 send continuously within their α-interval, $send^{\mathcal{I}}_{n_1}(t) \wedge send^{\mathcal{I}}_{n_2}(t)$ i.e. there exists message collision between n_1 and n_2 at t.

Point (2): Let $\mathcal{I}$ be a scheduled evolution $\mathcal{I}$ over T with slot length ω. let $0 < \varepsilon < \omega$, n be a node, $t \in \mathsf{Time}$ such that $t = t_2 - \frac{\phi}{2}$ (10). $[t_1, t_2)$ is the slot of n,

$$\varrho^{\mathcal{I}}_n(t) = -\left(\frac{\phi}{2} + \varepsilon\right) \wedge \varrho^{\mathcal{I}}_{ms(n)}(t) = \frac{\phi}{2}, \quad (11)$$

and n sends continuously during its α-interval. By Definition 4, $\varrho^{\mathcal{I}}_n(t) = clk^{\mathcal{I}}_n(t) - clk^{\mathcal{I}}_{cu(T)}(t)$. Thus, using premise (11) and (10), $clk^{\mathcal{I}}_n(t) = \varrho^{\mathcal{I}}_n(t) + clk^{\mathcal{I}}_{cu(T)}(t) = t_2 - \frac{\phi}{2} - \left(\frac{\phi}{2} + \varepsilon\right) = t_2 - \phi - \varepsilon$ which is in the α-interval of n, Thus $send^{\mathcal{I}}_n(t) = 1$ (12). Similarly, we obtain $clk^{\mathcal{I}}_{ms(n)}(t) = \varrho^{\mathcal{I}}_{ms(n)}(t) + clk^{\mathcal{I}}_{cu(T)}(t) = t_2 - \frac{\phi}{2} + \frac{\phi}{2} = t_2$ which is not in the slot assigned to n, thus $listen^{\mathcal{I}}_{ms(n)}(t) = 0$ i.e. there exists message loss at t. $\square$

PROOF OF LEMMA 3. By induction over the depth of nodes d. For the base case and the step, we first show "$\le$" and then "$\ge$" to obtain (1).

Base case $d = 1$: Let n be a node with $depth(n) = 1$. Let $\mathcal{I}$ be a synchronized evolution over T with slot length ω and least upper bound Δ^{max} on the clock drift rates, and let $\mathcal{I}$ be scheduled by assignment *assign*. Let $t_2 \in \mathsf{Time}$ be a point in time. By Definition 8, there is a synchronization point $t_1 \le t_2$ such that $clk^{\mathcal{I}}_n(\cdot)$ is differentiable on the interval (t_1, t_2). By Note 1, $\varrho^{\mathcal{I}}_n(t_2) = \varrho^{\mathcal{I}}_n(t_1) + \int_{t_1}^{t_2} \Delta^{\mathcal{I}}_n(t)dt$ (13). Because $depth(n) = 1$, the master of n is the central unit and t_1 is a synchronization point, thus by Definitions 4 and 8, $\varrho^{\mathcal{I}}_n(t_1) = clk^{\mathcal{I}}_n(t_1) - clk^{\mathcal{I}}_{cu(T)}(t_1) = 0$, thus $\varrho^{\mathcal{I}}_n(t_2) = \int_{t_1}^{t_2} \Delta^{\mathcal{I}}_n(t)dt$ (14). Because Δ^{max} is an upper bound on the clock drift rates in $\mathcal{I}$, $\varrho^{\mathcal{I}}_n(t_2) \le (t_2 - t_1)\Delta^{max}$, thus by Lemma 1, $\varrho^{\mathcal{I}}_n(t_2) \le (k+1) \cdot \omega\Delta^{max}$.

