FWF-Proposal ADynNet: Gracefully Degrading Agreement in Directed Dynamic Networks*

ULRICH SCHMID
Technische Universität Wien, Institut für Technische Informatik E182/2,
Treitlstraße 1–3, A-1040 Vienna (Austria)
s@ecs.tuwien.ac.at

December 18, 2014

1 Scientific Aspects

1.1 Introduction

Dynamic networks, instantiated, e.g., by wireless sensor networks, mobile ad-hoc networks and vehicle area networks as well as by peer-to-peer and even social networks, are becoming ubiquitous nowadays. The primary properties of such networks are (i) sets of participants (called processes in the sequel) that are a priori unknown and potentially time-varying, (ii) time-varying connectivity between processes, and (iii) the absence of central control. Such assumptions make it already very difficult to setup and maintain the basic communication system, and create particular challenges for the design of robust distributed services for applications running on such dynamic networks.

Accurately modeling communication in dynamic networks is challenging:

(i) Communication in many dynamic networks, in particular, in wireless networks like *mobile ad-hoc networks* (MANETS) [71], is inherently broadcast: When a process transmits, then every other process within its transmission range will observe this transmission — either by legitimately receiving the message or as some form of interference. This creates quite irregular communication behavior, such as capture effects and near-far problems [114], where certain (nearby) transmitters may "lock" a receiver and thus prohibit the reception of messages from other senders. Consequently, the "health" of a wireless link between two processes may vary heavily over time [36]. For low-bandwidth wireless transceivers, an acceptable link quality usually even requires communication scheduling [96] (e.g., time-slotted communication) for reducing the mutual interference. Overall, this results in a frequently changing spatial distribution of pairs of nodes that can communicate at a given point in time. Classic distributed systems theory and practice, which usually assumes a more or less fixed communication network, are hence of very limited utility here.

(ii) Process mobility, process crashes/recoveries, deliberate joins/leaves, and peculiarities in the low-level system design like duty-cycling (typically used to save energy in wireless sensor networks) make static communication topologies, as typically used in classic network models, inadequate for dynamic networks. Certain instances of dynamic networks, in particular, peer-to-peer networks [77]

*This is a revised version of the ADynNet proposal submitted in September 2014.

and inter-vehicle area networks [50], even suffer from significant churn, i.e., a large number of processes that can appear/disappear over time, possibly in the presence of faulty processes [6]: In sharp contrast to more classic dynamic networks like MANETS [71], sensor networks [5, 117] and even disaster relief applications [78], which have a *bounded* number of participants (even though this bound is usually unknown), they can have an *unbounded* total number of processes over time.

Consequently, assuming bidirectional links is not the right abstraction for many dynamic networks, in particular, in wireless ones [35]: Fading and interference phenomenons [60, 104], including capture effects and near-far problems, are *local* effects that affect only the receiver of a wireless link. Given that the sender, which is also the receiver of the reverse link, resides at a different location, the two receivers are likely to experience very different levels of fading and interference [54]. This effect is even more pronounced in the case of time-slotted communication, where forward and backward links are used at different times. Consequently, the existence of asymmetric communication links cannot be ruled out in practice: According to [91], 80% of the links in a typical wireless network are asymmetric.

Despite this fact, most of the dynamic network research we are aware of assumes bidirectional links [74, 76]. The obvious advantage of this paradigm is the relative simplicity of the algorithm design, as strong communication guarantees obviously make this task easier. Moreover, it allows the re-use of existing techniques for wireline networks, which naturally support bidirectional communication. However, there are also some major disadvantages of this convenient abstraction:

(1) For dynamic networks that operate in environments with unfavourable communication conditions, e.g. in disaster relief applications or, more generally, in settings with various interferers and obstacles that severely inhibit communication, bidirectional links may simply not be achievable. If we want to implement distributed services even in such settings, algorithms that do not need bidirectional links are mandatory.

(2) The entire system needs to be engineered in such a way that bidirectional single-hop communication can be provided within bounded time. This typically requires relatively dense networks and/or processes that are equipped with powerful communication interfaces, which incur significant cost when compared to sparser networks or/and cheaper or more energy-saving communication modules.

(3) If bounded bidirectional single-hop communication is guaranteed at the physical layer, a bidirectional link abstraction is easily created atop, by just letting a process wait long enough to receive all (non-lost) messages from its neighbors. If, however, directed single-hop communication was already sufficient to reach the desired goal (say, reaching some destination process) via multi-hop propagation, waiting for guaranteed single-hop bidirectional communication incurs a potentially significant, unnecessary delay. Obviously, in such settings, algorithmic solutions that do not need bidirectional single-hop communication could be significantly faster.

Hence, the first distinguishing property of ADynNet is that we explicitly address dynamic networks with *unidirectional* communication links, which have only rarely been addressed in the past. Note that this class of dynamic networks also covers bidirectional ones, as a bidirectional link can be modeled as a pair of unidirectional links.

The "traditional" approach for developing algorithmic solutions for such dynamic networks is to start out from assumed or known properties of existing networks, to develop candidate algorithms, and to evaluate those experimentally [41] or by means of simulations (based on suitable network models) [91].

By contrast, we employ a radically different approach, which constitutes the second distinguishing property of ADynNet: We start out from a simple abstract model of dynamic networks based on time-varying communication graphs, which covers a reasonably large class of applications. For a given problem to be solved (say, distributed consensus), we ask for weak (ideally: weakest) *network assumptions* (say, always strongly connected) that are to be guaranteed by the communication graphs (Task 1, Sec. 1.5.1). For every such network assumption, we develop solution algorithms, prove them correct and analyze their performance (Task 2, Sec. 1.5.2). Finally, we determine its assumption coverage, i.e., a measure that quantifies how likely it is that some particular network assumption holds in a given network (Task 3, Sec. 1.5.3). Note that this is the only task where simulations and experiments can make sense in ADynNet.

The third distinguishing property of ADynNet is its primay focus on distributed *agreement problems*. In fact, what appears to be a quite natural idea for building *robust* services in such networks is to employ distributed consensus to agree system-wide on (fundamental) system parameters like schedules, frequencies, and operating modes, as well as for agreeing on application-level issues: Such a solution would facilitate arbitrary (possibly centralized) algorithms for generating local proposals, which are supplied as inputs to a consensus algorithm that finally selects one of them consistently at all processes. Unlike in master-slave-based solutions, this approach would avoid the single point of failure formed by the process acting as the master.

Unfortunately, however, in larger-scale dynamic networks, implementing consensus is at best very difficult (and typically impossible), for several reasons:

(a) Solving deterministic (always safe) consensus requires communication graphs that are well-connected system-wide, for a sufficiently long period of time [21]. Network partitioning into multiple (partially connected) components cannot be ruled out in dynamic networks, however.

(b) Processes in dynamic networks typically know their "communication-active" neighborhood only. Consequently, they cannot be assumed to have a priori global information, like the number of processes in the system. It is usually even impossible to acquire complete and accurate local knowledge of the entire system at run-time, due to link/node unavailabilities, network partitioning, insufficient local memory, etc.

(c) Termination times can be large and are not necessarily bounded a priori, which makes it difficult for applications to (repeatedly) use consensus for (repeated) decision making: At some given time, different processes could rely on decisions from different instances of repeated consensus.

ADynNet will hence explore the solvability of *weaker* forms of distributed agreement in directed dynamic networks, in particular, gracefully degrading consensus and approximate agreement. Only very few results on this topic exist yet.

1.2 Project Overview

In ADynNet, we restrict our attention to a network model that comprises an *unknown but bounded* number of processes, which are interconnected by *directed* communication links. The system is assumed to be synchronous,[1] hence time is measured in discrete *rounds* that allow the processes to exchange at most one

[1] As synchronized clocks are typically required for basic communication in wireless systems anyway, e.g., for transmission scheduling and sender/receiver synchronization, this is not an unrealistic assumption: Global synchrony can be implemented directly at low system levels, e.g., via IEEE 1588 network time synchronization or GPS receivers, or at higher levels via time

message. Time-varying communication is modeled as a sequence of *communication graphs*, which contain a directed edge between two processes if the message sent in the corresponding round is successfully received. A bidirectional link is modeled by a pair of directed links that are considered independent of each other here.

We will develop theoretical foundations, network assumptions and solution algorithms, along with correctness proofs, performance and coverage analyses, for "relaxed" distributed agreement problems, in particular, gracefully degrading variants of k-set agreement [39] and approximate agreement [45], in directed dynamic networks: k-set agreement allows the processes to decide on one of at most k different values system-wide ($k = 1$ is equivalent to consensus), whereas approximate agreement allows decision values that deviate by some ε from each other. ADynNet shall yield both insights into the fundamental limitations and novel algorithms that will prove useful for solving distributed agreement reliably even under very weak communication guarantees. Somewhat surprisingly, and in sharp contrast to *undirected* dynamic networks [76], not much is known here in the case of directed dynamic networks.

Albeit we do not focus on a particular application domain in ADynNet, which is an FWF project and hence entirely devoted to basic research, we conjecture that our results may prove particularly useful for dependable applications that are to be deployed in environments suffering from highly time-variant but relatively sparse communication graphs connecting a bounded number of processes: In sharp contrast to densely populated sensor networks, for example, where the main problem is *not* to use all the available links (i.e., to exercise topology control for constructing a typically tree-like "overlay network" [5]), sparse ad-hoc networks as found e.g. in typical MANETS [71] and disaster relief applications [57, 78, 84] need to use all available links (sometimes even using tricks like "message ferrying" via mobile nodes [119]). Distributed decision making is also more likely to be useful in such sparse networks, both at the network level (e.g., agreement on communication schedules [96]) and at the application level (e.g., agreement on rescue team membership [57]). Note that we deliberately avoid dynamic networks with significant churn in ADynNet, as this would involve entirely different distributed problems, models and algorithmic solutions: It does not makes sense to also target the latter in the scope of a (necessarily focused) single 3-year project like ADynNet, which is already quite ambitious.

Project details. Partitioned into three reasonably decoupled tasks (Task 1–3), ADynNet will comprise research on the following major goals:

(1) We will model dynamic systems by means of dynamically changing *directed* communication graphs with a bounded but unknown number of processes, which may also fail.

