CrossMark

Upper and lower bounds for deterministic broadcast in powerline communication networks

Yvonne Anne Pignolet[1] · Stefan Schmid[2] · Gilles Tredan[3]

Abstract Powerline communication networks assume an interesting position in the communication network space: similarly to wireless networks, powerline networks are based on a shared broadcast medium; unlike wireless networks, however, the signal propagation is constrained to the power lines of the electrical infrastructure, which is essentially a *graph*. This article presents an algorithmic model to study the design of communication services over powerline communication networks. As a case study, we focus on the fundamental *broadcast problem*, and present and analyze a distributed algorithm COLORCAST which terminates in at most n communication rounds, where n denotes the network size, even in a model where link qualities are unpredictable and time-varying. For comparison, the achieved broadcast time is lower than what can be achieved by any unknown-topology algorithm (lower bounds $\Omega(n \log n / \log(n/D))$ and $\Omega(n \log D)$ are proved in Kowalski and Pelc (Distrib Comput 18(1):43–57, 2005) resp. Clementi et al. (2001) where D is the network diameter). Moreover, existing known-topology broadcast algorithms often fail to deliver the broadcast message entirely in this model. This article also presents a general broadcast lower bound for the powerline model.

✉ Stefan Schmid
stefan@net.t-labs.tu-berlin.de; schmiste@cs.aau.dk

Yvonne Anne Pignolet
yvonne-anne.pignolet@ch.abb.com

Gilles Tredan
gtredan@laas.fr

[1] ABB Corporate Research, Zurich, Switzerland

[2] Aalborg University, Aalborg, Denmark

[3] CNRS-LAAS, Toulouse, France

1 Introduction

Broadcast refers to the basic task of transmitting a single message originating from some source node s to all $n − 1$ remaining nodes $V \setminus \{s\}$. This fundamental problem has been studied in many settings, from wireless networks consisting of nodes with omnidirectional antennas, to wireline networks with point-to-point communication. An interesting and less well-understood communication network type gains importance with the modernization of the electrical grid infrastructure: powerline communication (*PLC*) networks. PLC is used by utilities to control and manage the electrical grid without building an additional (fiber or wireless) communication infrastructure. The powerlines carrying power from producers to consumers can be used to disseminate information as well by adding a modulated carrier signal to the wiring system. While powerline communication has been used on point-to-point links for many years in the electrical grid's high voltage backbone network, multi-hop low and medium voltage PLC networks can enable "smart grid" functionalities in the electrical distribution grid [19,29]. Smart grid applications are envisioned to use medium voltage PLC networks for control and monitoring, e.g., current and voltage values at transformers as well as the health status of electrical grid equipment are to be collected; and control commands and settings are to be distributed to switches, circuit breakers and transformers.

This article initiates the study of the broadcast problem in a medium voltage PLC network. For instance, broadcasts are useful in the context of *adaptive protection*: in order to ensure the reliable and efficient operation of electrical grids, adaptive protection changes the settings of the protection equipment depending on the network load, capacity and configuration. Thus, broadcast communication services are needed to disseminate currently valid settings. Broadcast

in powerline networks also constitutes a challenging algorithmic problem. Similarly to wireless networks, the signal propagation is typically subject to various types of interference and uncertainties. Indeed, today there is no *network model* for the analysis of media access control (MAC) or higher-layer protocols for medium voltage PLC.[1] In PLC, the achievable communication quality between two devices varies depending on the powerline paths between them and the current electrical conditions. While studies have shown that in medium voltage networks, the packet success rate is strongly dependent on the distance between the two nodes that communicate and potential concurrent transmission from other senders, many other static and dynamic factors such as the quality of the line, electrical switches, circuit breakers, transformers and loads influence packet transmission. As a consequence, although the powerline communication infrastructure is known, the uncertainty on the effective transmission ranges at runtime translates into a partly unknown communication (and interference) topology.

1.1 Our contribution

We study reliable broadcast algorithms in powerline networks, and present and discuss a novel graph model capturing important characteristics of the powerline communication channel. We put our PLC model into perspective with respect to existing communication network models, and introduce a variant where transmission ranges are unknown to the nodes and may change over time. We present a broadcast time complexity lower bound in this model, based on the diameter, the network size n and the level of uncertainty on the transmission range of the nodes.

We present and analyse the distributed and deterministic algorithm COLORCAST to solve the PLC broadcast problem. We prove an asymptotically optimal broadcast time of at most n (the "runtime" of the algorithm is measured in communication rounds). More specifically, we derive a non- asymptotic bound which gives hard and deterministic performance guarantees and which depends on the chromatic number ξ of the interference graph; generally, ξ can be much smaller than n. We show that COLORCAST is asymptotically better than the best possible broadcast algorithm for unknown topologies. Moreover, it is easy to see that existing known-topology algorithms often fail when faced with uncertainty.

