# A Survey on Algorithmic Problems in Wireless Systems

Simon Thelen, Klaus Volbert and Danilo Weber Nunes
*Faculty of Computer Science and Mathematics, Ostbayerische Technische Hochschule Regensburg, Germany*

Keywords: Algorithms, WSNs, Survey, Network Construction, Routing, Interference, Localization, Charging, Latency.

Abstract: Considering the ongoing growth of Wireless Sensor Networks (WSNs) and the challenges they pose due to their hardware limitations as well as the intrinsic complexity of their interactions, specialized algorithms have the potential to help solving these challenges. We present a survey on recent developments regarding algorithmic problems which have applications in wireless systems and WSNs in particular. Focusing on the intersection between WSNs and algorithms, we give an overview of recent results inside this intersection, concerning topics such as routing, interference minimization, latency reduction, localization among others. Progress on solving these problems could be potentially beneficial for the industry as a whole by increasing network throughput, reducing latency or making systems more energy-efficient. We summarize and structure these recent developments and list interesting open problems to be investigated in future works.

## 1 INTRODUCTION

Wireless Sensor Networks (WSNs) have been steadily growing in significance over the past two decades and, in light of the ongoing shifts in the energy and industry sector, appear to continue to do so in the future. By 2025, the number of online Internet-of-Things (IoT) devices is expected to reach nearly 75 billion (Ikpehai et al., 2019). Nodes in a WSN commonly have many restrictions regarding their processing power, range of communication and their limited energy supply, often being battery-powered (Singh and Kumar, 2012). These limitations impose additional complications when designing solutions to known problems such as package routing, electromagnetic interference (EMI), latency, among others.

Meanwhile, there have been many interesting advances in the field of algorithms in recent years concerning problems such as maximum flow (Bläsius et al., 2021; Abboud et al., 2022), dominating set (Abu-Khzam et al., 2022), or graph coloring (Ghaffari and Kuhn, 2022; Ansari et al., 2022). It becomes evident that these algorithmic problems have applications in problems faced in the WSN field. Advances in the field of algorithms could lead to higher overall network efficiency due to better routing strategies. They could increase the network's throughput with improved network architectures or reduce its energy consumption by using more efficient algorithms, to name a few possible improvements. This overlap of WSNs and algorithms holds promising new advancements and is an area to be explored. To the best of our knowledge, there has not been a survey work combining both areas of research.

In this survey, we present algorithmic problems in wireless systems that researchers from both fields are currently working on. We examine where significant progress has been made and explore open questions and possible further developments. We have selected twenty papers published in the last five years which address interesting problems that in our opinion best represent the intersection between algorithms and WSNs (refer to Table 1).

This work is structured as follows: In Sections 1 to 9 we detail the findings of our survey grouped by topic. In Section 10 we summarize the work's conclusions and outline possible further research directions.

## 2 NETWORK CONSTRUCTION AND INITIALIZATION

For problems like routing or interference reduction, the network is often viewed as a pre-existing entity. In practice, though, the construction/deployment of the network can be equally important. Multiple recent results deal with the process of building up a network graph or extending an existing one.

Mertzios et al. analyzed a specific version of a growth process with possible applications for WSNs



Thelen, S., Volbert, K. and Nunes, D.
A Survey on Algorithmic Problems in Wireless Systems.
DOI: 10.5220/0011791200003399
In *Proceedings of the 12th International Conference on Sensor Networks (SENSORNETS 2023)*, pages 101-111
ISBN: 978-989-758-635-4; ISSN: 2184-4380

Table 1: A list of the papers which are part of this survey, grouped by topic and categorized by which algorithmic aspects they are predominately concerned with: creating theoretical models describing real-life problems in WSNs; presenting algorithms that solve these problems fully, approximately or heuristically; performing mostly theoretical analyses, for example regarding the hardness of specific problems.

| | | Modeling | Algorithms | Analysis |
|---|---|---|---|---|
| Network construction and initialization | (Mertzios et al., 2021) | ✓ | ✓ | ✓ |
| | (Corò et al., 2019) | ✓ | ✓ | ✓ |
| | (Miller, 2019) | ✓ | ✓ | |
| Routing | (Jung et al., 2019) | ✓ | ✓ | |
| | (Kuo, 2019) | ✓ | ✓ | ✓ |
| | (Götte et al., 2021) | ✓ | ✓ | |
| | (Galesi et al., 2019) | | | ✓ |
| Interference | (Abu-Affash et al., 2020) | ✓ | ✓ | ✓ |
| | (Tsai et al., 2019) | ✓ | ✓ | |
| | (Esperet et al., 2021) | | ✓ | ✓ |
| | (Schindelhauer et al., 2019) | ✓ | ✓ | |
| Localization | (Bose et al., 2020) | | ✓ | ✓ |
| | (Lv et al., 2019) | ✓ | ✓ | |
| Charging and energy harvesting | (Madhja et al., 2018) | ✓ | | ✓ |
| | (Hanschke and Renner, 2019) | ✓ | | |
| Reducing latency | (Karakostas and Kolliopoulos, 2022) | ✓ | ✓ | ✓ |
| | (de Berg et al., 2019) | ✓ | ✓ | |
| Population protocols | (Raskin, 2021) | ✓ | | ✓ |
| Drones | (Betti Sorbelli et al., 2022) | ✓ | ✓ | |
| | (Danilchenko et al., 2020) | ✓ | ✓ | |

where a graph grows from a starting vertex in a crystalline fashion and edges are allowed to be removed later (Mertzios et al., 2021) (excess edges). In each time slot, vertices and edges can be added according to specific rules and the process terminates if all vertices of the target graph and a super-set of the edges have been added. The authors recognize a trade-off between the number of time slots needed and the number of excess edges. They show that using poly-logarithmically many time slots, any tree can be grown needing $O(n)$ excess edges and any planar graph can be grown using $O(n \log n)$ excess edges. For the case of not allowing any excess edges, the authors prove that, assuming P $\neq$ NP, there is no polynomial-time algorithm that finds the shortest growth schedule and no polynomial-time approximation algorithm for finding the shortest growth schedule with an approximation ratio of $n^{1-\varepsilon}$ (for every $\varepsilon > 1$). There remain many open questions concerning the problem presented in the paper: It is unclear if the problem of growing a (general) graph using the fewest number of (excess) edges is NP-hard. The paper mainly focuses on the two extremes in terms of the number of excess edges, namely there being zero excess edges and there being lots ($\log(n)$), leaving the space between the two extremes largely unexplored.