The key issue here is to find weak *network assumptions*, i.e., connectivity assumptions for the communication graphs, which are just strong enough to allow a certain distributed agreement problem to be solved. For example, we showed in [21] that consensus cannot be solved even if, at any time, the communication graph contains some rooted tree. Finding or at least approaching *weakest* network assumptions (which are necessary and sufficient) is not only of theoretical interest,[2] but also a mandatory prerequisite for k-set agreement algorithms that truly degrade *gracefully*: After all, such

synchronization protocols like FTSP [82] or even synchronizers [8]. As synchronization errors cannot be ruled out a priori, however, the need for modeling such process failures may arise in dependable applications.

[2] In the course of this work, we will also try to contribute to the chase for the weakest failure detector for message-passing k-set agreement [28], which is a long-standing problem in distributed algorithms: Raynal and Stainer [99] revealed that there are relations between this classic model and dynamic networks, and some of our recent research provided us with some tools ("easy impossibility proofs" [19], which are not based on algebraic topology [67]) that might allow us to approach this problem from a new angle.

an algorithm should solve k-set agreement with the *smallest* k possible in a given execution. Last but not least, weak network assumptions are also pivotal for maximizing the assumption coverage in real systems (see (3) below).

(2) We will primarily develop algorithmic solutions for relaxed forms of distributed agreement that can be considered *gracefully degrading* variants of consensus. Although we will primarily focus on deterministic algorithms, we will also look at randomized algorithms that are always safe ("Las Vegas"), as well as explore techniques for coping with synchronization errors. A main focus will be on mathematical correctness proofs and performance analyses of such algorithms, under the given network assumptions.

Essentially, there are three ways for relaxing consensus: (i) Dropping the bounded termination time requirement, which leads to asymptotic consensus, (ii) relaxing the agreement requirement such that the decision of two processes can be at most ϵ apart, which leads to approximate agreement, and (iii) dropping the requirement of achieving the same decision value system-wide, which naturally leads to k-set agreement. Asymptotic consensus [13] and approximate agreement [10], which have already been studied in certain dynamic networks [13], will provide a "baseline" regarding solvability, as these problems require only very weak network assumptions. On the other hand, k-set agreement, which has not been considered in dynamic networks research before, opens up a wide range of research questions which are interesting both from a theoretical and a practical perspective: First, k-set agreement has been a major target for the study of solvability in classic asynchronous distributed systems with and without failure detectors since decades. It is hence a very suitable problem for studying the fundamental solvability of agreement problems in dynamic networks as well. Second, from a practical point of view, k-set agreement algorithms are a natural extension of consensus for partitionable systems.

(3) Given a certain network assumption, its *assumption coverage* is the probability that it will not be violated in a given dynamic network. Essentially, it bridges the gap between the adversarial world of our network assumptions and the non-adversarial (typically, probabilistic) world of real systems. One, to the best of our knowledge, distinguishing features of ADynNet are efforts devoted to the analysis of the assumption coverage of the various network assumptions to be developed in the course of the project. Our primary "tool" here will be analytic approaches based on analytic combinatorics, albeit supporting simulations based on random graph models [104] will also be employed.

External collaborations. Leading international experts will also contribute to the work in ADynNet, by providing specific expertise: Yoram Moses (Technion Haifa) and Calvin Newport (Georgetown University) will help us to characterize necessary and sufficient network assumptions, Martin Biely (EPFL) and Christian Scheideler (University of Paderborn) will contribute to the design and analysis of fault-tolerant algorithms, and Peter Robinson (National University of Singapore) will bring in expertise on randomized algorithms.

Another very promising external collaboration has recently been setup with Christian Bettstetter (Lakeside Labs Klagenfurt), who will support our work in ADynNet by providing a realistic random graph model [104] as a starting point. Another recently established external collaboration with Kay Römer (TU-Graz) will not only allow us to incorporate systems engineering expertise for guiding the

problems/constraints studied in ADynNet, but will also be used for setting up a joint follow-up project: The latter will be devoted to a thorough experimental evaluation of suitably improved and engineered extensions of the models and algorithms to be developed in ADynNet.

In summary, ADynNet will explore a significant, uncharted territory of distributed computing research: It will investigate agreement services that adapt to conditions of varying favorability, and aims at precisely capturing the fundamental properties required for implementing those in directed dynamic networks with an unknown number of processes. We believe that our approach is a promising way to alleviate the disadvantages of classic bidirectional link abstractions. Moreover, it will facilitate algorithmic solution that work under very weak network assumptions, including ones that hold only eventually, which is interesting not only from the practical but also from the theoretical perspective.

Last but not least, thanks to our initial results [21, 23, 110], our external collaborations, and the large body of existing research devoted to related topics in distributed algorithms, we are convinced that these goals are achievable within the proposed project.

1.3 State of the Art

In this section, we will provide an overview[3] of the existing work related to ADynNet; our own related work will be provided in Section 1.3.5.

Dynamic networks have been studied intensively in distributed computing. Early work on this topic includes [3, 9]; an overview can be found in [75] (and the references therein). One basic assumption that can be used to categorize research in dynamic networks is whether the set of processes is assumed to be fixed, or subject to churn (i.e., a substantial number of processes that join and leave over time). The latter has mostly been considered in the area of peer-to-peer networks and the construction of overlays; the interested reader is referred to [77] for an overview of the research in this area.

When the set of processes is considered to be fixed, as is the scope of ADynNet, dynamicity in the network is modeled by changes in the network topology. Several approaches to modeling dynamic connectivity in networks have been proposed in the past.

1.3.1 Time-varying graph models

There is a rich body of literature on dynamic graph models going back to [63], which also mentions for the first time modeling a dynamic graph as a sequence of static graphs. Casteigts et al. [33] introduced a classification of the assumptions on the temporal properties of time varying graphs.

Whereas the focus of [74] resp. [62] is on the complexity of aggregation problems resp. information dissemination in dynamics networks, [7, 40, 76] focus on agreement problems: The work by Kuhn, Oshman and Moses [76] is devoted to the Δ-coordinated consensus problem, which extends consensus by requiring all processes to decide within Δ rounds of the first decision. [7] studies randomized algorithms for stable almost-everywhere agreement (a variant of the almost everywhere agreement problem introduced in [49]), which weakens the classic consensus problem in the sense that a small linear fraction of processes may remain undecided. The leader election problem in dynamic networks, which is also related to consensus, has been studied in [40].

[3] Given the fact that ADynNet lies at the heart of several hot topics in research, it is impossible to exhaustively survey the large body of existing results. Whereas we tried to list representative references for the most important research directions, we are well aware that, unfortunately, our selection necessarily discriminates all the other important papers in the respective field.

All the above work uses models that can be viewed as variants of the model introduced in [74], where distributed computations are organized in lock-step synchronous rounds. Communication is described by a sequence of per-round communication graphs, which must adhere to certain network assumptions (like T-interval connectivity, which says that there is a common connected subgraph in any interval of T rounds): In the context of agreement problems, to the best of our knowledge, only *undirected* graphs that are *connected in every round* have been considered. In terms of the classes of [32], the model of [74] is hence in one of the strongest classes (Class 10), in which every process is always reachable by every other process. Since node failures are not considered, solving consensus is possible in this model.

By contrast, the work in ADynNet will be based on *directed* communication graphs, as it has already been used in some general dynamic networking research like [73]. In sharp contrast to this work, however, we consider agreement under *weak* connectivity assumptions that do not guarantee bidirectional (multi-hop) communication between all processes. The model used in our initial work [21, 110] thus belongs to the weakest class of models in [32]. Whereas this is beneficial in terms of the assumption coverage, it comes at the price of considerably increased communication and memory complexity.

1.3.2 Transmission failure models

Instead of considering a time-varying graph that defines which processes can communicate when, an alternative approach is based on the (dual) idea of assuming a fully connected network of (potential) communication, and considering that communication/message transmission in a round can be corrupted or fail outright. The notion of transmission failures was introduced by Santoro and Widmayer [102], who assumed dynamic transmission failures and showed that $n-1$ dynamic transmission failures in the benign case (or $n/2$ in case of arbitrary transmission failures) render any non-trivial agreement impossible. As it assumes unrestricted transmission failures (the (only) case considered in their proof are failures that affect all the transmissions of a *single* process), it does not apply to any model which considers perpetual mutual reachability of processes (e.g., [76]).

The HO-model [38] is also based on transmission failures. It relies on the collection of sets of processes a process *hears of* (i.e., receives a message from) in a round. Different system assumptions are modeled by predicates over this collection of sets. The HO-model is totally oblivious to the actual reason *why* some process does not hear from another one: It does not discriminate whether the sender committed a send omission or crashed, the message was lost during transmission or is simply late, or the receiver committed a receive omission. A version of the HO-model that also allows communication to be corrupted is presented in [17]. Indeed, the HO-model is very close to our graph model, as an edge from p to q in the graph of round r corresponds to p being in the round r heard-of set of q.

The approach taken by Gafni [55] bears some similarities with the HO-model (of which it is a predecessor), but is more focused on process failures than the two approaches above. Here an oracle (a round-by-round failure detector) is considered to tell processes the set of processes they will not be able to receive data from in the current round. Unlike the approaches discussed above, it explicitly states how rounds are implemented; nevertheless, the oracle abstracts away the actual reason for not receiving a message. So, like in the HO-model, the same device is used to describe failures and (a)synchrony.

Another related model is our perception based failure model [24, 108], which uses a sequence of perception matrices (corresponding to heard-of sets) to express failures of processes and links. As for transmission failures, the impossibility result of Santoro and Widmayer is circumvented by putting sep-

arate restrictions on the number of outgoing and incoming links that can be affected by transmission failures [108]. Since transmission failures are counted on a per process/per round basis, agreement turned out to be possible in the presence of $O(n^2)$ total transmission failures per round.

Finally, [2] considered restricted adversaries such that problems solvable in wait-free read-write shared memory systems remain solvable in message-passing systems; [99] looks at the relationship between round-based models models and failure detectors. All these approaches consider a static set of nodes, however.

1.3.3 Degradable agreement

Approximate agreement, which can be considered a degradable form of consensus, is a well-studied problem in classic synchronous [10, 45] and asynchronous [11, 52] distributed systems, which has also been studied under hybrid process and communication failures and not fully-connected networks. Approximate agreement and variants like inexact agreement [80] have several applications in distributed systems, ranging from clock synchronization [115] to sensor fusion [31].