We also report on our simulation study on a Swiss medium voltage grid topology, and compare COLORCAST to a heuristic and randomized approach. Our results suggest that COLORCAST does not only provide good worst-case guarantees on the broadcast time complexity, but also performs well in realistic scenarios (Sect. 6).

---

[1] For other models, mostly targeted at low voltage use cases and for communication channel models, we refer to [5,6,28,31,32].

1.2 Organization

The remainder of this article is organized as follows. Section 2 introduces our formal model, and Sect. 3 puts our model into perspective with respect to related models arising in the context of wired and wireless networks. Section 4 highlights the challenge of the broadcast problem and reports on lower bounds. Section 5 presents our algorithm and analysis, and in Sect. 6, we present simulation results. After reviewing related work in Sect. 7, we conclude in Sect. 8.

2 PLC network model with and without uncertainty

2.1 Powerline communication between two nodes

Given a powerline and two PLC modems u and v attached to it, u can only communicate with v if the *length of the powerline* connecting u and v is not too large. Note that therefore two closely located PLC modems u and v cannot necessarily communicate: they also need to be linked by a short enough powerline. Apart from this distance, the communication quality of this link depends on many factors, such as electromagnetic interference and impedance effects from electric appliances. We model this time-dependent quality as a varying noise level $\rho^t(u, v) \in [0, \rho_{\max}]$. Note that we do not require that $\rho^t(u, v) = \rho^t(v, u)$ to account for the fact that communication links in PLC are not necessarily symmetric.

The larger distance d and the higher the noise level ρ, the less likely is a successful message reception. To be able to model this relation in a general way, we introduce the notion of a *link quality* function:

Definition 1 (*Link quality function*) Consider a function

$$f : \mathbb{R}^+ \times [0, \rho_{max}] \to \mathbb{R}^+$$
$$(d, \rho) \mapsto f(d, \rho).$$

Function f is a *link quality function* iff it is monotonically increasing in both d and ρ: $\frac{\partial f}{\partial d} > 0$.
We refer to $f(d, \rho)$ as the *virtual distance* of d under ρ.

By adjusting to the noise level, this model allows us to account for fading and time-varying behavior for channel attenuation, in addition to the influence of the link length. In other words, f can be used to compute a weight for a communication link at time t which determines its current link quality. We can interpret this weight as a *virtual distance* of this link. As scaling the link quality function with a constant factor does not violate the conditions, we assume without loss

of generality that u can communicate with v successfully at time t if $f(d, \rho^t) < 1$.[2]

2.2 Powerline communication networks

We represent the wiring of the electrical grid used for powerline communication as a weighted graph, where the n nodes represent the communication devices and the edges represent the powerlines connecting them as well as their distances, see Fig. 1. The communication devices (nodes) of the PLC infrastructure can communicate by *local broadcasts*, reaching a certain set of other devices. This set depends on: (i) the length of the powerlines between them, (ii) the current link qualities and (iii) simultaneous transmissions.

Formally, we model a PLC infrastructure topology as a weighted graph with PLC modems as nodes and powerline links, where the weights represent the length of the links. Figure 1b depicts an example.

Definition 2 (*PLC infrastructure topology I*) A PLC infrastructure topology is a graph $I = (V, E, d)$ which connects nodes V along powerline links $E \subset V \times V$. The length of edge $e = (u, v)$ is denoted by $d(e)$. Analogously to the noise level of a single link, we assigned a (time-varying) noise level $\rho^t(e) \in [0, \rho_{\max}]$ to each edge $e \in E$.

The (temporal) *communication graph* $G_{com}^t = (V, E_{com}^t)$ features links to nodes which can communicate at a certain time t. Thanks to the generic definition of the noise level and the link quality function, the node transmission range can be normalized to one unit. That is, we assume that a message reaches all nodes within virtual distance of at most 1.

Definition 3 (*Communication graph G_{com}*) Let $I = (V, E, d)$ be a weighted symmetric graph representing a powerline infrastructure, and let $\rho : E \to [0, \rho_{max}]$ be a noise distribution on I. The resulting communication graph G_{com} given a link quality function f satisfies

1. $V(G_{com}) = V$
2. $(i, j) \in E(G_{com})$ iff there exists a path P from i to j in I such that $\sum_{(v,w) \in P} f(d(v, w), \rho(v, w)) \le 1$.

We define $G_{com}^\perp$ to be the "worst case" communication graph where noise levels are maximal, and $G_{com}^\top$ to be the "best case" when the noise level is minimal on all edges. We denote the corresponding edge sets by $E^\perp$ and $E^\top$: $(v_i, v_j) \in E^\perp \Leftrightarrow j$'s message will always reach i, even in the worst case when $\rho(e) = \rho_{\max} \forall e$, and $(v_i, v_j) \in E^\top \Leftrightarrow j$'s message can reach i in ideal conditions with $\rho(e) = 0 \forall e$.