Let $n \in N$ be a node in the set of synchronized evolutions which are scheduled by *assign*, have slot length ω, and for which Δ^{max} is the least upper bound on the clock drift rates. There is an evolution $\mathcal{I}$ with $\Delta^{\mathcal{I}}_n(t) = \Delta^{max}$ and with two synchronization points $t_1, t_2 \in \mathsf{Time}$ such that $t_2 - t_1 = (k+1) \cdot \omega$ and such that $\Delta^{\mathcal{I}}_n$ is differentiable on (t_1, t_2). For this evolution, (14) applies and yields $\varrho^{\mathcal{I}}_n(t_2) = (t_2 - t_1)\Delta^{max} = (k+1) \cdot \omega \cdot \Delta^{max}$, thus $\varrho^{\mathcal{I}}_n(t_2) \ge (k+1) \cdot \omega\Delta^{max}$.

Induction step $d \to d+1$: Assume that 1 holds for all nodes of depth up to d. Let n be a node of depth $d+1$. Let $\mathcal{I}$ be a synchronized evolution over T with slot length ω and least upper bound Δ^{max} on the clock drift rates, and let $\mathcal{I}$ be scheduled by assignment *assign*. Let $t_2 \in \mathsf{Time}$ be a point in time. By Definition 8, there is a synchronization point $t_1 \le t_2$ such that $clk^{\mathcal{I}}_n(\cdot)$ is differentiable on the interval (t_1, t_2). By Note 1, $\varrho^{\mathcal{I}}_n(t_2) = \varrho^{\mathcal{I}}_{ms(n)}(t_1) + \int_{t_1}^{t_2} \Delta^{\mathcal{I}}_n(t)dt$ (15). The master $ms(n)$ of n has depth d, thus by induction hypothesis,

$$\varrho^{\mathcal{I}}_{ms(n)}(t_1) \le \left(\sum_{i=1}^{d-1} fdist_{assign}(n_i, n_{i+1}) + k + 1\right) \cdot \omega\Delta^{max}. \quad (16)$$

Because Δ^{max} is an upper bound on the drift rates in $\mathcal{I}$, $\int_{t_1}^{t_2} \Delta^{\mathcal{I}}_n(t)dt \le fdist(ms(n), n) \cdot \omega\Delta^{max}$ (17), thus, by (16) and (17),

$$\varrho^{\mathcal{I}}_n(t_2) \le \left(\sum_{i=1}^{(d+1)-1} fdist_{assign}(n_i, n_{i+1}) + k + 1\right) \cdot \omega\Delta^{max}.$$

Let $n \in N$ be a node of depth $d+1$ in the set of synchronized evolutions which are scheduled by *assign*, have slot length ω, and for which Δ^{max} is the least upper bound on the clock drift rates there is (by implicit induction hypothesis) an evolution $\mathcal{I}$ with

$$\Delta^{\mathcal{I}}_{ms(n)}(t) = \left(\sum_{i=1}^{d-1} fdist_{assign}(n_i, n_{i+1}) + k + 1 \right) \cdot \omega \Delta^{max}. \tag{18}$$

Because $ms(n)$ has depth d, and with $\Delta^{\mathcal{I}}_n(t) = \Delta^{max}$ and with two synchronization points $t_1, t_2 \in \mathsf{Time}$ such that $t_2 - t_1 = fdist(ms(n), n) \cdot \omega$ (also by implicit induction hypothesis: the node which satisfies (18) is synchronized as late as possible, i.e. at the right end of its slot), for this evolution, (14) applies and yields $\varrho^{\mathcal{I}}_n(t_2) = (t_2 - t_1)\Delta^{max} = fdist(ms(n), n) \cdot \omega \cdot \Delta^{max}$. Thus

$$\varrho^{\mathcal{I}}_n(t_2) \geq \left(\sum_{i=1}^{(d+1)-1} fdist_{assign}(n_i, n_{i+1}) + k + 1 \right) \cdot \omega \cdot \Delta^{max}.$$

$\square$

Proof of Theorem 2. We're looking for a safe guard time for T wrt. Δ^{max}, thus (by Definition 11) we are looking for $\phi \in \mathbb{R}^+_0$ such that each synchronized system Ω over T which is scheduled by an assignment *assign* and which has maximum drift rate Δ^{max} and guard time ϕ does not exhibit message collision or message loss.