Asymptotic consensus, i.e., consensus without termination, has partially been studied in dynamic networks. State transitions of the algorithms are typically expressed as the matrix product of the vector of the local values of the processes $v(t)$ at time t and a (stochastic) matrix $\mathbf{A}(t)$ that captures both the network connectivity at t and the averaging rule used for computing a new local value. Using well-known results from linear algebra, [13] proved the convergence of the local values (and hence approximate agreement) in the case of a fixed matrix $\mathbf{A}(t)$, and for restricted classes of time-varying matrices (in particular, bidirectional communication links) [64, 85]. Hendricks, Olshevsky and Tsitsiklis [65, 66] introduced a synchronous model that also captures non-linear algorithms, albeit restricted to fixed communication graphs.

By contrast, we are not aware of any existing work exploring other forms of gracefully degrading agreement. However, there have been several attempts to weaken the semantics of consensus, in order to cope with partitionable systems and excessive faults. Vaidya and Pradhan introduced the notion of *degradable* agreement [112], where processes are allowed to also decide on a (fixed) default value in case of excessive faults. The *almost everywhere agreement* problem introduced by [49] allows a small linear fraction of processes to remain undecided. Aguilera et. al. [4] considered quiescent consensus in partitionable systems, which requires termination only from processes of the majority partition.

Regarding k-set agreement in dynamic networks (not to speak of a k-uniform one), we are not aware of any existing work except our earlier paper [20]. Ingram et. al. [69] presented an asynchronous leader election algorithm for dynamic systems, where every component is guaranteed to elect a leader of its own. Whereas this behavior clearly matches our definition of graceful degradation, contrary to decisions, leader assignments are revocable and the algorithm of [69] is guaranteed to successfully elect leader(s) only if the topology eventually stabilizes.

Since we hope to be able to contribute to the long-standing chase for the weakest failure detector for message-passing k-set agreement [98] as well, we also briefly summarize the state-of-the art in this area. A failure detector [37] is an oracle that can be queried by processes in any step, before making a state transition. Failure detectors can be partially ordered w.r.t. their "solution power", i.e., their ability to make problems solvable in purely asynchronous systems. In [121], a failure detector called *anti*-Ω was shown to be the weakest for $n-1$-set agreement in shared memory systems [120]. A variant of anti-Ω, called

anti-Ω_k, returns $n-k$ processes and has been proved in [56] to be the weakest failure detector for k-set agreement in shared memory systems.

In message-passing systems, the "loneliness" failure detector $\mathcal{L}$ was shown to be the weakest failure detector for $(n-1)$-set agreement in [43]. With respect to general k-set agreement, [26, 27] introduced the quorum family Σ_k and proved that it is necessary for solving this problem. The paper also proved that the failure detector family $\Pi_k = \langle \Sigma_k, \Omega_k \rangle$ coincides with the weakest failure detectors $\langle \Sigma, \Omega \rangle$ for $k = 1$, and with $\mathcal{L}$ for $k = n-1$. Herein, Ω_k is a generalization of Ω introduced in [89], which returns sets of k process IDs that eventually stabilize and contain a correct process. However, for general values of $2 \leq k \leq n-2$, it was shown in [19, 30] that Π_k is too weak for k-set agreement. In [18, 23], we introduced the sufficiently strong $n-k$-loneliness failure detector $\mathcal{L}(k)$. In [87], it was shown that $\mathcal{L}(k)$ is equivalent to (Σ_k, X_k), where X_k adds some loneliness property to the output of Σ_k. Unfortunately, whereas $(\Sigma_k \Pi_k)$ is too weak, (Σ_k, X_k) turned out to be too strong for being the weakest failure detector for message-passing k-set agreement.

1.3.4 Connectivity in wireless networks

Although the focus of ADynNet is on algorithmic design and correctness proofs, we will also consider the aspect of assumption coverage of our models. We hence provide a glimpse of the wealth of literature on analyzing the connectivity in wireless networks, in particular, *wireless sensor networks* (WSN), and mobile ad-hoc networks below. Not surprisingly, both theoretical analyses based on random graphs, simulations, and measurements of real systems are abundant, see e.g. [79, 91] for an overview.

Random graphs are the result of constructing a graph according to some stochastic process, like the random placement of nodes in a metric space in conjunction with the assumption that only nodes within some maximum transmission range r can communicate with each other. It is well-known that such graphs exhibit fast phase-transitions: For example, there is some critical radius r_0 such that the random graph essentially consists of isolated small components for $r < r_0$ but contains a huge connected *giant component* [70] for $r > r_0$ [58, 93]. Celebrated papers like [61, 101] established conditions on r that guarantee connectivity almost surely; [92] and the work of Bettstetter [15] give conditions for k-connectivity. Particularly interesting in our context are [46, 90], which study directed random graphs.

Modeling real wireless networks by means of the above "regular" random graphs has the advantage of being analytically tractable, but has questionable assumption coverage in real systems. Researchers have hence studied variants of random graph models. Examples are non-uniform transmission ranges [116], where bounds on the critical node degree that ensures connectivity almost surely were established. In order to also incorporate interference, models based on a minimal *signal-to-interference-plus-noise* (SNIR) ratio have been proposed [47]. Shadow effects due to obstacles in the signal propagation paths, typically log-normal, have also been incorporated in the work by Bettstetter and Hartmann [16]. A particularly promising extension of this model, which also incorporates multipath fading, is the work by Schilcher, Bettstetter and Brandner [104].

A popular alternative to analytic models are network simulation models, which are executed on discrete-event network simulators such as ns-2 [51], Emstar [59] or SWAN [91]. Essentially, those simulators allow to place network nodes (along with their software, including TCP/IP stack and routing, for example) and to simulate the behavior of the resulting system. A wide variety of different radio models have been developed for this purpose, ranging from simple free-space propagation to general shadowing

models; more advanced models [35, 91] have been tailored to match the behavior of real systems.

Finally, there is a substantial body of work describing measurements in various prototype systems. Particularly interesting for ADynNet is the wealth of open testbeds [111] like IoT-Lab [44], WISEBED [41] or SmartSantander [88], which provide a powerful infrastructure for dedicated experiments. Unfortunately, however, the (statistical) data provided in existing experimental evaluations typically address the properties of individual links or system-wide properties like throughput only [35, 36, 91]. By contrast, in ADynNet, we are primarily interested in structural properties of communication graphs.

There is a large body of work devoted to experimentally exploring network topologies, which use active probing or passive monitoring and may or may not require support from intermediate nodes. However, the inferred topology information is usually quite restricted, typically to network cardinality [1] or capacity [14]. Moreover, the topology of the underlying network is often limited. For example, the approach described in [103] uses the data correlation caused by intermediate network coding for inferring tree or DAG topologies. By contrast, [94] uses active probing with traceroute data, and primarily addresses problems caused by anonymous/non-cooperative intermediate nodes and the resulting uncertainties in topology inference. Pure network tomography approaches infer the network topology solely from data available at end nodes, typically using statistical approaches [34].

There is also a substantial body of work on connectivity monitoring in wireless sensor networks. Both active probing [42, 118], where (a subset of) the network nodes query their neighborhood and forward connectivity data to some sink node, and passive techniques using data available at end-nodes only, as in network tomography approaches, can be employed here. Typical approaches using the latter, like [72], assume that the WSN topology is a convergecast tree, where all nodes periodically send their data to a sink, using data aggregation. The topology is then reconstructed from the data received at the sink.

All these solutions provide, with varying accuracy, (part of) the entire topology. However, we are not aware of approaches that directly infer sub-graph properties such as, for example, the presence of a rooted spanning tree or a strongly connected component in the communication graph.

1.3.5 Existing own results

Our group has actively contributed to several areas of research relevant for this proposal in the past. Besides the perception-based hybrid failure model [24,108] and a Byzantine extension of the HO-Model [17] already described in Section 1.3.2, we contributed significantly to interval-based clock synchronization [83, 106], including advanced algorithms for approximate agreement [107]. Our rich experience in the mathematical analysis of algorithms [105] and protocols [48] will prove useful in the planned analytic treatment of the assumption coverage [108]. More recently, we provided substantial contributions to agreement with weak timely links in classic distributed systems [23, 68], and to the chase for the weakest failures detector for message-passing k-set agreement [18, 23], which also includes a novel tool [19] for "easy impossibility proofs".

The above research also stimulated our initial work on agreement in dynamic systems [21], where we provided the first consensus algorithm for directed graphs. It requires that, in every round, the communication graph is both (i) weakly connected and (ii) contains a single *root component*, i.e., a strongly connected component (SCC) without incoming links. The latter must eventually become *vertex stable*, in the sense that its vertex set remains the same (with possibly changing interconnection topology) for a certain number of rounds; such a SCC is called a *vertex-stable root component* (VSRC). Since these

assumptions do not guarantee bidirectional reachability system-wide, the model in [21] falls between the weakest and second weakest class of models defined in [33].

In [20], we studied k-set agreement under quite strong network assumptions, which stipulate the existence of a *static* skeleton sub-graph (that exists in all rounds). Clearly, the latter is even less likely to be found in a dynamic network than the network assumptions of [21].

The above results—that is to say, their deficiencies and the resulting research questions—fueled our interest in dynamic systems in general (and also triggered the setup and writing of the project proposal ADynNet). And indeed, since the submission of the first version of our proposal, we made important steps forward in our research agenda: Most importantly, we very recently published a paper [109, 110] at the premium distributed computing conference PODC'14, which contains the very first k-uniform k-set agreement algorithm that works under very weak networks assumptions. Note that this also stimulated the new external collaborations on ADynNet-related topics mentioned in this revised version of our proposal.

1.4 General Methodological Approach

Generally speaking, the focus of ADynNet is on algorithmic design and correctness proofs, and on general solvability conditions that render gracefully degrading consensus in dynamic networks feasible. The work will be primarily conceptual (model and algorithm design) and theoretical (correctness proofs, performance analysis and coverage analysis), but will also involve the setup of a follow-up project devoted to the experimental evaluation of the models and algorithms developed in ADynNet. External collaborations both with systems engineers and leading experts in related fields will complement our local expertise in several parts of the project.