---

[2] Concurrent transmissions might lead to interference and prevent correct message reception. This case is treated in the subsequent section.
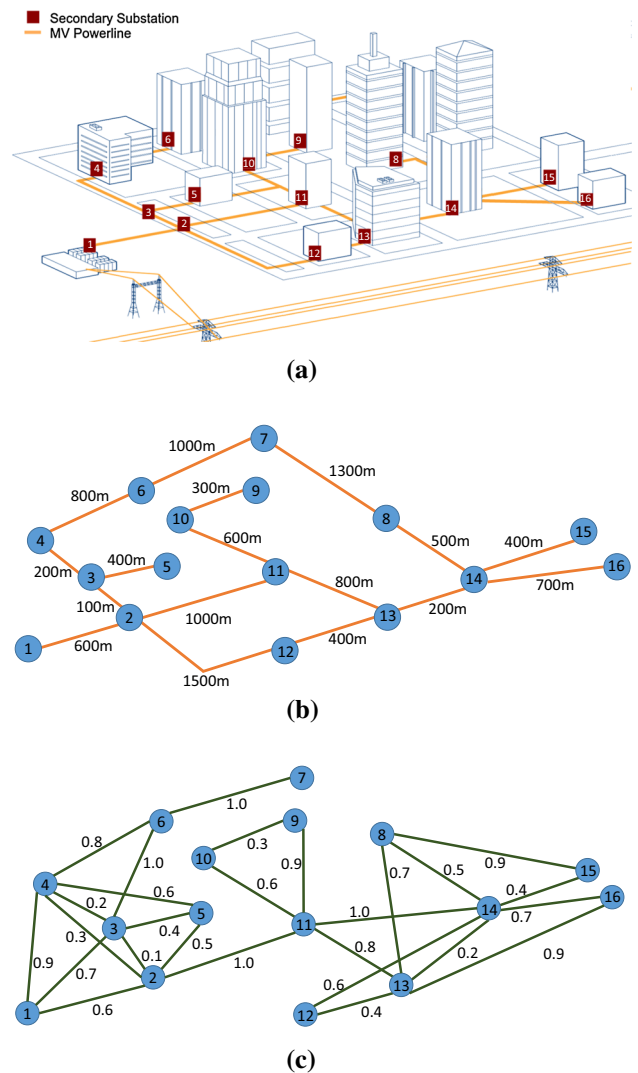

Fig. 1 Model illustration: **a** Electrical distribution grid: secondary substations transforming medium voltage into low voltage (*red boxes*) and the powerlines (*orange lines*) connecting them. **b** Graph I: PLC infrastructure topology model with communication devices at secondary substations and lengths of power lines. **c** Communication graph representation example G_{com}^t, where $\rho^t(u, v) = 0$ for all $u, v \in V$ and link quality function $f(d, \rho) = d/1000m + \rho$, i.e., nodes can communicate if there is a path of at most 1000 m between them (color figure online)

Note that the edge set of G_{com} does not have to be a subset or superset of G. Depending on ρ and f there are cases where G_{com} is the same as G and there are cases where it has fewer or more edges than G. In particular, nodes which are physically close to each other, but not connected by a short powerline path cannot communicate. Yet, it is not necessary to have a direct and separate powerline (an edge in the infrastructure graph) for each edge of the communication graph: signals can be received over multiple hops in the infrastructure graph, as long as the path between the nodes is short enough.

Since powerlines form a shared medium where concurrent transmissions can collide, we state the conditions that need to be met to guarantee a successful transmission.

Definition 4 (*Conditions for successful concurrent transmissions*) A node v_j receives the message sent from v_i at round t if

1. nodes v_i and v_j are in communication range in this round: $(v_i, v_j) \in E_{com}^t$
2. node v_i is the only node in v_j's range to send a message

If another node v_k in the communication range sends concurrently, i.e., $(v_k, v_j) \in E_{com}$, then the two messages might interfere and v_j may or may not be able to receive any of v_i and v_j's messages.[3] In other words, we study a simple, binary interference model, where signals travel along the graph structure of the powerline network, and we assume that a message arriving via different paths at a node does not interfere with itself.

We address the broadcast problem on PLC networks in this article.

Definition 5 (*PLC broadcast problem* **PBC**) The broadcast problem on $I(V, E, d)$, where some node $s \in V$ (the broadcast *root* or *source*) needs to send a message to all other nodes $V \setminus \{s\}$ and the temporal noise level distribution ρ^t is unknown is called **PBC**.