Let T be a network topology of depth d with $k \in \mathbb{N}^+$ sensors, and upper bound $\Delta^{max} \in \mathbb{R}^+_0$ on clock drift rates. Let $\alpha \in \mathbb{R}^+$ be the length of the α-intervals, and *assign* be an assignment of slots to nodes. By Corollary 2,

$$\varrho^{max}_{\omega,\Delta^{max}}(assign) \leq (D_{assign} + k + 1) \cdot \omega \cdot \Delta^{max}.$$

By Theorem 1 and Lemma 2, a sufficient and necessary criterion for $\phi \in \mathsf{Time}$ being a safe guard time is that for all evolutions $\mathcal{I} \in \Omega$, $\forall n \in N, t \in \mathsf{Time} \bullet |\varrho^{\mathcal{I}}_n(t)| \leq \frac{\phi}{2}$ (19). By Definition 9, $\omega = 2\phi + \alpha$, thus ϕ is safe if $(D_{assign} + k + 1) \cdot (2\phi + \alpha) \cdot \Delta^{max} \leq \frac{\phi}{2}$ (20). Simplification (and reordering) yields

$$\Delta^{max} \cdot (D_{assign} + k + 1) \cdot (2\phi + \alpha) \cdot \leq \frac{\phi}{2} \tag{21}$$

$$\iff 2 \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \cdot (2\phi + \alpha) \leq \phi \tag{22}$$

$$\iff 4\phi \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \tag{23}$$

$$+ 2 \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \cdot \alpha \leq \phi \tag{24}$$

$$\iff 2 \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \cdot \alpha \tag{25}$$

$$\leq (1 - 4 \cdot \Delta^{max} \cdot (D_{assign} + k + 1)) \cdot \phi. \tag{26}$$

Distinguish two cases: Case $\Delta^{max} = 0$: Equation (26) has the unique solution $\phi = 0$. Case: $\Delta^{max} > 0$: There only is a solution if $(1 - 4 \cdot \Delta^{max} \cdot (D_{assign} + k + 1)) \neq 0$ (27), otherwise the intended division is not defined. Distinguish two cases:

- For $(1 - 4 \cdot \Delta^{max} \cdot (D_{assign} + k + 1)) > 0$ $(*)$, (26) yields

$$\frac{2 \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \cdot \alpha}{1 - 4 \cdot \Delta^{max} \cdot (D_{assign} + k + 1)} \leq \phi \tag{28}$$

  by $(*)$ and $D_{assign} > 0$, $\Delta^{max} > 0$, $d > 0$, $k > 0$. The smallest choice of ϕ satisfying inequation (28) is

$$\alpha \cdot \frac{2 \cdot (D_{assign} + k + 1) \cdot \Delta^{max}}{1 - 4 \cdot (D_{assign} + k + 1) \cdot \Delta^{max}} \tag{29}$$

  which is greater or equal to 0, thus a proper guard time. $(*)$ yields the claimed bound on Δ^{max}.

- For $(1 - 4 \cdot \Delta^{max} \cdot (D_{assign} + k + 1)) < 0$, (26) yields

$$\phi \leq \frac{2 \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \cdot \alpha}{1 - 4 \cdot \Delta^{max} \cdot (D_{assign} + k + 1)} \tag{30}$$

  (because we divide both sides by a negative number). We have $2 \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \cdot \alpha > 0$, because $\Delta^{max} > 0$, $d > 0$, $k > 0$, $\alpha > 0$, thus

$$\frac{2 \cdot \Delta^{max} \cdot (D_{assign} + k + 1) \cdot \alpha}{1 - 4 \cdot \Delta^{max} \cdot (D_{assign} + k + 1)} < 0 \tag{31}$$

  thus in this case there does not exists a proper (positive) guard time. $\square$