The particular work in ADynNet is partitioned into three reasonably independent tasks, which have a clear methodological focus each and hence require very different talents and skills. Every task will be assigned to a dedicated PhD student.[4]

Task 1 **Development of network assumptions and solvability conditions**

Task 2 **Development of algorithms, correctness proofs and performance analyses**

Task 3 **Coverage analysis**

The tasks are independent of each other, except for the inherent collaboration required at the level of the common "interface", i.e., the candidate network assumptions. Note carefully that the setup of our tasks does not even require to synchronize their internal progress, since our already existing recent results allow every task to start immediately at the beginning of the project. Consequently, every task can just progress at its own pace.

1.5 Detailed Project Goals and Workplan

1.5.1 Task 1: Development of network assumptions and solvability conditions

Start	Duration	Responsible persons
$T0$	3 years	Kyrill Winkler (PhD)

[4]Note that both the amount of work in every task and the very diverse required skills make it impossible to assign more that one task to a single PhD student.

Given that it is of course impossible to solve any meaningful problem in a distributed computing system without at least some restriction of the *adversary*[5] (as just prohibiting any communication trivially leads to impossibility results for any task that requires at least some information exchange among the processes), a pivotal part of ADynNet is to come up with suitable network assumptions for solving a certain weak agreement problem: On the one hand, they must be strong enough to render the problem solvable, on the other hand, they must be weak enough to have a sufficiently high assumption coverage in real dynamic networks.

More generally, as already argued, knowing or at least approaching the *exact* solvability/impossibility border is interesting for several reasons: First, it is interesting from a theoretical point of view. In particular, k-set agreement has been a major target for the study of solvability in classic asynchronous distributed systems, augmented with failure detectors, since decades. Second, striving for weak network assumptions is always advantageous w.r.t. the assumption coverage in real systems, as they are typically more likely to hold in a given dynamic network. Finally, a set of network assumptions close to the necessary and sufficient ones is needed for developing agreement algorithms that indeed degrade *gracefully*: For a given communication scenario, such an algorithm should solve the strongest form of agreement possible.

Obviously, the problem is further acerbated if process failures and relaxed synchrony are added to the picture: Unfortunately, in real systems, one cannot rule out the possibility of processes that send out erroneous information, due to many reasons, ranging from synchronization errors to Byzantine behavior [17, 24]. Whereas benign process failures are easily incorporated in our model, this is not the case for non-benign failures. Defining appropriate network assumptions that also allow for (some) non-benign failures and (some) relaxed synchrony remains a major challenge. One particularly important class of failures, which shall be addressed, are processes with out-of-sync clocks.

Fortunately, there is a solid basis of existing theoretical work, which allows us to perform Task 1 quite independently of the algorithmic work conducted in Task 2. Thanks to our recent efforts, we have already a fairly clear picture of the first steps to be conducted in ADynNet. We will provide a glimpse of our ideas below, part of which will also involve external collaborations.

Network assumptions for approximate agreement. We will use the fact that the matrix-based approach used by Hendricks, Olshevsky and Tsitsiklis [65, 66] can be generalized to our model of time-varying communication graphs. Since it is easy to express classic averaging approximate agreement algorithms in this setting, network assumptions translate into properties of stochastic matrices, which can be analyzed by powerful tools from (linear) algebra. This part of our work will be conducted in a well-established collaboration with Bernadette Charron-Bost (Ecole Polytechnique Paris).

Network assumptions for k-set agreement. We will use the model proposed in [21, 109, 110] as a starting point: It rests on the number of *vertex-stable root components* (VSRC) that exist in a run, which are strongly connected components without incoming edges from processes not in the VSRC. Every weakly connected graph has at least one VSRC, and [21] proves that solving consensus requires exactly one VSRC that is stable for some minimal duration. Interestingly, for general k-set agreement, recent findings [109, 110] disproved the initial conjecture that at most k VSRCs might be a sufficiently strong network assumption. In fact, it turned out that not even at most $k/2$ VSRCs suffice for correctly solving k-set agreement. Consequently, additional properties ("majority influence") had to be added to

[5] Distributed algorithms research considers anything not in the sphere of control of an algorithm (like times when a process executes its computing steps, times when messages are delivered to the receiver, and also occurrences of process failures and message loss) as something controlled by an adversary. Hence, network assumptions actually restrict the power of the adversary.

the network assumptions to make k-set agreement solvable by the algorithm given in [109].

However, we are convinced that the additional "majority influence" property can be weakened considerably. More generally, VSRCs are certainly not the only (not to speak of the ultimate) possibility for defining such network assumptions. Our idea is to consider other variants of T-interval connectivity [74] and combine it with ideas in the fundamental work on Δ-coordinated consensus in bidirectional graphs by Kuhn, Oshman and Moses [76]. More specifically, we envision the definition of abstract "meta-communication-graphs", which are derived from the actual communication graphs by means of a suitable composition relation. We hope to be able to find relatively simple properties of these meta-communication-graphs, which are necessary and sufficient for solving k-set agreement.

New impossibility proof techniques. We are convinced that our generic "easy impossibility proof" technique [19], which provides k-set agreement impossibilities using a reduction to consensus impossibilities (like the ones established in [22]), will allow us to obtain new impossibility results for k-set agreement in dynamic networks. Furthermore, we conjecture that it is worthwhile to explore the topological equivalent of [19], as it may provide additional insights into the algebraic structure of the protocol complex of k-set agreement. Finally, if appropriate, we will of course also utilize proof techniques based on algebraic topology resp. Sperner's lemma [29, 67, 100].

Furthermore, we expect that directed dynamic networks are particularly amenable to knowledge-based impossibility proofs. Impossibility proofs typically exploit some uncertainty of the local processes about the global system state, which, in case of dynamic networks, primarily results from the time-variability of the effective (multi-hop) communication delays caused by the dynamicity of the network topology. Unifying timing uncertainty and topology changes by causality, i.e., knowledge, in the spirit of [12, 76], is hence a particularly appealing idea that shall be explored in the collaboration with Yoram Moses (Technion Haifa) and Calvin Newport (Georgetown University).

Overall, we hope that this will lead to even more powerful impossibility proof techniques specifically designed for dynamic networks.

Work plan. The work to be done in Task 1 will be performed by a PhD student (Kyrill Winkler), who has already proved his interest in and talent for theoretical work in [110]. The efforts will be structured in the following two concurrent sub-tasks:

(1a) Devise new impossibility proofs and candidate network assumptions, guided by existing impossibility results. Our approach here is to strengthen the network assumptions that render k-set agreement impossible, to find one that either (i) circumvents the respective impossibility proof or (ii) allows to extend the impossibility proof. Case (i) results in a candidate network assumption, case (ii) in a stronger impossibility result. Since the space for choosing additional properties is huge, the challenge here is to find some appropriate strengthening:[6] With respect to exploring the possibility/impossibility border, outcome (i) is particularly interesting if the strengthening has been as little as possible, whereas (ii) is particularly effective if the strengthening has been as much as possible.

(1b) Hand over candidate network assumptions (resulting from (1a).ii) to Task 2 (see Section 1.5.2) of ADynNet, for defining candidate algorithms and corresponding correctness proofs and performance

[6] To be more specific here, the results of [109] revealed that even considerably less than k VSRCs in every round are *not* sufficient for solving k-set agreement, if these VSRCs cannot "observe" each other (in the sense of not knowing of their existence). Hence, we added a "majority influence" property to our network assumptions, which guarantees that a majority of processes in a VSRC must have heard something from (some) other VSRCs. This provided a network assumption, which eventually turned out to be sufficiently strong, as it allowed us to develop a correct k-set agreement algorithm.

analyses, and to Task 3 for the analysis of the assumption coverage.

(2) Wait for feedback from Task 2, and possibly also from Task 3, to refine/modify candidate network assumptions according to (1) until they are satisfactory, i.e., allow the implementation of a correct solution algorithm and have reasonable assumption coverage.

Note carefully that Task 1 can indeed proceed at its own pace, since the iteration loop in (1a)+(1b) does not (necessarily) require feedback from (2) to proceed.

To support the external collaboration with Bernadette Charron-Bost and Yoram Moses, we foresee additional financial support for two 1-month research visits at Ecole Polytechnique and Technion Haifa (EUR 2.500,– each).

1.5.2 Task 2: Solution algorithms and correctness proofs

Start	Duration	Responsible persons
$T0$	3 years	Manfred Schwarz (PhD)

Task 2 is devoted to the detailed design of solution algorithms, mathematical correctness proofs and performance analyses.

The major challenges of this task are algorithmic problems, which are orthogonal to the issue of sufficient network assumptions considered in Task 1: First, given some network assumption, one needs to find specific graph properties guaranteed by the adversary, which can be exploited algorithmically. However, given the dynamicity inherent in a perpetually changing communication graph, locally detecting at a process that such a property indeed holds is a major challenge. Second, weak network properties like the ones assumed in [21, 109] currently entail algorithms with high communication and memory complexity. There is certainly a tradeoff between the weakness of network assumptions and algorithmic complexity (memory consumption, communication complexity) and hence performance. The question of finding *efficient* algorithms, ideally matching complexity lower bounds, is hence not only of practical relevance, but also interesting for exploring this tradeoff. Third, coping with non-benign faulty processes and synchronization errors makes it inevitable to incorporate fault-tolerance techniques [17] in the algorithms. This algorithmic work to be conducted in Task 2 will continue our long-standing and successful collaboration with Martin Biely (EPFL) and will also involve a collaboration with Christian Scheideler (University of Paderborn).

Although primarily focusing on deterministic algorithms, we will also consider randomized algorithms here, albeit our strive for always safe ones ("Las Vegas"-type) limits the power of randomization: In [113], it has been shown that randomization is not really effective for the coordinated attack problem in the presence of strong adversaries, even for "Monte Carlo"-type solutions (that allow a non-zero probability of violating agreement). In fact, existing MC solutions (like [6], which solves consensus almost everywhere in dynamic networks, albeit with churn and even Byzantine faulty processes) typically assume relatively strong network properties (typically expander graphs). Even stronger properties had to be used for "Las Vegas"-type solutions; some examples are expander graphs + polynomially-sized messages [7] or a restricted number of link omission faults [86]. We expect that randomized algorithms that can guarantee agreement among *all* communication neighbors in fact require similar network properties as deterministic ones. This part of the work in ADynNet will benefit from our long-standing collaboration with Peter Robinson (National University of Singapore), who is particularly interested in randomized algorithms for dynamic networks.