A synchronous environment is assumed in which time proceeds in discrete rounds: a message transmitted in round t is received in the same round. At $t_0 = 0$, the source s transmits the message and we want to minimize the time t until all nodes V have successfully received the message.

To guarantee that the broadcast can terminate successfully regardless of the network conditions, we require the worst case scenario to still offer a solution. In other words, we only consider scenarios where $G_{com}^\perp$ is connected. If $G_{com}^\perp$ is not connected, no solution to the broadcast is possible if $\rho^t = \rho_{max}, \forall t$.

We assume that the nodes know the PLC infrastructure I in advance. This assumption makes sense in a smart grid scenario where the devices are static and designed to be in operation for decades. Nodes know an upper bound on the noise level $\rho_{\max}$, and the function f. What is unknown to the nodes is the current noise level $\rho^t, \forall t$ and hence the resulting current communication graph.

---

[3] We do not assume that we can always predict what happens if there are multiple concurrent senders in range. It depends on the received signal strengths and the available hardware if messages can be decoded in this case. Complexity-wise it is harder to solve problems in this model. See also the discussion on http://www.wisdom.weizmann.ac.il/~oded/p_bgi.html.

In the remainder of this article, we will refer to the powerline broadcast problem with unknown link qualities by **PBC**. We will measure the total number of communication rounds used by an algorithm in the worst case. Concretely, the time complexity is the time when each node has received the message.

3 Putting the powerline model into perspective

In contrast to wireless models where the signal is typically assumed to spread in "free space", the signal in the powerline network is bound to spread along the electrical grid infrastructure, such as overhead wires or underground cables. In this section, we have a first closer look at the communication graphs of PLC and also put these graphs into perspective with related graphs known in wireless settings, such as Unit-Disk Graphs or r-restricted graphs.

Let us start with the Unit Disk Graph. *Unit-Disk Graph (UDG)* model [7,9,14]. The specific structure of the corresponding propagation and interference models can be exploited in the algorithm design of efficient protocols.

While many properties are similar in wireless and powerline communication, i.e., with respect to interference from simultaneous transmissions, we show that the communication topologies for power line communication are more general than the ones produced by, e.g., Unit Disk Graphs.

To enable a simpler comparison with geometric graph models (capturing free-space signal propagations), we restrict ourselves to PLC infrastructures with an embedding in the two dimensional Euclidean space.

Let $\mathcal{G}_{UDG}$ be the set of "UDG-compliant" communication graphs, in the sense that any graph $G \in \mathcal{G}_{UDG}$ can result from an embedding $\mu : V \to \mathbb{R}^2$ mapping nodes v to a position $\mu(v)$ in the Euclidean plane, and where two nodes v and w are connected iff $d_2(\mu(v), \mu(w)) \leq 1$, where d_2 represents the 2D Euclidean distance.

To compare PLC communication graphs against UDG communication graphs, we restrict ourselves to situations where the weighted infrastructure graph $I = (V, E, d)$ is a representation of the electrical infrastructure's 2D layout.

Definition 6 (*Euclidean embeddings of $I(V, E, d)$, $com(\mu, G, \rho)$ and $\mathcal{G}_{PLC}$*) Given $I(V, E, d)$, a valid PLC embedding $\mu : V \to \mathbb{R}^2$ satisfies that $\forall (v, w) \in E, d(v, w) = d_2(\mu(v), \mu(w))$. As a shorthand, given an unweighted graph G, an embedding μ (which defines the link lengths implicitly), and a noise distribution ρ, the resulting 2-D embedded communication graph is denoted by $com(\mu, G, \rho)$. Let $\mathcal{G}_{PLC}$ denote the set of all possible 2-embedded PLC communication graphs: $\mathcal{G}_{PLC} = \{com(\mu, G, \rho) : \forall G, \forall \mu, \forall \rho\}$, and let $\mathcal{G}_{PLC_0}$ be the

set of all possible communication graphs ignoring noise:
$\mathcal{G}_{PLC_0} = \{com(\mu, G, 0) : \forall G \; \forall \mu\}$.

The following lemma shows that the set of UDGs is a strict subset of PLC communication graphs.

Lemma 1 *The set of possible 2-D embedded PLC communication graphs, even if restricted to $\rho = 0$, is a strict superset of the set of possible UDGs, i.e., $\mathcal{G}_{UDG} \subsetneq \mathcal{G}_{PLC_0}$.*

Proof We first show $\mathcal{G}_{UDG} \subseteq \mathcal{G}_{PLC_0}$. Let G be a UDG graph resulting from an embedding μ of $|V(G)| = n$ nodes. Let $d(e)$ be the length of the edge $e = (u, v)$ in this embedding, i.e., the distance between nodes u and v in the Euclidean plane according to their positions. We now construct a powerline graph G' with the same edge set as G, by placing a powerline of length $d(u, v)$ between u and v. Hence $com(\mu, G', 0) = G = G'$ and $\mathcal{G}_{UDG} \subseteq \mathcal{G}_{PLC_0}$ (Fig. 2).