Proof of Lemma 4. For network topologies of depth 1, both claims hold trivially. Let T be a network topology of depth $d \geq 2$. Regarding Point (1): Let *assign* be an assignment. Given that *assign* is bijective, by Definition 13, we have for each two different sensors $n, n' \in Sn(T)$, that $fdist_{assign}(n, n') \leq k - 1$ (32), thus for any path $n_0, ..., n_m$ in T, where $m \leq d$, $\sum_{i=1}^{m-1} fdist_{assign}(n_i, n_{i+1}) \leq (d-1)(k-1)$. (33)

Regarding Point (2): "$\Longleftarrow$": Let *assign* have a (2)-path, i.e. the slaves on the path are synchronized immediately before their master. By (2) and Definition 13, $fdist_{assign}(n_i, n_{i+1}) = k - 1$ (34). Thus $\sum_{i=1}^{d-1} fdist_{assign}(n_i, n_{i+1}) = (d-1)(k-1)$ for the (2)-path. Thus, with (33), $D_{assign} = (d-1)(k-1)$.

"$\Longrightarrow$": Let *assign* be an assignment with $D_{assign} = (d-1)(k-1)$. by Definition 13 and by (32), there is a path $n_0, n_1, \ldots, n_d$, $n_0 = cu(T)$ where $fdist_{assign}(n_i, n_{i+1}) = (k-1)$. Thus, by Definition 13, there is a (2)-path. $\square$

Proof of Lemma 5. For network topologies of depth 1, the two claims hold trivially. Let T be a network topology of depth $d \geq 2$. First we show that if conditions (2a) and (2b) are satisfied, then $D_{assign} = K - 1$. Second, we show that if any of (2a) and (2b) is violated, then $D_{assign} > K - 1$.

Let *assign* be an assignment of slots to nodes such that conditions (2a) and (2b) are satisfied. Let $n_1, ..., n_m$ with $(cu(T), n_1) \in E$ be a path in T. By (2a), for each $1 < j \leq m$: $fdist_{assign}(n_1, n_j) \leq (K-1)$ (35). By (2b), for each $1 \leq i < m - 1$: $fdist_{assign}(n_i, n_{i+2}) = fdist_{assign}(n_i, n_{i+1}) + fdist_{assign}(n_{i+1}, n_{i+2})$ (36). By (35) and (36): $\sum_{i=1}^{j-1} fdist_{assign}(n_i, n_{i+1}) = fdist_{assign}(n_1, n_j) \leq (K-1)$ (37). Let n_m be a sensor in a maximal subtree of T which is assigned the (modulo k) latest slot (this sensor must exist by Definition of subtree). Then $fdist_{assign}(n_1, n_m) = K - 1$ (38). Thus, by (37) and (38), $D_{assign} = K - 1$.

We show that, if (2a) or (2b) are violated, then $D_{assign} > K - 1$. (i) (2a) is violated. Let *assign*$'$ be an assignment such that (2a) is violated. Then there exists a path $n_0, ..., n_m$, where $n_0 = cu(T)$ and $m \leq d$, such that $fdist_{assign'}(n_1, n_m) > (K-1)$ (39). Thus by Definition 13, $D_{assign'} > (K-1)$. (ii) (2b) is violated. Let *assign*$''$ be an assignment such that (2b) is violated. Then there exist three nodes $n, n', n'' \in N$ such that $(n, n'), (n', n'') \in E$ and $fdist_{assign''}(n, n') \geq fdist_{assign''}(n, n'')$. By Definition 7, *assign* is bijective, and therefore $fdist_{assign''}(n, n') \neq fdist_{assign''}(n, n'')$. By Definition 13 and the order of n, n', n'', $fdist_{assign''}(n, n') + fdist_{assign''}(n', n'') > k$ (40). As n, n', n'' lie on a path in T, $D_{assign} > k > (K-1)$. $\square$