A different aspect that shall also be addressed to some extent in Task 2, is the utility of weak distributed agreement in dynamic network applications. In particular, in an external collaboration with the systems engineering expert Kay Römer (TU Graz), we will look out for meaningful low-level services that could be built atop of gracefully degrading k-set agreement and other forms of gracefully degrading consensus. Note that this collaboration will also address related aspects like how to actually accomplish transmission scheduling negotiation, partition merging, defining alternative validity conditions for k-set agreement that better capture partitioning, etc.

Thanks to our initial work, we have already some very specific starting points for the particular work to be conducted in Task 2:

New paradigms for algorithm design. The algorithms developed in our initial work [21, 109] could be described as "sender-based" decision making: Some nodes decide after determining that it is "safe to decide" and enforce this decision on all nodes in the network using explicit *decided* messages. An interesting alternative is "receiver-based" decision making, where all members in the network decide after observing a system configuration they deem "safe to decide" individually. This entails that processes must somehow infer locally that the system has reached a configuration where all other processes will sooner or later determine that it is safe to decide (on the very same decision value). Although being algorithmically more complex, such algorithms may provide faster termination times.

Efficient gracefully degrading consensus algorithms. The algorithms provided in [21] and [109] suffer from high complexity, both in terms of communication and local memory requirements. Finding (much) more efficient algorithms is hence instrumental from a theoretical as well as a practical perspective.

From a knowledge point of view, the ability of at least one process to spread some local value to every process in the system is a necessary condition for achieving consensus. Interestingly, our preliminary results show that the existence of such an "unknown broadcaster" can always be guaranteed in a directed dynamic network, even under very weak conditions. Besides solving the interesting (but non-trivial) question for a tight worst-case bound on the broadcasting time, which gives a lower bound for the consensus termination time, solving the question of how to exploit the existence of an unknown broadcaster algorithmically may be the key to more efficient algorithms.

Applications of weak distributed agreement. The k-uniform k-set agreement algorithm developed in [109] has the interesting property that it also respects partitions: If the system splits into multiple isolated parts, it ensures that processes in the same single partition deliver the same decision value, i.e., reach consensus. We argue that an interesting application of this type of algorithms could be agreement on communication schedules [96], which are typically used for mutual interference reduction [97] and energy saving [81]. Obviously, only processes that can and will communicate (frequently) with each other—and are hence in the same partition—need to agree on a communication schedule. Note that such a solution goes far beyond approaches like the one proposed by Boano, Zuniga, Römer and Voight [25], which implement agreement between neighbors (i.e., 2-process consensus) only.

Interestingly, this application would also allow us to go somewhat beyond the usual perspective of dynamic networks as a game between an algorithm and an *oblivious* adversary, which is not influenced by the execution of the algorithm: Installing the agreed-upon schedule after completion of k-set agreement should *improve* the communication between the processes in the partition, i.e., change the communication graphs observed later on. Clearly, this induces some *control* of the algorithm over the network topology,[7]

[7]Note that this is evident in dynamic networks with high mobility, where the movement of the processes (controlled by the

i.e., the adversary, which raises a number of interesting theoretical and practical questions for future research.

Work plan. The work to be done in Task 2 will be performed by a PhD student (Manfred Schwarz), who has already proved his interest in and talent for algorithm design and correctness proofs in [110]. Similarly to Task 1, the efforts will be structured in the following three concurrent sub-tasks:

(1a) Starting out from network assumptions provided by Task 1 (see Section 1.5.1), design candidate algorithms for the specific problem at hand (primarily variants of k-set agreement). Initially, (1a) starts with improving the existing algorithms, as outlined above.

(1b) Work out the detailed correctness proofs, which (hopefully) show that a given algorithm indeed solves the problem under the given network assumptions.

(1c) Give feedback to Task 1 on the network assumptions provided: Suggest some possible strengthening in case they are too weak for an algorithmic solution.

(2) Provide an analysis of the communication and memory complexity of the algorithm, and devise possible improvements, if necessary.

(3) Assess/improve the utility of the weak agreement problems at hand, based on external systems engineering expertise.

Needless to say, (1a)+(1b)+(1c) and (2) have to be conducted for every candidate network assumption developed in the course of ADynNet. To support the external collaboration with Christian Scheideler (University of Paderborn), we foresee additional financial support for one 1-month research visit at University Paderborn (EUR 2.500,–).

1.5.3 Task 3: Coverage analysis

Start	Duration	Responsible persons
$T0$	3 years	NN (PhD)

Task 3 is devoted to the analysis of the assumption coverage in real-world systems. Whereas we will primarily use theoretical analysis and simulations based on random graph models, our external collaboration with Kay Römer (TU-Graz) will also be used to setup a follow-up project: It shall complement the theoretical results obtained here by a complementary experimental validation and performance evaluation, using appropriate testbeds in the future.

Recall that, given some specific network assumption and some real-world dynamic network (or a realistic model thereof), the purpose of a coverage analysis is to compute some measure (typically, a probability) that the network assumption holds in the real-world dynamic network. Essentially, it bridges the gap between the adversarial world of our network assumptions and the non-adversarial (typically, probabilistic) world of real systems. Whereas the need for a coverage analysis has been widely recognized in fault-tolerant distributed computing research [95], to the best of our knowledge, it has only rarely [108] been used for assessing network assumptions in dynamic network-related research.

algorithm) obviously affects communication. It is not so evident in the restricted setting of ADynNet, where we deliberately excluded substantial churn.

Whereas a weaker network assumption cannot have a smaller assumption coverage than a stronger one, it is in general not possible to infer the assumption coverage from network assumptions in general: For example, two very different network assumptions such as (i) a single strongly connected component in every round (sufficient for consensus) and (ii) a rooted tree in every round (not sufficient for consensus, but conjectured to be sufficient for approximate agreement), which clearly satisfy the relation that (i) implies (ii) but not vice versa, may have almost the same assumption coverage in a given dynamic network: In a random graph model above the critical threshold (where the giant component has formed), (i) and (ii) have essentially the same probability. So whereas weaker semantics are always advantageous in terms of the assumption coverage, the actual benefit may be less than expected.

Consequently, determining whether a certain algorithm A, based on some weak network assumption and thus entailing high algorithmic complexity, is indeed preferable over some more efficient algorithm B that relies on a stronger network assumption, is impossible without a dedicated coverage analysis.

In Task 3, we will primarily focus on *analytic* coverage analysis, combined with supporting simulations. Restricting our attention to analytic coverage analysis has the advantage of being "exhaustive", in the sense that the results are representative and not just a more or less random sample of possible behaviors (as typically provided by simulations and experiments). Moreover, analyzing the assumption coverage in a "generic" model, rather than in a very specific dynamic network, potentially leads to wider applicability of the results.

We expect the actual work to be conducted in Task 3 to evolve from the following starting points:

Analytic coverage analysis using analytic combinatorics. Starting out from suitable random graph models, in particular, the one by Schilcher, Bettstetter and Brandner [104] that incorporates interference and fading, and [46, 90] that supports unidirectional edges and arbitrary degree distributions, we will employ methods from analytic combinatorics [53] for computing the assumption coverage of a certain network assumption, say, the existence of a rooted tree or some other topological sub-structure, in a (sequence of) random graphs. Whereas we expect to be able to re-use some existing results in certain cases (e.g., regarding k-connected components [15, 92] provided by Penrose and Bettstetter), it is quite likely that most of our network assumptions will force us to develop new results. For this purpose, we will try to employ transfer methods for deriving functional equations of the associated probability generating functions, which can usually be attacked by suitable asymptotic methods to derive the required probabilities. Note that this analytic approach has been successfully applied to various problems in our past research, see e.g. [48, 105].

This part of our work in Task 3 will be supported by an external collaboration with Christian Bettstetter (Lakeside Labs Klagenfurt), who will provide us with both expertise regarding their random graph model [104] and appropriate simulation environments.

Supporting simulations. As is frequently the case with analytic approaches nowadays, simulations (e.g. using MATLAB) are a very flexible and powerful means for getting clues about analytic solutions. Note carefully, though, that these simulations are deliberately performed at the model level, i.e., are not meant to be realistic in the sense of e.g. also incorporating the software and protocol stack of real network nodes (as is possible with simulators such as ns-2 [51], Emstar [59] or SWAN [91]). After all, these simulations shall be representative for the particular random graph model, not for a specific dynamic network implementation.

Complementary experiments. Since a "generic" coverage analysis as outlined above cannot replace

real experimental evaluation/simulation entirely, as it necessarily abstracts away many aspects of real implementations, we envision to complement the analytic work to be done in Task 3 by some experiments.

Given that sound experiments require substantial engineering efforts, however, it does not make sense to incorporate a fully-fledged experimental evaluation of the models and algorithms to be developed in ADynNet in this proposal. After all, ADynNet is a foundational project that runs only 3 years. What makes sense, and will hence be part of ADynNet, is to additionally conduct some limited topology monitoring in real testbeds such as IoT-Lab [44], WISEBED [41] or SmartSantander [88], and to analyze the connectivity data to validate a given network assumption. As for the analytic part of Task 3, this boils down to detecting/inferring certain topological sub-structures in the communication graphs from measurement results, cf. [94].

We have recently developed a prototype measurement infrastructure, which allows us to do some topology monitoring using our existing educational wireless sensor network nodes (Atmel motes running TinyOS). It comprises the implementation of a synchronous rounds abstraction based on synchronized clocks, a suitable multi-hop data gathering scheme, and a flexible post-processing tool for the actual analysis and validation of candidate network assumptions.

This preparatory work was very useful for setting up the external collaboration with Kay Römer (TU Graz), which shall eventually lead to a future experimental evaluation of appropriately extended and engineered variants of the models and solution algorithms developed in ADynNet, using more suitable testbeds.

Work plan. The work to be done in Task 3 will be performed by a PhD student (NN1) with a strong background in mathematics (graph theory, combinatorics, probability theory), who still needs to be hired. The efforts will be structured in the following sequential steps:

(0) Select/adapt/develop a random graph model suitable for modeling directed dynamic networks, as well as a set of coverage analysis methods for "typical" graph structures.