It remains to prove that there are graphs which are not UDGs. Let W_{14} be a 14-*wheel graph*: a star graph consisting of 15 nodes (1 central node and 14 leaves) whose leaves are additionally connected in a circular manner. It is easy to see that $W_{14} \notin \mathcal{G}_{UDG}$: for the neighborhod of the center node there exists an independent set of cardinality larger than six: a contradiction to the unit disk assumption. However, $W_{14} \in \mathcal{G}_{PLC_0}$. Let A, B, C be the positions in the Euclidean plane of an equilateral triangle of side length 3/4. Let μ be the following embedding: $\mu(1) = A$, $\mu(v) = B$ if v is even, $\mu(v) = C$ otherwise. Consider $com(\mu, W_{14}, 0)$: since $d(B, C) = d(A, B) = d(A, C) < 1$, $E(W_{14}) \subset E(com(\mu, W_{14}), 0)$. Since $2d(A, B) = 2d(B, C) = 2d(A, C) = d(A, B) + d(A, C) > 1$, no two nodes of the parity can communicate together, thus $E(W_{14}) = E(com(\mu, W_{14}), 0)$. Therefore $W_{14} \in \mathcal{G}_{PLC_0}$. ∎

Some algorithms developed for UDGs can still be applied in $\mathcal{G}_{PLC_0}$ networks. To ensure they work as planned, there must exist an embedding into the Euclidean plane of the PLC network $G(V, E, d)$ meeting two conditions: (i) the UDG edge set induced by the embedding must be equal to E and (ii) all length constraints d must be satisfied. Otherwise,
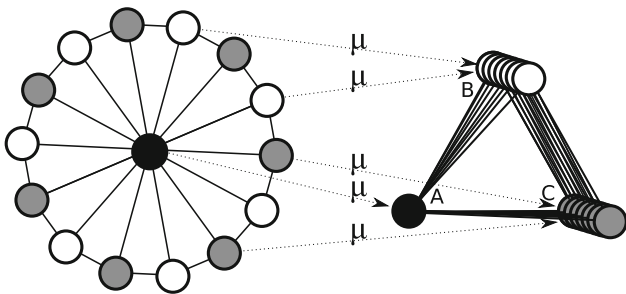
however, an algorithm may fail as it relies on the bounded independence and/or on the fact that all nodes in at most unit distance can hear each other. Both of these assumptions do not hold in general for $\mathcal{G}_{PLC_0}$. Algorithms for the UDG generalizations of Unit Ball Graphs (more than two dimensions) and Quasi-Unit Disk Graphs (QUDGs), allowing for a grey zone where communication links may or may not exist [26], can also be applied for PLC networks if the respective generalizations of the conditions mentioned above are met. Related results in prior work are discussed in more detail in Sect. 7.

To identify further subclasses of $\mathcal{G}_{PLC}$, the technique of the previous proof can be generalized. For this, let us introduce the definition of the *chromatic contraction* of a graph. The concept will simplify the task of finding a PLC embedding of any unweighted graph G.

Definition 7 (*Chromatic Contraction (CC) Graph*) Let G be an arbitrary graph. G_C is the *Chromatic Contraction (CC) graph* of G if and only if there exists a proper coloring $\xi : V(G) \mapsto V(G_C)$ of the graph G: nodes of the same color in G are mapped to the same node in G_C, and two color nodes are connected in G_C if the corresponding nodes are connected in G, i.e., $\forall (v, w) \in E(G), \; (\xi(v), \xi(w)) \in E(G_C)$.

In other words, the chromatic contraction graph is a compact representation of the original graph with respect to a coloring, where all the nodes sharing the same color in the original graph are grouped into a single node of the CC graph, while the CC graph edges preserve the relations of the nodes in the original graph.

We need the following lemma. Intuitively, a chromatic contraction defines an embedding in the plane where we collocate or "stack" nodes of the same color. Nodes of the same color never share a link, also no link exists between nodes of the same "color stack". Moreover, since the minimal distance between two stacks is at least 1/2, link lengths are between 1/2 and 1: the transmission can never bridge two hops.

Lemma 2 *Let $CC(G)$ be the set of Chromatic Contraction Graphs of G. If $\exists G_C \in CC(G)$ such that $G_C \in \mathcal{G}_{PLC}$ with an embedding μ satisfying $\forall v, w \in V(G_C), d(\mu(v), \mu(w)) > 1/2$ then $G \in \mathcal{G}_{PLC}$.*

Proof By construction, let $G_C \in \mathcal{G}_{PLC}$ be the chromatic contraction graph of G. Since G_C is in $\mathcal{G}_{PLC}$, let μ be an embedding respecting the UDG constraints. Since $G_C \in CC(G)$, let ξ be an arbitrary legal coloring of G. Let $\mu' : V(G) \mapsto \mathbb{R}^2$ be the embedding such that $\mu'(v) = \mu(\xi(v))$.