A different kind of growth is known as *graph augmentation*. Here, given a graph, one should add the minimum number of edges to achieve a certain condition, like making the graph connected. This problem is not new and has already been studied in the 70s (Eswaran and Tarjan, 1976). Cor'o et al. analyzed a different variant of this problem where you are allowed to add a fixed number of edges to maximize the total gained connectivity, i.e. the number of reachable nodes, summed over all nodes (Corò et al., 2019). The authors present algorithms that solve a generalization of this problem optimally for trees and approximately for DAGs with one source (node with in-degree zero) and one sink (node with out-degree zero). They show that for general graphs with only one source and one sink the problem is NP-complete. They also show that in general this version of the problem is NP-hard to approximate within a factor of $1 - \frac{1}{\varepsilon}$. The authors are also interested in solving the problem for DAGs for multiple sources and sinks. It would also be interesting to solve a version of the problem where each added edge costs a specified amount and there is a limited budget.

For many problems in WSNs, such as routing, it is essential for nodes to know the nodes in their neighborhood beforehand. This problem is known as *neighbor discovery*. In a recent paper, Miller investigated multiple dynamic variants of *neighbor discovery*, being able to show that all of these can be reduced to a single problem, further named $\delta$-local gossip (Miller, 2019). For $\delta$-local gossip, every node in the network has a piece of information it wants to send to nearby nodes. After a certain time interval, each node $v$ should have transmitted its information to the nodes which were inside a $\delta$-multiple of $v$'s transmission range at some point during the execution of the algorithm. After all nodes have managed to do so, every node should terminate. Miller presents an algorithm that solves δ-local gossip for moving nodes on a line with bounded velocity. The author is interested in how a solution to this problem might be improved if the movement of nodes is partially known, which is common for practical applications like public transportation or self-driving vehicles. As the work shows the general applicability of δ-local gossip for neighbor discovery problems, investigating to which other kinds of problems δ-local gossip can be applied to might be an interesting topic for future research.

# 3 ROUTING

One of the most important problems in WSNs is how to route messages from one node to another.

## 3.1 Radio Holes and Ad Hoc Networks

Routing is especially challenging for dynamically changing ad hoc networks where the overhead of many classical routing approaches, which were designed for relatively static networks, is simply too high. Consequently, there has been a lot of algorithmic research concerning ”on-demand” or ”online” routing protocols. One issue of a greedy strategy like sending each message to the closest neighbor is that moving nodes can create radio holes, which might require some messages to take long detours to reach their destination (Jung et al., 2019). There have been multiple approaches to address this problem. Kuhn et al. presented a routing algorithm which is both fast in practice and proven to be asymptotically optimal in the worst-case in the sense that it achieves an asymptotically optimal competitive ratio compared to the best-possible offline algorithm under certain assumptions like reliance on local information only (Kuhn et al., 2003). Another result (Rührup and Schindelhauer, 2006) proves asymptotic optimality in a grid-like setting with possibly failed nodes.

A recent paper (Jung et al., 2019) approaches the problem of radio holes by, in addition to the regular short-range communication, giving nodes limited access to long-distance communication (e.g. satellite, cellular). They present an $O(\log^2 n)$ communication-round algorithm to compute an abstract version of the network (called an overlay network) which includes both the short- and long-distance connections and results in $c$-competitive routing paths for some constant $c$. Their algorithm heavily relies on the convex hulls of the radio holes in the network and assumes that the convex hulls do not overlap. It would be interesting to know how the problem can be solved if this restriction is lifted. The authors also mainly focus on the case of a static network, although they present a dynamic solution based on recomputing the overlay network periodically. It seems likely that this relatively simple method can be improved.

There is a strong algorithmic connection between routing and many classical graph-theoretic problems. Much attention in recent research has been on the DOMINATINGSET problem, which in its simplest form can be formulated as follows Given an undirected graph $G = (V,E)$, find the smallest vertex subset $S \subseteq V$, such that for every vertex $v \in V \setminus S$, there exists an adjacent vertex $v' \in S$. This NP-complete problem has useful applications for routing: As every node is adjacent to at least one node in the dominating set, routing can be simplified by only having nodes in the dominating set function as routers (Kuhn and Wattenhofer, 2003). A common variant of this problem is CONNECTEDDOMINATINGSET where the subgraph induced by the dominating set must be connected.

A connected dominating set in a network is especially useful for routing as it can be used as a *virtual backbone*: For any two nodes $s,t$, there exists a path from $s$ to $t$ such that each node on that path (except for possibly $s$ and $t$) is part of the virtual backbone. Motivated by this idea, a natural problem to study is: What is the smallest virtual backbone (connected dominating set) such that the number of internal nodes necessary to get from any node $s$ to any node $t$ increases at most by a factor of $\alpha$?. This is known as the *Connected Dominating Set Problem with Routing Cost Constraint* (CDR-$\alpha$). As discovered by (Ding et al., 2011), you do not actually need to consider all pairs $(s,t)$, but only those with a shortest distance of 2, resulting in the 1-DR-$\alpha$ problem: Any approximation algorithm for 1-DR-$\alpha$ applies CDR-$\alpha$ as well, retaining its approximation ratio. The difficulty of CDR-$\alpha$ largely depends on the choice of $\alpha$: For $\alpha = 1$, there exists an $O(\log n)$-approximation algorithm, by reducing the problem to SETCOVER (Ding et al., 2011). For sufficiently large $\alpha$, the problem can be reduced to regular CONNECTEDDOMINATINGSET, again yielding an $O(\log n)$-approximation algorithm (Guha and Khuller, 1998). So the question arises: Does there exist a polynomial-time $O(\log n)$-approximation algorithm for every $\alpha \geq 2$ (Du and Wan, 2013)? In a recent paper, Kuo was able to show that the answer to this questions is *no* by proving that no polynomial-time $2^{\log^{1-\varepsilon} n}$-approximation algorithm exists for any constant $\varepsilon > 0$, assuming that NP $\not\subseteq$ DTIME$(n^{\text{poly}\log n})$ (Kuo, 2019). However, the authors do present an $O\left(n^{1-\frac{1}{\alpha}}(\log n)^{\frac{1}{\alpha}}\right)$-approximation algorithm for any constant $\alpha > 1$ as well as an $O(\sqrt{n}\log n)$-approximation algorithm for $\alpha \geq 5$. They also show that 1-DR-$\alpha$ (and therefore

CDR-α) can be viewed as a special case of the SUBMODULARCOSTSETCOVER problem and the MINIMUMRAINBOWSUBGRAPH problem. If better approximation algorithms for those problems are found in the future, this might lead to better approximation algorithms for CDR-α.