(1) For every network assumption provided by Task 1 (see Section 1.5.1):

- Develop suitable analysis techniques for computing the respective assumption coverage, possibly using supporting simulations.
- Initiate supporting topology monitoring experiments.
- Provide feedback on the actual assumption coverage to Task 1.

To support the external collaboration with Christian Bettstetter and Kay Römer, we foresee additional financial support for two 1-month research visits at Lakeside Labs Klagenfurt and TU Graz (EUR 1.250,– each).

2 Organizational Aspects

2.1 Institution: Institut für Technische Informatik, TU-Wien

The research and teaching activities of the *Institut für Technische Informatik* (Institute of Computer Engineering, E182) at TU Vienna are devoted to cyber-physical systems, with particular emphasis on dependable embedded systems and hybrid systems.

Research is primarily devoted to fundamental and technological problems that require a scientific approach and a long time horizon. Major research topics are hybrid systems, fault-tolerant distributed algorithms, time-triggered real-time systems, and dependable Systems-on-Chip for critical embedded systems. The spectrum of methods applied to these problems ranges from formal mathematical analysis to experimental evaluation of prototype implementations. With respect to teaching, the Institut für Technische Informatik runs a Master and Bachelor program "Technische Informatik", which offer a thorough scientific and engineering education in the field of dependable embedded systems and hybrid systems.

The work in ADynNet will be conducted by the institute's *Embedded Computing Systems* (ECS) group E182/2, which provides (and integrates) expertise both in fault-tolerant distributed algorithms and dependable integrated circuits. The ECS group actually consists of one full professor (Ulrich Schmid, who works on fault-tolerant distributed algorithms), one associate professor (Andreas Steininger, who works on dependable digital circuits), four assistant professors, one secretary, and one technician. Rooms and infrastructure for the required additional project staff, as well as the required infrastructure and support for embedded systems design (required for the planned sensor network experiments) are available.

ADynNet will contribute to secure (and hopefully increase) the existing reputation of our group in fault-tolerant distributed algorithms research.

2.2 Contributing Human Resources

- The project leader **Ulrich Schmid** is full professor and head of the Embedded Computing Systems Group since 2003. He authored and co-authored numerous papers in the field of theoretical and technical computer science and received several awards and prizes, like the Austrian START-prize 1996. Ulrich Schmid also spent several years in industrial electronics and embedded systems design. His current research interests focus on the mathematical analysis of fault-tolerant distributed algorithms and real-time systems, with special emphasis on their application in systems-on-chips and networked embedded systems. As ADynNet lies in his major areas of interest, Ulrich Schmid will actively contribute to the modeling and correctness analysis work to be conducted in ADynNet.

- **Matthias Függer** is assistant professor (PostDoc) in the Embedded Computing Systems Group. His research interests lie in the area of formal analysis of highly parallel distributed systems of computationally weak devices (like VLSI systems and sensor networks), fault-tolerant clock synchronization algorithms, and formal verification. ADynNet will allow Matthias Függer, who is working towards his habilitation, to further develop his scientific career into this important direction of research.

- **Kyrill Winkler** is a PhD student interested in the modeling and analysis of distributed algorithms, in particular, in impossibility and lower bound results and formal-mathematical analysis in general. Thanks to his Master thesis *Easy impossibility proofs for k-set agreement*, he has provided key impossibility results to [109, 110] and is hence exceptionally qualified for Task 1 of ADynNet. The cutting-edge research topics addressed in this proposal will allow him to both deepen and broaden his knowledge and his analytic skills, and to gain international recognition.

- **Manfred Schwarz** is a PhD student interested the design and analysis of distributed algorithms. In his Master thesis *Solving k-Set Agreement in Dynamic Networks*, he developed the k-uniform k-set agreement algorithm published in [109, 110]. Manfred Schwarz is hence ideally qualified for the work in Task 2 of ADynNet. The cutting-edge research topics addressed in this proposal will allow

him to both deepen and broaden his knowledge and his analytic skills, and to gain international recognition.

- The PhD position **NN1** is still open.
- ADynNet will also involve a number of external collaborations, in particular, **Christian Bettstetter** (Lakeside Labs Klagenfurt), **Martin Biely** (EPFL), **Bernadette Charron-Bost** (Ecole Polytechnique Paris), **Yoram Moses** (Technion Haifa), **Calvin Newport** (Georgetown University) **Peter Robinson** (Singapore National University), **Christian Scheideler** (University of Paderborn), **Kay Römer** (TU Graz).

2.3 Required Other Costs

To support the external collaborations in ADynNet, additional financial support for 5 x 1-month research visits (estimated total costs EUR 10.000,–) are foreseen. All the required infrastructure (including sensor network nodes) will be available, so there are no further additional costs.

3 Expected Additional Benefits

Since we hope to improve the scientific state-of-the-art in the area of dynamic networks both with respect to theory and practice (in particular, by providing new algorithmic solutions), we expect ADynNet not only to improve the state of the art in networking, distributed computing and theoretical computer science in general, but also to generate mid-term benefits for the industrial practice. Moreover, given the importance of dynamic networks in other disciplines, including systems biology and social sciences, even cross-disciplinary benefits may eventually emerge from our research.

4 Abbreviations

Abbrev.	**Description**	**Definition**
ECS	Embedded computing systems	Sect. 2.1
MANET	Mobile ad-hoc network	Sect. 1.1
SCC	Strongly connected component	Sect. 1.3.5
SNIR	Signal-to-interference-plus-noise ratio	Sect. 1.3.4
VSRC	Vertex-stable root component	Sect. 1.3.5
WSN	Wireless sensor network	Sect. 1.3.4

References

[1] H. B. Acharya and M. G. Gouda. On the hardness of topology inference. In *ICDCN*, pages 251–262, 2011.

[2] Y. Afek and E. Gafni. Asynchrony from synchrony. In D. Frey, M. Raynal, S. Sarkar, R. Shyamasundar, and P. Sinha, editors, *Distributed Computing and Networking*, volume 7730 of *Lecture Notes in Computer Science*, pages 225–239. Springer Berlin Heidelberg, 2013.

[3] Y. Afek, E. Gafni, and A. Rosen. The slide mechanism with applications in dynamic networks. In *ACM PODC*, pages 35–46, 1992.

[4] M. K. Aguilera, W. Chen, and S. Toueg. Using the heartbeat failure detector for quiescent reliable communication and consensus in partitionable networks. *Theoretical Computer Science*, 220(1):3–30, June 1999.

[5] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393 – 422, 2002.

[6] J. Augustine, G. Pandurangan, and P. Robinson. Fast byzantine agreement in dynamic networks. In *Proceedings PODC'13*, pages 74–83, 2013.

[7] J. Augustine, G. Pandurangan, P. Robinson, and E. Upfal. Towards robust and efficient computation in dynamic peer-to-peer networks. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '12, pages 551–569. SIAM, 2012.

[8] B. Awerbuch. Complexity of network synchronization. *Journal of the ACM (JACM)*, 32(4):804–823, 1985.

[9] B. Awerbuch, B. Patt-Shamir, D. Peleg, and M. E. Saks. Adapting to asynchronous dynamic networks. In *STOC'92*, pages 557–570, 1992.

[10] M. H. Azadmanesh and R. M. Kieckhafer. Exploiting omissive faults in synchronous approximate agreement. *IEEE Transactions on Computers*, 49(10):1031–1042, Oct. 2000.

[11] M. H. Azadmanesh and R. M. Kieckhafer. Asynchronous approximate agreement in partially connected networks. *International Journal of Parallel and Distributed Systems and Networks*, 5(1):26–34, 2002.

[12] I. Ben-Zvi and Y. Moses. Beyond Lamport's happened-before: On the role of time bounds in synchronous systems. In N. Lynch and A. Shvartsman, editors, *Distributed Computing*, volume 6343 of *Lecture Notes in Computer Science*, pages 421–436. Springer Berlin / Heidelberg, 2010.

[13] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, MA, 1989.

[14] A. Bestavros, J. W. Byers, and K. A. Harfoush. Inference and labeling of metric-induced network topologies. *IEEE Transactions on Parallel and Distributed Systems*, 16(11):1053–1065, 2005.

[15] C. Bettstetter. On the minimum node degree and connectivity of a wireless multihop network. In *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pages 80–91, 2002.

[16] C. Bettstetter and C. Hartmann. Connectivity of wireless multihop networks in a shadow fading environment. *Wirel. Netw.*, 11(5):571–579, Sept. 2005.

[17] M. Biely, B. Charron-Bost, A. Gaillard, M. Hutle, A. Schiper, and J. Widder. Tolerating corrupted communication. In *Proceedings of the 26th ACM Symposium on Principles of Distributed Computing (PODC'07)*, pages 244–253, Portland, OR, USA, Aug. 2007. ACM.

[18] M. Biely, P. Robinson, and U. Schmid. Weak synchrony models and failure detectors for message passing k-set agreement. In *Proceedings of the International Conference on Principles of Distributed Systems (OPODIS'09)*, LNCS, pages 285–299, Nimes, France, Dec 2009. Springer Verlag.

[19] M. Biely, P. Robinson, and U. Schmid. Easy impossibility proofs for k-set agreement in message passing systems. In *Proceedings 15th International Conference on Principles of Distributed Systems (OPODIS'11), Springer LNCS 7109*, pages 299–312, 2011.

[20] M. Biely, P. Robinson, and U. Schmid. Solving k-set agreement with stable skeleton graphs. In T. Kikuno and T. Tsuchiya, editors, *IPDPS Workshops*, pages 1488–1495. IEEE, 2011.

[21] M. Biely, P. Robinson, and U. Schmid. Agreement in directed dynamic networks. In *Proceedings 19th International Colloquium on Structural Information and Communication Complexity (SIROCCO'12)*, LNCS 7355, pages 73–84. Springer-Verlag, 2012.

[22] M. Biely, P. Robinson, and U. Schmid. Agreement in directed dynamic networks. *arXiv:1204.0641*, 2012.

[23] M. Biely, P. Robinson, and U. Schmid. The generalized loneliness detector and weak system models for k-set agreement. *IEEE Transactions on Parallel and Distributed Systems*, 25(4):1078–1088, Apr. 2014.