We need to show that $com(\mu', G, 0) = G$. Assume $com(\mu', G, 0) = G' \neq G$. Since $V(G) = V(com(\mu', G, 0)) = V(G')$, graphs can only differ by their links. Let $M = E(G) \setminus E(G')$ be the set of missing communication links, and let $X = E(G') - E(G)$ be the set of extra communication links.

Fig. 2 Illustration of the embedding μ constructed in Lemma 1: μ maps W_{14} to an equilateral triangle ABC. For clarity the μ on some vertices is omitted, and nodes stacked on B and C are slightly spread

Assume $M \neq \emptyset$, and let $(v, w) \in M$. Since (v, w) is in the infrastructure graph, but not in the communication graph, it must hold that $d_2(\mu'(v), \mu'(w)) > 1$. This contradicts the definition of G_C since $(v, w) \in E(G_C)$ and therefore $d_2(\mu(\xi(v)), \mu(\xi(w))) \leq 1$. Thus there cannot exist any missing links.

Assume $X \neq \emptyset$, and let $(u, w) \in A$. Since $(u, w) \in E(G'), d_{G'}(u, w) \leq 1$. But since $(u, w) \notin E(G)$, we deduce that u and w are not directly connected. Let us consider the shortest path between u and w. The length of this path is at least two. If the path is exactly two, in which case there exists a node v such that $(u, v) \in E(G), (v, w) \in E(G)$ and $d_2(\mu'(u), \mu'(v)) + d_2(\mu'(v), \mu'(w)) \leq 1$. Since $(u, v) \in E(G), (v, w) \in E(G), \xi(v) \neq \xi(u)$ and $\xi(v) \neq \xi(w)$. Therefore $d_2(\mu'(u), \mu'(v)) + d_2(\mu'(v), \mu'(w)) > 1/2 + 1/2 = 1$: no such connection can exist, as communication links have distance at most 1. Clearly, no such connections can exist for longer paths neither. Hence there cannot be any additional links and hence $com(\mu', G, 0) = G$. This concludes the proof. □

Using Lemma 2, we can show that PLC communication graphs include large graph families.

Theorem 1 *All five-colorable graphs, short $\mathcal{G}_5$, are PLC communication graphs: $\mathcal{G}_5 \subsetneq \mathcal{G}_{PLC}$*

Proof Five colorable graphs have a chromatic contraction graph G_C containing of at most five nodes. Consider a regular pentagon of side length $s = 1/2 + \epsilon$. The diagonal length is $2s \cos \pi/10 \approx 1.91s < 1$. Let μ be the function mapping each node of a five node graph to one vertex of the pentagon graph; μ satisfies: $\forall v, w \in V(G_C) \times V(G_C), d_2(\mu(v), \mu(w)) > 1/2$. Observe that for any graph G with $|V(G)| = 5, com(\mu, G, 0) = G$: all 5-node graphs are in $\mathcal{G}_{PLC}$. Thus we can apply Lemma 2 to derive that $\mathcal{G}_5 \subset \mathcal{G}_{PLC}$. Finally, observe that $K_6 \notin \mathcal{G}_5$ and $K_6 \in \mathcal{G}_{PLC}$. □

As a direct consequence of Theorem 1, any planar graph can be a powerline communication graph: planar graphs are 4-colorable.

Corollary 1 *Let $\mathcal{P}$ be the set of planar graphs: $\mathcal{P} \subsetneq \mathcal{G}_{PLC}$*

This shows that the set $\mathcal{G}_{PLC}$ forms an interesting super class of unit disk graphs and planar graphs. Moreover $\mathcal{G}_{PLC}$ includes graphs of unbounded neighbor independence: an example is the k-wheel graph which is in $\mathcal{G}_{PLC}$ for any k, since it is 3-colorable regardless of k.

Another interesting model which has recently received attention is the *r-restricted model* [17]: this model defines a set of unreliable edges which may or may not be usable and all edges in this set connect nodes of hop-distance at most r. Note that this is a superset of $\mathcal{G}_{PLC}$ for the special case where

all distances in the PLC infrastructure graph are set to 1, with f and ρ_{max} chosen adequately (for instance $f(d, \rho) = (d + \rho)/r$ and $\rho_{max} = r - 1$). In r-restricted graphs it is possible that a node cannot receive a message originating from hop-distance two, while one of its neighbors (at hop-distance three from the sender) might receive it. In $\mathcal{G}_{PLC}$ receiving a message at distance r' successfully implies that all nodes at distance less than r' from the sender can decode it as well (unless there are collisions). Apart from this, $\mathcal{G}_{PLC}$ can be seen as a generalization of r-restricted graphs to a more detailed model using edge lengths instead of hop counts.