An NP-complete problem which is closely related to DOMINATINGSET is SETCOVER. Here, a universe of elements $U$ and a set of subsets $S$ of $U$ is given ($\forall s \in S : s \subseteq U$) and the objective is to find the smallest number of sets in $S$ whose union equals $U$, i.e. these subsets cover $U$ completely. In terms of approximability, DOMINATINGSET and SETCOVER are in some sense almost the same problem as they can be converted into each other using L-reductions (Kann, 1992). Approximating these problems in a distributed setting has been studied extensively: e.g. DOMINATINGSET in (Kuhn and Wattenhofer, 2003; Jia et al., 2002) and SETCOVER in (Kuhn et al., 2006).

A recent paper (Götte et al., 2021) tackles SETCOVER in a distributed model where each node represents either an element of $U$ or of $S$ and edges represent $\in$-relations. Previous results focused on the LOCAL the CONGEST model where in each time step nodes can send distinct messages to all their neighbors. This new result considers two different models: In the $KT_0$-CONGEST model, each of the $n$ nodes can send one $O(\log n)$ size message per time step to some or all of its neighbors. In the BEEPING model, at every time step a node can either beep or receive. Receiving nodes only learn whether or not at least one neighboring node has beeped in the current time step. The $KT_0$-CONGEST algorithm presented achieves an expected approximation ratio of $O(\log \Delta)$, requiring $O(\log^2 \Delta)$ time and $\tilde{O}(\sqrt{\Delta} \cdot n)$ messages with high probability where $\Delta$ is the maximum node degree in the communication graph ($\tilde{O}$ hides poly-logarithmic factors). The BEEPING algorithm takes $O(k^3)$ for any chosen $k > 3$, achieving an approximation ratio of $O\left(\log^2(\Delta) \cdot \sqrt[k]{\Delta}^3\right)$. The parameter $k$ allows fine-tuning the trade-off between approximation ratio and runtime. The known lower bound for reaching an approximation ratio of $O(\sqrt[k]{\Delta})$ is $O(k)$ (Kuhn et al., 2006). If the problem can be solved in $o(k^3)$ with the same approximation ratio is still an open question.

## 3.2 Tomography

In order to route messages from one node to another, it is integral to know if all nodes along the way are up and running correctly. An interesting approach for doing this is *network tomography*, specifically *Boolean network tomography* where one only considers the paths that past messages were sent along. If one message does not reach its destination, one can deduce that at least one of the nodes along the path does not work correctly. Using multiple of these kinds of deductions, failed nodes can be identified. There has been lots of research concerning tomography (Kakkavas et al., 2020; Gray et al., 2020). Galesi et al. studied the influence of *vertex connectivity* on Boolean network tomography (Galesi et al., 2019). Vertex connectivity of a graph is defined as the minimum number of vertices which need to be removed until the graph is not connected anymore. The authors use the measure of *maximal identifiability* to describe how well Boolean tomography identifies nonworking nodes in a network. They prove tight bounds on the relationship between vertex connectivity and maximal identifiability for so-called Line of Sight networks and weaker bounds for general networks. They also extend these results to random graphs.

# 4 INTERFERENCE

Minimizing interference is clearly one of the most fundamental and most relevant problems in WSNs.

## 4.1 Optimizing Transmission Ranges

In theoretic works regarding interference minimization, researchers distinguish the asymmetric model (Agrawal and Das, 2013), where a node $u$ can send messages to some other node $v$ if and only if $v$ is inside $u$'s transmission range, and the symmetric model (Halldórsson and Tokuyama, 2008), where $u$ additionally has to be inside $v$'s transmission range.

In a widely studied interference-related problem, nodes are located at fixed positions in the 2D plane, and the task is to select the transmission range of each node to achieve minimum interference while retaining the connectedness of the network. In the *sender-centric* model (Moaveninejad and Li, 2005), the interference of each node is defined as the number of nodes in its transmission range and the network's interference is defined as the maximum node interference. The *receiver-centric* model (Von Rickenbach et al., 2009) defines the network's interference analogously in terms of receiving nodes. This receiver-centric notion seems to be the most widely accepted in the literature. While sender-centric interference can be minimized in polynomial time (Moaveninejad and Li, 2005) in both the symmetric and asymmetric model, minimizing receiver-centric interference is NP-hard (Buchin, 2008; Brise et al., 2014). Interference can also be modelled in terms of graph edges instead of vertices. In (Meyer auf der Heide et al., 2004), the interference of an edge $(u,v)$ is defined as the number of other edges whose endpoints lie within the transmission range of $u$ or of $v$.

Instead of minimizing the maximum node interference, some researchers focus on minimizing the sum of of all node interferences: *total interference*. Since the sum of all in-degrees in a graph is equal to the sum of all out-degrees, sender-centric and receiver-centric total interference is identical. Significant progress has been made in analyzing the total interference in the symmetric network model for both the 2D and the 1D case, where all nodes lie on a straight line. While the 1D case can be solved optimally in $O(n^4)$ time (Tan et al., 2011), the 2D case is NP-hard (Lam et al., 2010). However, for the 2D case, an $O(\log n)$-approximation algorithm exists (Moscibroda and Wattenhofer, 2005) as well as an algorithm that guarantees $O(\log n)$ total interference for all inputs (Fussen et al., 2005).

Abu-Affash et al. consider the problem of minimizing total interference in the asymmetric network model (Abu-Affash et al., 2020). Analogously to the previous results on the symmetric variant, this new paper shows that solving the 1D case optimally is feasible in polynomial time by presenting an $O(n^3)$ algorithm. The authors prove that the 2D case is NP-hard, but they do present a 2-approximation algorithm. They conjecture that techniques of their 1D algorithm could be used to improve the running time of the corresponding algorithm for the symmetric model.

In general, many questions regarding interference minimization still seem to not have satisfying answers. The problem of minimizing maximum node interference in the receiver-centric model seems to be particularly challenging. Apart from its NP-hardness, not much seems to be known about this problem.