[24] M. Biely, U. Schmid, and B. Weiss. Synchronous consensus under hybrid process and link failures. *Theoretical Computer Science*, 412(40):5602 – 5630, 2011. http://dx.doi.org/10.1016/j.tcs.2010.09.032.

[25] C. A. Boano, M. A. Zuniga, K. Romer, and T. Voigt. Jag: Reliable and predictable wireless agreement under external radio interference. *2013 IEEE 34th Real-Time Systems Symposium*, pages 315–326, 2012.

[26] F. Bonnet and M. Raynal. Looking for the weakest failure detector for k-set agreement in message-passing systems: Is Π_k the end of the road? In R. Guerraoui and F. Petit, editors, *11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2009)*, volume 5873 of *Lecture Notes in Computer Science*, pages 129–164, 2009.

[27] F. Bonnet and M. Raynal. On the road to the weakest failure detector for k-set agreement in message-passing systems. *Theoretical Computer Science*, In Press, Corrected Proof:–, 2010.
[28] F. Bonnet and M. Raynal. On the road to the weakest failure detector for k-set agreement in message-passing systems. *Theoretical Computer Science*, 412(33):4273 – 4284, 2011.
[29] E. Borowsky and E. Gafni. Generalized FLP impossibility result for t-resilient asynchronous computations. In *STOC '93: Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, pages 91–100, New York, NY, USA, 1993. ACM.
[30] Z. Bouzid and C. Travers. $(\text{anti—}\Omega^x \times \Sigma_z)$-based k-set agreement algorithms. In *Proceedings of the 14th international conference on Principles of distributed systems*, OPODIS'10, pages 189–204, Berlin, Heidelberg, 2010. Springer-Verlag.
[31] R. R. Brooks and S. S. Iyengar. Robust distributed computing and sensing algorithms. *IEEE Computer*, pages 53–60, June 1996.
[32] A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro. Time-varying graphs and dynamic networks. In H. Frey, X. Li, and S. Rührup, editors, *ADHOC-NOW*, volume 6811 of *Lecture Notes in Computer Science*, pages 346–359. Springer, 2011.
[33] A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro. Time-varying graphs and dynamic networks. *IJPEDS*, 27(5):387–408, 2012.
[34] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu. Network tomography: Recent developments. *Statistical Science*, 19(3):499–517, 08 2004.
[35] A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak, and D. Estrin. Statistical model of lossy links in wireless sensor networks. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, IPSN '05, Piscataway, NJ, USA, 2005. IEEE Press.
[36] A. Cerpa, J. L. Wong, M. Potkonjak, and D. Estrin. Temporal properties of low power wireless links: Modeling and implications on multi-hop routing. In *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '05, pages 414–425, New York, NY, USA, 2005. ACM.
[37] T. D. Chandra and S. Toueg. Unreliable failure detectors for reliable distributed systems. *Journal of the ACM*, 43(2):225–267, March 1996.
[38] B. Charron-Bost and A. Schiper. The Heard-Of model: computing in distributed systems with benign faults. *Distributed Computing*, 22(1):49–71, Apr. 2009.
[39] S. Chaudhuri. More *choices* allow more *faults*: Set consensus problems in totally asynchronous systems. *Information and Control*, 105(1):132–158, July 1993.
[40] H. C. Chung, P. Robinson, and J. L. Welch. Optimal regional consecutive leader election in mobile ad-hoc networks. In *Proceedings of the 7th ACM SIGACT/SIGMOBILE International Workshop on Foundations of Mobile Computing*, FOMC '11, pages 52–61, New York, NY, USA, 2011. ACM.
[41] G. Coulson, B. Porter, I. Chatzigiannakis, C. Koninis, S. Fischer, D. Pfisterer, D. Bimschas, T. Braun, P. Hurni, M. Anwander, G. Wagenknecht, S. P. Fekete, A. Kröller, and T. Baumgartner. Flexible experimentation in wireless sensor networks. *Commun. ACM*, 55(1):82–90, Jan. 2012.
[42] B. Deb, S. Bhatnagar, and B. Nath. Stream: Sensor topology retrieval at multiple resolutions. *Telecommunication Systems*, 26(2-4):285–320, 2004.
[43] C. Delporte-Gallet, H. Fauconnier, R. Guerraoui, and A. Tielmann. The weakest failure detector for message passing set-agreement. In *DISC '08: Proceedings of the 22nd international symposium on Distributed Computing*, pages 109–120, Berlin, Heidelberg, 2008. Springer-Verlag.
[44] C. B. des Roziers, G. Chelius, T. Ducrocq, E. Fleury, A. Fraboulet, A. Gallais, N. Mitton, T. Noël, and J. Vandaele. Using senslab as a first class scientific tool for large scale wireless sensor network experiments. In *Networking (1)*, pages 147–159, 2011.
[45] D. Dolev, N. Lynch, S. S. Pinter, E. W. Stark, and W. E. Weihl. Reaching approximate agreement in the presence of faults. *Journal of the ACM*, 33(3):499–516, July 1986.
[46] S. N. Dorogovtsev, J. F. F. Mendes, and A. N. Samukhin. Giant strongly connected component of directed networks. *Phys. Rev. E*, 64:025101, Jul 2001.
[47] O. Dousse, F. Baccelli, and P. Thiran. Impact of interferences on connectivity in ad hoc networks. *IEEE/ACM Trans. Netw.*, 13(2):425–436, Apr. 2005.
[48] M. Drmota and U. Schmid. The analysis of the expected successful operation time of slotted ALOHA. *IEEE Transactions on Information Theory*, 39(5):1567–1577, 1993.
[49] C. Dwork, D. Peleg, N. Pippenger, and E. Upfal. Fault tolerance in networks of bounded degree. *SIAM J. Comput.*, 17(5):975–988, 1988.
[50] M. Faezipour, M. Nourani, A. Saeed, and S. Addepalli. Progress and challenges in intelligent vehicle area networks. *Commun. ACM*, 55(2):90–100, Feb. 2012.
[51] K. Fall and K. V. (eds.). The ns manual. http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf.

[52] A. D. Fekete. Asynchronous approximate agreement. *Information and Computation*, 115(1):95–124, November 15 1994.

[53] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009. http://algo.inria.fr/flajolet/Publications/books.html.

[54] M. Fussen, R. Wattenhofer, and A. Zollinger. Interference arises at the receiver. In *Wireless Networks, Communications and Mobile Computing, 2005 International Conference on*, volume 1, pages 427–432 vol.1, June 2005.

[55] E. Gafni. Round-by-round fault detectors (extended abstract): unifying synchrony and asynchrony. In *Proceedings of the Seventeenth Annual ACM Symposium on Principles of Distributed Computing*, pages 143–152, Puerto Vallarta, Mexico, 1998. ACM Press.

[56] E. Gafni and P. Kuznetsov. The weakest failure detector for solving k-set agreement. In *28th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing (PODC 2009)*, 2009.

[57] S. George, W. Zhou, H. Chenji, M. Won, Y. O. Lee, A. Pazarloglou, R. Stoleru, and P. Barooah. Distressnet: a wireless ad hoc and sensor network architecture for situation management in disaster response. *Communications Magazine, IEEE*, 48(3):128–136, March 2010.

[58] E. N. Gilbert. Random plane networks. *SIAM Journal*, 9(4):533–543, Dec. 1961.

[59] L. Girod, N. Ramanathan, J. Elson, T. Stathopoulos, M. Lukac, and D. Estrin. Emstar: A software environment for developing and deploying heterogeneous sensor-actuator networks. *ACM Trans. Sen. Netw.*, 3(3), Aug. 2007.

[60] A. Goiser, S. Khattab, G. Fassl, and U. Schmid. A new robust interference reduction scheme for low complexity direct-sequence spread-spectrum receivers: Performance. In *Proceedings 3rd International IEEE Conference on Communication Theory, Reliability, and Quality of Service (CTRQ'10)*, pages 15–21, Athens, Greece, June 2010.

[61] V. Grassi and F. L. Presti. Markov analysis of the prma protocol for local wireless networks. *Wireless Networks*, 4(4):297–306, 1998.

[62] B. Haeupler and D. Karger. Faster information dissemination in dynamic networks via network coding. In *ACM PODC*, pages 381–390, 2011.

[63] F. Harary and G. Gupta. Dynamic graph models. *Mathematical and Computer Modelling*, 25(7):79 – 87, 1997.

[64] J. M. Hendrickx and V. D. Blondel. Convergence of linear and non-linear versions of Vicsek's model. CESAME Research Report 2005.57, Université catholique de Louvain, Louvain-la-Neuve, 2005.

[65] J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis. Distributed anonymous function computation in information fusion and multiagent systems. In C. Beck and P. Viswanath, editors, *Proceedings of the 47th Allerton Conference on Communication, Control, and Computing*, pages 1582–1589. IEEE, New York, NY, 2009.

[66] J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis. Distributed anonymous discrete function computation. *IEEE Transactions on Automatic Control*, 56(10):2276–2289, 2011.

[67] M. Herlihy and N. Shavit. The asynchronous computability theorem for t-resilient tasks. In *STOC '93: Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, pages 111–120, New York, NY, USA, 1993. ACM.

[68] M. Hutle, D. Malkhi, U. Schmid, and L. Zhou. Chasing the weakest system model for implementing omega and consensus. *IEEE Transactions on Dependable and Secure Computing*, 6(4):269–281, 2009.

[69] R. Ingram, P. Shields, J. E. Walter, and J. L. Welch. An asynchronous leader election algorithm for dynamic networks. In *IPDPS*, pages 1–12, 2009.

[70] S. Janson, D. E. Knuth, T. Luczak, and B. Pittel. The birth of the giant component. *Random Structures Algorithms*, 4:233–358, 1993.

[71] W. Kiess and M. Mauve. A survey on real-world implementations of mobile ad-hoc networks. *Ad Hoc Netw.*, 5(3):324–339, Apr. 2007.

[72] T. Kontos, G. S. Alyfantis, Y. Angelopoulos, and S. Hadjiefthymiades. A topology inference algorithm for wireless sensor networks. *2013 IEEE Symposium on Computers and Communications (ISCC)*, 0:000479–000484, 2012.