4 The challenge of unknown link quality

Unknown noise conditions render the broadcast problem significantly more difficult, even if there is no temporal variation.

Theorem 2 *There exist problem instances where the broadcast time of any (optimal) deterministic algorithm ALG is $\Omega(n)$ times higher than without uncertainty.*

Proof Recall that we can choose an arbitrary link quality function f which is monotonic in the link distance and the noise level, assigning links a weight at each point in time, which can be interpreted as a virtual distance. Let $\check{\ell} \in (0, 1]$ denote a constant distance where $f(\check{\ell}, \rho_{max}) < 1$ and analogously, let $\hat{\ell} \in (0.5, 1]$ denote a distance where $f(\hat{\ell}, \rho_{max}) > 1$. For the proof it is sufficient to focus on link lengths $\check{\ell}$ and $\hat{\ell}$ only.

Consider the powerline graph G depicted in Fig. 3a. The broadcast source s is connected to k nodes (denoted by V_1) which in turn are connected to the sink t (the value of k will be determined later). In addition we create a path of $n - k - 2$ nodes called V_2 from s to t and connect all nodes to t directly as well. All links incident to s, the links among the set V_2
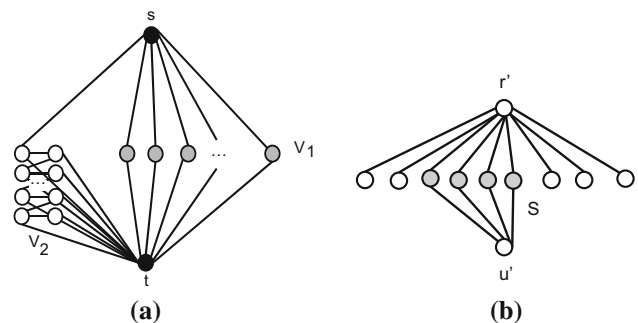

Fig. 3 The lower bound powerline graph G for PBC is depicted in (**a**). The nodes in V_2 are of degree 3 each, as they form a path from s to t among each other and are connected to t directly in addition. The broadcast problem can be reduced to an unknown radio network graph depicted in (**b**)

and the link from the last node of V_2 to t are of length $\check{\ell}$, all others of length $\hat{\ell}$. This ensures that the communication graph is connected, even under worst case noise conditions. As a next step we describe how the noise is assigned to these links.

We construct a noise assignment which guarantees that there is an interference-free path of length $|V_2|$ from s to t using the nodes in V_2. Concretely, we only use a binary noise value of 0 and $\rho_{\max}$, and assume that the following edges do not experience any noise: the edges between s and V_1, the edges among V_2 nodes, the edges from t to V_2 nodes, the edge from s to the first node of V_2, and the edge from the lowest node of V_2 and t. We assign $\rho_{\max}$ to all other edges from V_2 to t. This assignment is known to the algorithm; the only uncertainty concerns the edges between V_1 nodes and t, where the algorithm does not know which edges are free of noise and which are experiencing a noise level of $\rho_{\max}$.

We note that scheduling any subset of V_1 nodes for concurrent transmissions can result in one of two outcomes: if exactly one node of the set is incident to an edge without noise, then the message can reach t. Otherwise, if there is no node incident to a noise-free edge, then node t cannot be reached; also, if two or more nodes with noise-free edges transmit simultaneously their messages will collide. The last two cases cannot be distinguished by the algorithm.

To deliver a message from V_1 to t, an algorithm must select a set with exactly one node incident to a noise-free edge. In this scenario there is at least one path of length two from s to t, and if the algorithm knew one link between a node $v \in V_1$ and t, then it could devise a schedule of length three (s, v, t): Node s sends the message in slot one, v in slot two and t in slot three, completing the broadcast. Hence, an algorithm solving broadcast in time $T < |V_2|$ on this graph G must find subsets $S_1, S_2, \ldots S_T$ of nodes in V_1 which send simultaneously in round i, hoping that exactly one of the nodes in S_i has an edge without noise to t, and all others are noisy (*success condition*). If the algorithm has chosen such a set, then node t can forward the message to all nodes in the subsequent time slot. In contrast, broadcasting along the long V_2 path from v to t, takes $\Omega(\check{\ell}(n-k))$.