## 4.2 Graph Coloring

Interference in WSNs can be reduced by using multiple frequencies/channels. This motivates the problem of selecting the right channel for any given message. In practice, nodes in a WSN often have multiple network interfaces and can consequently send/receive on multiple channels at the same time. To reduce interference, neighboring nodes can send/receive messages using different frequencies. The connection between problems regarding channel selection in WSNs and the graph theoretic problems EDGECOLORING and STRONGEDGECOLORING has long been known (Barrett et al., 2006). EDGECOLORING is the problem of coloring the edges of a graph, using as few colors as possible, such that no two adjacent edges have the same color. STRONGEDGECOLORING additionally requires that the endpoints for no pair of same-color edges are connected via a single edge. In the context of WSNs, these two graph problems are strongly related to problems of avoiding primary and secondary interference respectively. Although the NP-hardness of both EDGECOLORING (Holyer, 1981) and STRONGEDGECOLORING (Stockmeyer and Vazirani, 1982) has been proven, there is still active research concerning the two problems (Saberi and Wajc, 2021; Balliu et al., 2022). There are also many open questions regarding the strong variant. It has been conjectured that the optimal strong edge coloring of any graph uses at most $5\Delta^2/4$ colors where $\Delta$ is the maximum node degree (Halász and Sós, 1989). Despite some significant progress, this conjecture remains unproven (Deng et al., 2019).

Tsai et al. studied the *multi-channel assignment problem*. Here, each node in the network has $r$ radio interfaces which for each time slot can send/receive on one of $k$ possible channels (Tsai et al., 2019). The goal is to find a channel schedule for all nodes that optimizes a certain objective function. In contrast to previous results, which focused on maximizing the total number of connections per time slot or on minimizing the time needed for sending a given list of messages (Chaporkar et al., 2008; Kumar et al., 2004), Tsai et al. focus on minimizing latency. More specifically, they try to find a periodic schedule which minimizes $\max_{i \in E} T_i w_i$, where $E$ is the set of edges, $T_i$ is the maximum message delay of any message sent along edge $i$ for that schedule, and $w_i$ is edge $i$'s weight. The authors show that for unit weights and $r = 1$, their channel assignment problem reduces to EDGECOLORING for $k \geq \lfloor n/2 \rfloor$ and to STRONGEDGECOLORING for $k = 1$. In addition, they present an approximation algorithm for general weights which is based on the idea of duplicating edges depending on their weight. Simulations confirm that the proposed algorithms work well even if they are generalized to $r > 1$. In the future, this result might be extended to a more realistic interference model, such as SINR (Signal-to-Interference-plus-Noise Ratio), or even to other kinds of scheduling problems.

As seen above, channel selection has clear algorithmic connections to edge coloring problems. If we consider the one-channel-case, avoiding simultaneously sending, neighboring nodes has a surprising link to coloring a graph's vertices: If all nodes with color A send first, then all nodes with color B and so on, no two nodes within transmission distance of each other will send simultaneously. Esperet et al. tackled the problem of coloring a unit disk graph with as few colors as possible in a distributed setting with local communication, specifically in the LOCAL model where only the communication rounds count towards the running time (Esperet et al., 2021). In the case that nodes know their location, the authors present an algorithm which produces a $(3+\varepsilon)\omega(G)+6$-coloring in $O(1)$ rounds of communication for any constant $\varepsilon > 0$ where $\omega(G)$ is the size of the largest clique. For certain kinds of graphs (McDiarmid and Reed, 1999; McDiarmid, 2003), this is an improvement over a previously-known 3-approximation algorithm. In the case that nodes do not know their location, Esperet et al. demonstrate how to achieve a $5.68\omega(G)$-coloring in $O(\log^3 \log n)$ rounds with high probability and, assuming that $\omega(G) = O(1)$, how to achieve such a coloring in $O(\log^* n)$ rounds deterministically. Aditionally, the authors show that the average degree of any unit disk graph $G$ is at most $5.68\omega(G)$. They conjecture that this can be improved to $4\omega(G)$. The authors are interested in whether algorithms might exist that achieve $c\omega(G)$-colorings for a lower $c$ under different round constraints. Additionally, it might be interesting to know if similar techniques can be applied to general disk graphs where nodes can have different transmission ranges.

In general, there are many open questions in the field of graph coloring. For unit graphs in particular, we know that the minimum number of colors needed for coloring a graph is at most $3\omega(G)-2$, but little progress has been made in improving the constant 3 in this result (Esperet et al., 2021).

### 4.3 Constructive Interference

Although interference in wireless systems should usually try to be avoided, there exists the concept of constructive interference where multiple signals overlap in such a way that they amplify each other. Schindelhauer et al. study a problem where one node needs to broadcast a signal to every other node as quickly as possible using constructive interference (Schindelhauer et al., 2019). In particular, they are interested in the multiple input/multiple output model (MIMO) where multiple nodes can cooperate to produce a stronger signal. Instead of trying to broadcast a real message, the main focus lies on the collaboration aspect to achieve the signal strength needed so the signal can be detected in a noisy environment. The algorithm presented needs $O(\log\log n - \log\log \rho)$ rounds to broadcast a message where $n$ is the number of nodes and $\rho$ is the node density. The authors ignore interference effects which would arise when sending an actual message. Message encodings that would prevent these effects might be an interesting area of research. The authors also assume a path loss exponent of $\alpha = 2$ in their work. Despite some conjectures, the situation for larger values of $\alpha$ is left unexplored.

## 5 LOCALIZATION

Another interesting topic in wireless networks is localizing nodes. Global protocols like GPS guarantee accurate localization in many cases. However, they have relatively high power-consumption and do not work in indoor situations. Alternatively, the known location of some nodes (often called anchors) is used to determine the positions of all nodes. Multiple methods have been proposed, including triangulation (Savarese et al., 2001), multi-dimensional scaling (Ji and Zha, 2004) and trilateration (Moore et al., 2004) where, given the positions of two points of a triangle and the lengths of all three sides, the position of the third point can be deduced.

Bose et al. study the problem of localizing nodes in a unit-disk graph by starting at only three anchors and localizing other nodes using trilateration (Bose et al., 2020). Assuming RP $\neq$ NP where RP is the complexity class commonly known as ”Randomized Polynomial-time”, this localization problem is NP-hard even if it is known that there exists a unique solution (Aspnes et al., 2004). The authors therefore focus on localizing only some of the nodes. They define a node to be interior if every point on the boundary of its unit disk is covered by at least one other unit disk. A node is strongly interior if it and all its neighbors are interior. The authors propose a distributed algorithm that localizes at least all strongly interior nodes given three strongly interior anchor nodes, assuming all strongly interior nodes are connected. There are multiple ways to generalize their approach: What if the graph of all strongly interior nodes consists of multiple connected components? What if the transmission range of different nodes is different? etc.