[73] F. Kuhn, N. Lynch, C. Newport, R. Oshman, and A. Richa. Broadcasting in unreliable radio networks. In *Proceedings of the 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, PODC '10, pages 336–345, New York, NY, USA, 2010. ACM.

[74] F. Kuhn, N. A. Lynch, and R. Oshman. Distributed computation in dynamic networks. In *STOC*, pages 513–522, 2010.

[75] F. Kuhn and R. Oshman. Dynamic networks: Models and algorithms. *SIGACT News*, 42(1):82–96, 2011.

[76] F. Kuhn, R. Oshman, and Y. Moses. Coordinated consensus in dynamic networks. In *Proceedings of the 30th annual ACM SIGACT-SIGOPS symposium on Prin ciples of distributed computing*, PODC '11. ACM, 2011.

[77] F. Kuhn, S. Schmid, and R. Wattenhofer. Towards worst-case churn resistant peer-to-peer systems. *Distributed Computing*, 22(4):249–267, 2010.

[78] F. Legendre, T. Hossmann, F. Sutton, and B. Plattner. 30 years of wireless ad hoc networking research: What about humanitarian and disaster relief solutions? what are we still missing? In *International Conference on Wireless Technologies for Humanitarian Relief (ACWR 11)*, Amrita, India, 2011. IEEE.

[79] J. Li, L. L. Andrew, C. H. Foh, M. Zukerman, and H.-H. Chen. Connectivity, coverage and placement in wireless sensor networks. *Sensors*, 9(10):7664–7693, 2009.

[80] S. R. Mahaney and F. B. Schneider. Inexact agreement: Accuracy, precision, and graceful degradation. In *Proceedings 4th ACM Symposium on Principles of Distributed Computing*, pages 237–249, Minaki, Canada, Aug. 1985.

[81] J. Mao, Z. Wu, and X. Wu. A {TDMA} scheduling scheme for many-to-one communications in wireless sensor networks. *Computer Communications*, 30(4):863 – 872, 2007. Nature-Ispired Distributed Computing.

[82] M. Maróti, B. Kusy, G. Simon, and A. Lédeczi. The flooding time synchronization protocol. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, pages 39–49, New York, NY, USA, 2004. ACM.

[83] K. A. Marzullo. *Maintaining the Time in a Distributed System: An Example of a Loosely-Coupled Distributed Service*. PhD dissertation, Stanford University, Department of Electrical Engineering, Feb. 1984.

[84] H. Mehendale, A. Paranjpe, and S. Vempala. Lifenet: A flexible ad hoc networking solution for transient environments. *SIGCOMM Comput. Commun. Rev.*, 41(4):446–447, Aug. 2011.

[85] C. Metra, M. Omana, D. Rossi, J. Cazeaux, and T. Mak. The other side of the timing equation: a result of clock faults. In *20th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT 2005)*, pages 169–177, Oct. 2005.

[86] H. Moniz, N. F. Neves, M. Correia, and P. Veríssimo. Randomization can be a healer: Consensus with dynamic omission failures. In *Proceedings of the 23rd International Conference on Distributed Computing*, DISC'09, pages 63–77, Berlin, Heidelberg, 2009. Springer-Verlag.

[87] A. Mostéfaoui, M. Raynal, and J. Stainer. Relations linking failure detectors associated with k-set agreement in message-passing systems. In X. Défago, F. Petit, and V. Villain, editors, *SSS*, volume 6976 of *Lecture Notes in Computer Science*, pages 341–355. Springer, 2011.

[88] M. Nati, A. Gluhak, H. Abangar, and W. C. Headley. Smartcampus: A user-centric testbed for internet of things experimentation. In *WPMC*, pages 1–6, 2013.

[89] G. Neiger. Failure detectors and the wait-free hierarchy (extended abstract). In *Proceedings of the fourteenth annual ACM symposium on Principles of distributed computing*, PODC '95, pages 100–109, New York, NY, USA, 1995. ACM.

[90] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64:026118, Jul 2001.

[91] C. Newport, D. Kotz, Y. Yuan, R. S. Gray, J. Liu, and C. Elliott. Experimental Evaluation of Wireless Simulation Assumptions. *SIMULATION: Transactions of The Society for Modeling and Simulation International*, 83(9):643–661, Sept. 2007.

[92] M. D. Penrose. On k-connectivity for a geometric random graph. *Random Structures Algorithms*, 15(2):145–164, 1999.

[93] T. K. Philips, S. S. Panwar, and A. N. Tantawi. Connectivity properties of a packet radio network model. *IEEE Transactions on Information Theory*, 35(5):1044–1047, 1989.

[94] Y. A. Pignolet, S. Schmid, and G. Trédan. Misleading stars: what cannot be measured in the internet? *Distributed Computing*, 26(4):209–222, 2013.

[95] D. Powell. Failure mode assumptions and assumption coverage. In *Proc. 22nd IEEE Int. Symp. on Fault-Tolerant Computing (FTCS-22)*, pages 386–395, Boston, MA, USA, 1992. (Revised version available as LAAS-CNRS Research Report 91462, 1995).

[96] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves. Energy-efficient collision-free medium access control for wireless sensor networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, SenSys '03, pages 181–192, New York, NY, USA, 2003. ACM.

[97] S. Ramanathan and E. L. Lloyd. Scheduling algorithms for multihop radio networks. *IEEE/ACM Trans. Netw.*, 1(2):166–177, Apr. 1993.

[98] M. Raynal. Failure detectors to solve asynchronous k-set agreement: a glimpse of recent results. Technical report, pi-1959, INRIA, 2010.

[99] M. Raynal and J. Stainer. Synchrony weakened by message adversaries vs asynchrony restricted by failure detectors. In *Proceedings ACM Symposium on Principles of Distributed Computing (PODC'13)*, pages 166–175, 2013.

[100] M. Saks and F. Zaharoglou. Wait-free k-set agreement is impossible: The topology of public knowledge. *SIAM J. Comput.*, 29(5):1449–1483, 2000.

[101] P. Santi and D. M. Blough. The critical transmitting range for connectivity in sparse wireless ad hoc networks. *Mobile Computing, IEEE Transactions on*, 2(1):25–39, 2003.

[102] N. Santoro and P. Widmayer. Time is not a healer. In *Proc. 6th Annual Symposium on Theor. Aspects of Computer Science (STACS'89)*, LNCS 349, pages 304–313, Paderborn, Germany, Feb. 1989. Springer-Verlag.

[103] P. Sattari, C. Fragouli, and A. Markopoulou. Active topology inference using network coding. *Physical Communication*, 6:142–163, 2013.

[104] U. Schilcher, C. Bettstetter, and G. Brandner. Temporal correlation of interference in wireless networks with rayleigh block fading. *IEEE Transactions on Mobile Computing*, 11(12):2109–2120, 2012.

[105] U. Schmid. Random trees in queueing systems with deadlines. *Theoretical Computer Science*, 144(1-2):277–314, 1995.

[106] U. Schmid and K. Schossmaier. Interval-based clock synchronization. *Real-Time Systems*, 12(2):173–228, Mar. 1997.

[107] U. Schmid and K. Schossmaier. How to reconcile fault-tolerant interval intersection with the Lipschitz condition. *Distributed Computing*, 14(2):101 – 111, May 2001.

[108] U. Schmid, B. Weiss, and I. Keidar. Impossibility results and lower bounds for consensus under link failures. *SIAM Journal on Computing*, 38(5):1912–1951, 2009.

[109] M. Schwarz, K. Winkler, U. Schmid, M. Biely, and P. Robinson. Gracefully degrading consensus and k-set agreement under dynamic link failures. Research Report TUW-220473, Technische Universität Wien, Institut für Technische Informatik, Treitlstr. 1-3/182-2, 1040 Vienna, Austria, 2013. http://publik.tuwien.ac.at/files/PubDat_220473.pdf.

[110] M. Schwarz, K. Winkler, U. Schmid, M. Biely, and P. Robinson. Brief announcement: Gracefully degrading consensus and k-set agreement under dynamic link failures. In *Proceedings of the 33th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, PODC '14, New York, NY, USA, 2014. ACM. (to appear).

[111] A.-S. Tonneau, N. Mitton, and J. Vandaele. A survey on (mobile) wireless sensor network experimentation testbeds. In *Proceedings of the 2014 IEEE International Conference on Distributed Computing in Sensor Systems*, DCOSS '14, pages 263–268, Washington, DC, USA, 2014. IEEE Computer Society.

[112] N. H. Vaidya and D. K. Pradhan. Degradable agreement in the presence of Byzantine faults. In *International Conference on Distributed Computing Systems*, pages 237–244, 1993.

[113] G. Varghese and N. A. Lynch. A tradeoff between safety and liveness for randomized coordinated attack protocols. In *Proceedings of the 11th Annual ACM Symposium on Principles of Distributed Computing*, pages 241–250, Vancouver, British Columbia, Canada, August 1992.

[114] C. Ware, J. Judge, J. Chicharo, and E. Dutkiewicz. Unfairness and capture behaviour in 802.11 adhoc networks. In *2000 IEEE International Conference on Communications. ICC 2000. Global Convergence Through Communications.*, 2000.

[115] J. L. Welch and N. A. Lynch. A new fault-tolerant algorithm for clock synchronization. *Information and Computation*, 77(1):1–36, 1988.

[116] F. Xue and P. R. Kumar. The number of neighbors needed for connectivity of wireless networks. *Wireless Networks*, 10(2):169–181, Mar. 2004.

[117] J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292 – 2330, 2008.

[118] M. Zhang, M. C. Chan, and A. L. Ananda. Connectivity monitoring in wireless sensor networks. *Pervasive and Mobile Computing*, 6(1):112–127, 2010.

[119] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pages 187–198. ACM, 2004.

[120] P. Zielinski. Automatic classification of eventual failure detectors. In *Distributed Computing, 21st International Symposium, DISC 2007*, volume 4731 of *Lecture Notes in Computer Science*, pages 465–479, Lemesos, Cyprus, Sept. 2007. Springer Verlag.

[121] P. Zielinski. Anti-Ω: the weakest failure detector for set agreement. In *PODC '08: Proceedings of the twenty-seventh ACM symposium on Principles of distributed computing*, pages 55–64, New York, NY, USA, 2008. ACM.