Approaches like (Bose et al., 2020) rely on the assumption that distances between nodes can be determined accurately. In practice, these distances can only be approximated. But in some scenarios even that is not possible. The authors of (Lv et al., 2019) design a localization model, which they call *BSLoc*, based on telecommunication localization. It uses base stations of a telecommunication network (e.g. cellular), which the device is connected to. Typically, the radio signal strength indications (RSSI) of the base stations are used as a distance estimator to localize devices (Vaghefi et al., 2011; Zhu et al., 2016; Margolies et al., 2017). Lv et al. are interested in how to locate devices if these signal strengths are not known. They propose a two-level solution where a Hidden Markov Model in the first level produces a rough location estimate which is improved by a machine learning model in the second level. Their method uses both historical data as well as speed information of the devices and achieves comparable results to state-of-the-art RSSI-based methods in their experiments.

# 6 CHARGING AND ENERGY HARVESTING

Nodes in WSNs are usually battery-powered. To reduce maintenance, they are often capable of recharging their batteries. The authors of (Hanschke and Renner, 2019) study the scheduling of tasks which are performed on WSN nodes under time and energy constraints, with the additional complication that sensors are powered using energy harvesting, specifically using solar power. Sensors perform tasks like measuring humidity or fine dust with (time) dependencies between tasks, such as "to measure fine dust, a recent humidity measurement must exist". Sensors can charge their internal capacitor if the current energy is insufficient to complete the next task. Given an energy prediction for a future time period, the goal is to maximize the number of (periodically executed) tasks that are performed. Using a concept called *task graphs*, the authors solve the problem with an integer linear program solver. Using a simple performance comparison, the authors conjecture that their algorithm works even on low-end hardware under reasonable time constraints. They do not, however, provide an actual implementation to verify this claim.

Another concept in WSNs is Wireless Power Transfer. Madhja et al. investigated the situation of an ad hoc network consisting of multiple mobile agents, moving on a random-walk-like path, and a single static charging device with a specified finite energy supply, capable of charging the batteries of nearby agents (Madhja et al., 2018). The objective is to have the network up and running for as long as possible using the given energy supply. Unlike other publications, they allow for changing the charging power (and therefore charging range) dynamically. The energy received by each device is determined by a simplified version of the Friis transmission equation (Friis, 1946). The authors demonstrate the theoretical benefit of using an adaptive charging range by showing that any fixed charging range is sub-optimal for at least one possible agent scenario. They also show that, under slightly simplified conditions, finding the optimal charging range schedule offline is NP-hard. Then, they present several heuristic approaches to solving the problem online, showing the practicality of these approaches using experiments. It might be interesting to investigate if better results can be achieved by considering the predictable behavior of real-life agents (e.g. using machine learning).

# 7 REDUCING LATENCY

The paper (Tsai et al., 2019) mentioned in Section 4 deals with the problem of selecting radio channels to reduce latency. Interestingly, multiple other recent works have latency reduction as a common theme.

Karakostas and Kolliopoulos study a problem with applications in 2-way synchronization for Digital Twins (Karakostas and Kolliopoulos, 2022). For Digital Twins, data typically should be kept fresh, meaning as up-to-date as possible. One challenge when trying to achieve this is that critical resources like CPUs or are often shared by multiple Twins. If a Twin task takes $c$ clock cycles and the allowed time constraint equals $T$ cycles, then the total number of simultaneous tasks on that CPU cannot exceed $\frac{T}{c}$. The authors formulate a generalized version of this problem where $n$ tasks are distributed onto $m$ machines. For each task $i$ and machine $j$, a so-called *tolerance constraint* $u_{ij}$ is defined as follows: If task $i$ is assigned to machine $j$, then machine $j$ can have at most $u_{ij}$ tasks in total, including task $i$. The authors analyze two optimization variants of this problem: In the first variant, each job has a weight and the goal is to maximize the total weight of the assigned jobs assigned under the given constraints. The authors present a $(1-\frac{1}{\varepsilon})$-approximation algorithm for this variant. For the second variant, tolerance constraints can be loosened by a global scaling factor $\rho \geq 1$. The goal is to minimize $\rho$ so all tasks can be assigned. The authors prove that, assuming $P \neq NP$, there is no polynomial-time $(n^{1/2-\varepsilon})$-approximation for this variant, even if all machines share tolerance constraints. They prove even stronger hardness results for specific approaches, namely the configuration linear program (LP) and quadratic programming. The gap between the general hardness result and these specific hardness results is mentioned by the authors as an open problem.

Another latency-focused result deals with broadcasting messages. Broadcasting algorithms are often evaluated using metrics like throughput. De Berg et al. instead focus on *displacement*: How much do the packets of the message arrive out of order (de Berg et al., 2019)? For any node and any time step, the authors define displacement as the difference between the number of the latest packet received and the number of the earliest packet which has not been received yet. Low displacement can reduce buffer sizes as well as latency. The authors analyze different broadcasting algorithms that try to minimize displacement, showing that these algorithms tend to maximize throughput as well. However, which kinds of applications are most positively affected by displacement-focused broadcast algorithms still needs to be investigated.

# 8 POPULATION PROTOCOLS

Many methods and frameworks can be used to define how the elements in a network are connected and communicate. One of them is population protocols.

Population protocols were first proposed by (Angluin et al., 2004) and are a powerful and flexible communication and computing model, especially for mobile ad hoc networks (Aspnes and Ruppert, 2009). A population protocol consists of a collection of simple and mobile agents, which can assume a finite set of states. Like in a WSN, these agents are allowed to communicate if they are close. This communication is somewhat limited, as a scheduler determines which agents will interact. These interactions cause a state update for the two agents involved and are based on a defined transition table. Agents have a constant amount of memory to hold their state. An agent's movement is unpredictable but is subject to constraints to keep the system fair, in other words, all agents must have a chance to be selected for interaction. As all agents have the same set of states and are updated using the same set of rules, they can be considered to be virtually anonymous.

The interactions between the nodes carry the computation along. As shown by (Angluin et al., 2007b), in a setup where there are no restrictions on which agents can interact (i.e. fair scheduling), if the given predicates are *semilinear*, meaning that they are definable in first-order *Presburger arithmetic* (Presburger, 1929), these predicates are always computable by population protocols. This allows population protocols to perform quite powerful computations.

A population protocol is formally defined by the tuple ($Q$, $\Sigma$, $Y$, $\iota$, $\omega$, $\delta$), where $Q$ is the set of states, $\Sigma$ the set of input and $Y$ the set of output symbols, $\iota : \Sigma \to Q$ the function that maps the input to an internal state, $\omega : Q \to Y$ the function that maps the agent's internal state to an output and $\delta : Q \times Q \to Q \times Q$ the function that defines the state transitions of two interacting agents. Eventually the output of all agents will converge to the same value. This structure defines a very generic and flexible framework. Different tasks, network topologies, communication protocols, and other aspects of a WSN can be modelled and even integrated in the network itself. Due to their simplicity, population protocols can be implemented using essentially just three lookup tables. This allows them to run on even extremely low-powered hardware.

Consider the following IO population protocol: $\Sigma$=$Y$=$Q$=$\{0,?,1\}$, $\iota$ and $\omega$ are the identity function, $\delta = \{(0,0) \to (0,0),\ (?,0) \to (0,0),\ (1,0) \to (?,0), (0,?) \to (0,?),\ (?,?) \to (?,?),\ (1,?) \to (1,?), (0,1) \to (?,1),\ (?,1) \to (1,1), (1,1) \to (1,0)\}$. In $O(n \log n)$ interactions, this simple population protocol is able to determine with high probability which state (1 or 0) the majority of the agents were in at the beginning of the execution (Angluin et al., 2007a). Populations protocols can also be applied to more complex tasks, such as counting. By defining $\delta = \{(a,b) \to (\lfloor \frac{a+b}{2} \rfloor, \lceil \frac{a+b}{2} \rceil)\}$, the population protocol can not only determine the majority state, but also by what margin it was the majority in $O(n \log n)$ interactions (Mocquard et al., 2015).

Michail and Spirakis studied another important, real-world application of population protocols. By allowing each agent to store not only its current state, but also its connection state, it enables populations protocols to eventually generate a stable spanning star network topology (Michail and Spirakis, 2014). The method works under dynamic conditions, enabling a network to adapt over time in response to an increased network load and thus improving its total capacity.

Message loss can also be added to the agent's communication dynamic by simply adding the possibility of only one agent updating its local state and the state of the other agent involved in the communication remaining unchanged, as if the communication did not happen (Raskin, 2021).

The combination of the last two works could lead to interesting further developments not only for the field of population protocols but also to the broader WSN field, improving dynamic topology generation and the network's resiliency to network instabilities.

# 9 DRONES

Multiple recent works deal with algorithmic problems motivated by applications for aerial drones.

Sorbelli et al. considered a smart agriculture scenario where sensors are deployed in an area (Betti Sorbelli et al., 2022). Instead of using a multi-hop approach to transmit the data to a depot, a drone is used. This drone starts at the depot, flies across the area, collecting data from the sensors, and returns to the depot. The drone's storage and energy supply is limited. Some data like images might have a higher priority or weight. The authors define the *Single-drone Data-collection Maximization Problem* as finding the drone "mission" (consisting of a path and the data to collect) which maximizes the mission's reward, i.e. the sum of the collected data's weights. The authors show that the problem is NP-hard. They also present an approximation algorithm, based on known approximation algorithms for KNAPSACK, MINIMUMSETCOVER and TSP, as well as two simple heuristic solutions. All three algorithms perform well on synthetically generated data. Interesting extensions to this problem include multiple drones or a more realistic model for the drone's energy. It might be interesting to know in which scenarios this unusual drone setup is advantageous.

Danilchenko et al. studied the (NP-hard) problem of covering points in the 2D plane with a fixed number circular disks or squares of identical size, maximizing the weighted sum of all covered points (Danilchenko et al., 2020). As motivation, the authors name is a surveillance application where a small and fixed number of aerial drones with a certain view radius cover a large number of ground users. The authors study both the static case and the dynamic case under user additions and deletions. Two models are considered: In the case that all drones can communicate, the authors use a known PTAS for the static version (Jin et al., 2018; Khuller et al., 2014) to solve the dynamic version with approximition ratio 4 for disks and 7 for squares, needing $O(n\log n)$ initiliaztion time and $O(\log n)$ update time. For the second model, drones can only communicate if they are within a distance of $R_{COM}$, a constant parameter, under the $\ell_1$ norm. Assuming the number of drones is increased from $m$ to $O(m\sqrt{m})$, the authors present an $O(1)$-approximation algorithm dynamic case with the same running times.

## 10 CONCLUSIONS

The fields of WSNs and algorithms are constantly evolving. Many advances have been made in recent years, not only by solving old problems but also by creating entire new fields of research. In this survey, we presented some important challenges and current developments in algorithmic problems with potential applications in WSN systems. We also detailed possible further research paths that have the potential to significantly advance and improve current WSNs.

Particularly interesting are in our opinion the developments concerning interference reduction (Section 4) and population protocols (Section 8). There are many open interference-related problems which are worth exploring, including minimizing receiver-centric maximum interference as well as some open questions concerning vertex and edge coloring in a graph. Population protocols present an interesting approach to model a wireless system. We believe them to have many practical applications left to explore.

In conclusion, there is a clear connection between advances in algorithmic research and WSNs improvements and we believe it should be further explored.

## ACKNOWLEDGEMENTS

This work was supported by the Regensburg Center of Energy and Resources (RCER). Further information under www.rcer.de.

## REFERENCES

Abboud, A., Krauthgamer, R., Li, J., Panigrahi, D., Saranurak, T., and Trabelsi, O. (2022). Breaking the cubic barrier for all-pairs max-flow: Gomory-hu tree in nearly quadratic time. FOCS.

Abu-Affash, A. K., Carmi, P., and Katz, M. J. (2020). Minimizing total interference in asymmetric sensor networks. In *ALGOSENSORS*, pages 1–16.

Abu-Khzam, F. N., Fernau, H., Gras, B., Liedloff, M., and Mann, K. (2022). Enumerating minimal connected dominating sets. In *ESA 2022*.

Agrawal, P. and Das, G. K. (2013). Improved interference in wireless sensor networks. In *ICDCIT*, pages 92–102.

Angluin, D., Aspnes, J., Diamadi, Z., Fischer, M. J., and Peralta, R. (2004). Computation in networks of passively mobile finite-state sensors. In *Proc. of the 23rd ACM PODC*, pages 290–299.

Angluin, D., Aspnes, J., and Eisenstat, D. (2007a). A simple population protocol for fast robust approximate majority. In *Distributed Computing*, pages 20–32.

Angluin, D., Aspnes, J., Eisenstat, D., and Ruppert, E. (2007b). The computational power of population protocols. *Distributed Computing*, 20(4):279–304.

Ansari, M., Saneian, M., and Zarrabi-Zadeh, H. (2022). Simple streaming algorithms for edge coloring. In *ESA 2022*.

Aspnes, J., Goldenberg, D., and Yang, Y. R. (2004). On the computational complexity of sensor network localization. In *ALGOSENSORS*, pages 32–44.

Aspnes, J. and Ruppert, E. (2009). *An Introduction to Population Protocols*, pages 97–120.

Balliu, A., Brandt, S., Kuhn, F., and Olivetti, D. (2022). Distributed edge coloring in time polylogarithmic in $\Delta$. In *PODC 2022*.

Barrett, C. L., Istrate, G., Kumar, V. A., Marathe, M. V., Thite, S., and Thulasidasan, S. (2006). Strong edge coloring for channel assignment in wireless radio networks. In *PERCOMW'06*, pages 5–pp.

Betti Sorbelli, F., Navarra, A., Palazzetti, L., Pinotti, C., and Prencipe, G. (2022). Optimal and heuristic algorithms for data collection by using an energy-and storage-constrained drone. In *ALGOSENSORS*.

Bläsius, T., Friedrich, T., and Weyand, C. (2021). Efficiently Computing Maximum Flows in Scale-Free Networks. In *ESA 2021*.

Bose, K., Kundu, M. K., Adhikary, R., and Sau, B. (2020). Distributed localization of wireless sensor network using communication wheel. In *ALGOSENSORS*, pages 17–31.

Brise, Y., Buchin, K., Eversmann, D., Hoffmann, M., and Mulzer, W. (2014). Interference minimization in asymmetric sensor networks. In *ALGOSENSORS*, pages 136–151.

Buchin, K. (2008). Minimizing the maximum interference is hard. *CoRR*, abs/0802.2134.

Chaporkar, P., Kar, K., Luo, X., and Sarkar, S. (2008). Throughput and fairness guarantees through maximal scheduling in wireless networks. *IEEE Transactions on Information theory*, 54(2):572–594.

Corò, F., D'Angelo, G., and Pinotti, C. M. (2019). On the maximum connectivity improvement problem. In *ALGOSENSORS*, pages 47–61.

Danilchenko, K., Segal, M., and Nutov, Z. (2020). Covering users by a connected swarm efficiently. In *ALGOSENSORS*, pages 32–44.

de Berg, M., Carstens, C. J., and Mandjes, M. (2019). Throughput and packet displacements of dynamic broadcasting algorithms. In *ALGOSENSORS*, pages 158–174.

Deng, K., Yu, G., and Zhou, X. (2019). Recent progress on strong edge-coloring of graphs. *Discrete Mathematics, Algorithms and Applications*, 11(05):1950062.

Ding, L., Wu, W., Willson, J., Du, H., Lee, W., and Du, D.-Z. (2011). Efficient algorithms for topology control problem with routing cost constraints in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 22(10):1601–1609.

Du, D.-Z. and Wan, P.-J. (2013). Routing-cost constrained cds. In *Connected Dominating Set: Theory and Applications*, pages 119–131.

Esperet, L., Julliot, S., and de Mesmay, A. (2021). Distributed coloring and the local structure of unit-disk graphs. In *ALGOSENSORS*, pages 61–75.

Eswaran, K. P. and Tarjan, R. E. (1976). Augmentation problems. *SIAM Journal on Computing*, 5(4):653–665.

Friis, H. (1946). A note on a simple transmission formula. *Proc. of the IRE*, 34(5):254–256.

Fussen, M., Wattenhofer, R., and Zollinger, A. (2005). Interference arises at the receiver. In *WiMob 2005*, volume 1, pages 427–432.

Galesi, N., Ranjbar, F., and Zito, M. (2019). Vertex-connectivity for node failure identification in boolean network tomography. In *ALGOSENSORS*, pages 79–95.

Ghaffari, M. and Kuhn, F. (2022). Deterministic distributed vertex coloring: Simpler, faster, and without network decomposition. In *2021 IEEE 62nd Annual Symposium on FOCS*, pages 1009–1020.

Götte, T., Kolb, C., Scheideler, C., and Werthmann, J. (2021). Beep-and-sleep: Message and energy efficient set cover. In *ALGOSENSORS*, pages 94–110.

Gray, C., Mosig, C., Bush, R., Pelsser, C., Roughan, M., Schmidt, T. C., and Wahlisch, M. (2020). Bgp beacons, network tomography, and bayesian computation to locate route flap damping. In *Proc. of the ACM Internet Measurement Conference*, pages 492–505.

Guha, S. and Khuller, S. (1998). Approximation algorithms for connected dominating sets. *Algorithmica*, 20(4):374–387.

Halász, G. and Sós, V. T. (1989). *Irregularities of Partitions*, chapter "Problems", pages 161–165. Springer.

Halldórsson, M. M. and Tokuyama, T. (2008). Minimizing interference of a wireless ad-hoc network in a plane. *Theoretical Computer Science*, 402(1):29–42.

Hanschke, L. and Renner, C. (2019). Time- and energy-aware task scheduling in environmentally-powered sensor networks. In *ALGOSENSORS*, pages 131–144.

Holyer, I. (1981). The np-completeness of edge-coloring. *SIAM Journal on computing*, 10(4):718–720.

Ikpehai, A., Adebisi, B., Rabie, K. M., Anoh, K., Ande, R. E., Hammoudeh, M., Gacanin, H., and Mbanaso, U. M. (2019). Low-Power Wide Area Network Technologies for Internet-of-Things: A Comparative Review. *IEEE Internet of Things Journal*, 6(2):2225–2240.

Ji, X. and Zha, H. (2004). Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling. In *IEEE INFOCOM 2004*, volume 4, pages 2652–2661.

Jia, L., Rajaraman, R., and Suel, T. (2002). An efficient distributed algorithm for constructing small dominating sets. *Distributed Computing*, 15(4):193–205.

Jin, K., Li, J., Wang, H., Zhang, B., and Zhang, N. (2018). Near-linear time approximation schemes for geometric maximum coverage. *Theoretical Computer Science*, 725:64–78.

Jung, D., Kolb, C., Scheideler, C., and Sundermeier, J. (2019). Competitive routing in hybrid communication networks. In *ALGOSENSORS*, pages 15–31.

Kakkavas, G., Gkatzioura, D., Karyotis, V., and Papavassiliou, S. (2020). A review of advanced algebraic approaches enabling network tomography for future network infrastructures. *Future Internet*, 12(2):20.

Kann, V. (1992). *On the approximability of NP-complete optimization problems*. PhD thesis, Royal Institute of Technology Stockholm.

Karakostas, G. and Kolliopoulos, S. G. (2022). Resource time-sharing for iot applications with deadlines. In *ALGOSENSORS*, pages 131–144.

Khuller, S., Purohit, M., and Sarpatwar, K. K. (2014). Analyzing the optimal neighborhood: Algorithms for budgeted and partial connected dominating set problems. In *Proc. of the 25th annual ACM-SIAM SODA*, pages 1702–1713.

Kuhn, F., Moscibroda, T., and Wattenhofer, R. (2006). The price of being near-sighted. In *Proc. of the 17th Annual ACM-SIAM SODA*, page 980–989.

Kuhn, F. and Wattenhofer, R. (2003). Constant-time distributed dominating set approximation. In *Proc. of the 22nd ACM PODC*, pages 25–32.

Kuhn, F., Wattenhofer, R., and Zollinger, A. (2003). Worst-case optimal and average-case efficient geometric ad-hoc routing. In *Proc. of the 4th ACM MOBIHOC*, pages 267–278.

Kumar, V. A., Marathe, M. V., Parthasarathy, S., and Srinivasan, A. (2004). End-to-end packet-scheduling in wireless ad-hoc networks. In *Proc. of the 15th annual ACM-SIAM SODA*, pages 1021–1030.

Kuo, T.-W. (2019). On the approximability and hardness of the minimum connected dominating set with routing cost constraint. In *ALGOSENSORS*, pages 32–46.

Lam, N. X., Nguyen, T. N., and Huynh, D. T. (2010). Minimum total node interference in wireless sensor networks. In *ADHOCNETS*, pages 507–523.

Lv, J., Zhao, Q., Li, J., Zhang, Y., Di, X., Rao, W., Yuan, M., and Zeng, J. (2019). Bsloc: Base station id-based telco outdoor localization. In *ALGOSENSORS*, pages 206–219.

Madhja, A., Nikoletseas, S., and Voudouris, A. A. (2018). Mobility-aware, adaptive algorithms for wireless power transfer in ad hoc networks. In *ALGOSENSORS*, pages 145–158.

Margolies, R., Becker, R., Byers, S., Deb, S., Jana, R., Urbanek, S., and Volinsky, C. (2017). Can you find me now? Evaluation of network-based localization in a 4g lte network. In *IEEE INFOCOM 2017*, pages 1–9.

McDiarmid, C. (2003). Random channel assignment in the plane. *Random Structures & Algorithms*, 22(2):187–212.

McDiarmid, C. and Reed, B. (1999). Colouring proximity graphs in the plane. *Discrete Mathematics*, 199(1-3):123–137.

Mertzios, G. B., Michail, O., Skretas, G., Spirakis, P. G., and Theofilatos, M. (2021). The complexity of growing a graph. *CoRR*, abs/2107.14126.

Meyer auf der Heide, F., Schindelhauer, C., Volbert, K., Grünewald, M., et al. (2004). Congestion, dilation, and energy in radio networks. *Theory of Computing Systems*, 37(3):343–370.

Michail, O. and Spirakis, P. G. (2014). Simple and efficient local codes for distributed stable network construction. In *Proc. of ACM PODC 2014*, pages 76–85.

Miller, A. (2019). Local gossip and neighbour discovery in mobile ad hoc radio networks. In *ALGOSENSORS*, pages 1–14.

Moaveninejad, K. and Li, X.-Y. (2005). Low-interference topology control for wireless ad hoc networks. *Ad Hoc Sens. Wirel. Networks*, 1(1-2):41–64.

Mocquard, Y., Anceaume, E., Aspnes, J., Busnel, Y., and Sericola, B. (2015). Counting with population protocols. In *IEEE NCA 2015*, pages 35–42.

Moore, D., Leonard, J., Rus, D., and Teller, S. (2004). Robust distributed network localization with noisy range measurements. In *Proc. of SenSys 2004*, pages 50–61.

Moscibroda, T. and Wattenhofer, R. (2005). Minimizing interference in ad hoc and sensor networks. In *Proc. of the 2005 joint workshop on Foundations of mobile computing*, pages 24–33.

Presburger, M. (1929). Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. In *Comptes-Rendus du ler Congres des Mathematiciens des Pays Slavs*.

Raskin, M. (2021). Population protocols with unreliable communication. In *ALGOSENSORS*, pages 140–154.

Rührup, S. and Schindelhauer, C. (2006). Online multi-path routing in a maze. In *Algorithms and Computation*, pages 650–659.

Saberi, A. and Wajc, D. (2021). The greedy algorithm is not optimal for on-line edge coloring. In *ICALP 2021*.

Savarese, C., Rabaey, J., and Beutel, J. (2001). Location in distributed ad-hoc wireless sensor networks. In *2001 IEEE ICASSP*, volume 4, pages 2037–2040.

Schindelhauer, C., Oak, A., and Janson, T. (2019). Collaborative broadcast in $O(\log\log n)$ rounds. In *ALGOSENSORS*, pages 119–136.

Singh, M. and Kumar, P. (2012). An efficient forward error correction scheme for wireless sensor network. *Procedia Technology*, 4:737–742.

Stockmeyer, L. J. and Vazirani, V. V. (1982). Np-completeness of some generalizations of the maximum matching problem. *Information Processing Letters*, 15(1):14–19.

Tan, H., Lou, T., Wang, Y., Hua, Q.-S., and Lau, F. C. (2011). Exact algorithms to minimize interference in wireless sensor networks. *Theoretical Computer Science*, 412(50):6913–6925.

Tsai, S.-Y., Yang, H.-T., Liu, K. S., Lin, S., Chowdhury, R., and Gao, J. (2019). Multi-channel assignment and link scheduling for prioritized latency-sensitive applications. In *ALGOSENSORS*, pages 137–157.

Vaghefi, R. M., Gholami, M. R., and Ström, E. G. (2011). Rss-based sensor localization with unknown transmit power. In *2011 IEEE ICASSP*, pages 2480–2483.

Von Rickenbach, P., Wattenhofer, R., and Zollinger, A. (2009). Algorithmic models of interference in wireless ad hoc and sensor networks. *IEEE/ACM Transactions on Networking*, 17(1):172–185.

Zhu, F., Luo, C., Yuan, M., Zhu, Y., Zhang, Z., Gu, T., Deng, K., Rao, W., and Zeng, J. (2016). City-scale localization with telco big data. In *Proc. of the 25th ACM International CIKM*, pages 439–448.