We now show that any algorithm solving **PBC** on this graph in $o(T)$ time can be used to solve broadcast on unknown radio networks on graphs C_T for $T+2$ nodes [4] in time $o(T)$. Since the time complexity for this problem is $\Omega(T)$ this leads to a contradiction. A member of C_l defined by a set $S \subseteq \{1, 2, \ldots, l\} = [l]$ is a graph with one node s' connected to all nodes with IDs in $[l]$. The nodes of S are connected to another node u'. The problem of radio broadcast without knowing the topology thus reduces to choosing a subset M of the nodes $[l]$ that contains exactly one node of S without knowing S. Reference [4] demonstrates a lower bound of $\Omega(l)$ time slots to pick such a set M (without collision detection). The fact that the noise level between V_1 and t is uncertain corresponds to the situation where $C_{|V_1|}$ is unknown. The time complexity is thus $\Omega(\min(k, \check{\ell}(n-k)))$ which is maximized for $k = \lceil \check{\ell}n/(1+\check{\ell}) \rceil$. Thus this topology and noise level assignment is a scenario where any deterministic algorithm has a broadcast time of at least $\Omega(n)$ times more than the effective diameter: a constant. □

5 Deterministic broadcast upper bounds

We now present an algorithm COLORCAST (Algorithm 1) which solves the broadcast problem in powerline networks, even for unknown link qualities. The basic idea of the COLORCAST algorithm is to use a coloring on the interference graph to compute a schedule without any collisions.

COLORCAST seeks to conservatively avoid collisions by scheduling two nodes u and v which may interfere at some node w if $(u, w) \in E^\top \land (v, w) \in E^\top$ in different rounds. This is achieved by computing the following coloring-based schedule. First, an arbitrary *Minimal (Connected) Dominating Set* (MDS) is computed on $G_{com}^\perp$. The MDS ensures connectivity in the sense that any two dominators are within each other's communication range, even in the worst case. Subsequently, starting from the source, COLORCAST computes a breadth-first spanning tree on $G_{com}^\perp$, the communication graph under low transmission ranges, and divides the tree into layers L_i of increasing distances. Then COLORCAST computes for each layer L_i separately, a minimal coloring on the dominator nodes of the MDS in the L_i-induced subgraph of $G_{com}^\top$ *of high transmission range*, henceforth simply denoted by $G(L_i)$. Let ξ_i denote the chromatic number of $G(L_i)$ and $\xi = \sum_{i=1} \xi_i$ the sum of the chromatic number over all layers.

By this layer coloring, COLORCAST *avoids collisions* entirely: each color constitutes an independent set on the interference graph, and the nodes cannot interfere, even if some or all nodes have a better-than-worstcase transmission range ($\rho^t < \rho_{max}$).

Figure 4 illustrates the layering and coloring of the COLORCAST Algorithm.

First, let us note that all nodes that can exchange information in normal conditions can do so in ideal conditions: $\forall t, E_{com}^t \subset E^\top$. To see this, let $e \in E_{com}^t$. We have $f(d(e), \rho^t(e)) \leq 1$ since e exists. Thanks to the monotonicity of f in ρ, we have: $d(e) = f(d(e), 0) \leq f(d(e), \rho^t(e)) \leq 1$.

We first prove that the algorithm is collision free, and then prove its correctness. Finally we prove the complexity.

Lemma 3 COLORCAST *(Algorithm 1) produces a collision-free schedule.*

Algorithm 1 COLORCAST $(G(V, E, d), s \in V, G_{com}^{\perp}, G_{com}^{\top})$

1: $D \leftarrow \text{CDS}(G_{com}^{\perp})$
2: $T \leftarrow$ a spanning tree of the $G_{com}^{\perp}$ subgraph induced by $D \cup \{s\}$
3: let $L_i = \{v \in T | d_T(v, s) = i\}, \forall i \in [0, \max_{v \in T} d_T(v, s)]$
4: let R be an integer array of size $|V|$ initialized at $\bot$
   /* round assignment */
5: $round \leftarrow 0$;
6: **for each** L_i **do**
7:   $G(L_i) \leftarrow (L_i, E_i = \{ (E^{\top} \times E^{\top}) \cap L_i\})$
8:   let $\Xi_i : L_i \rightarrow [1, \xi_i]$ be a coloring of $G(L_i)$
9:   **for each** $v \in L_i$ **do**
10:    $R[v] \leftarrow round + \Xi_i(v)$
11:   $round \leftarrow round + \xi_i$
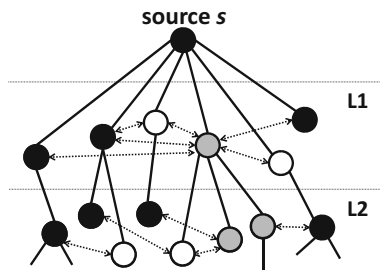12: **return** schedule R

source s

Fig. 4 Visualization of COLORCAST. The algorithm structures nodes along layers (here: two layers), starting from the source s and at low-range intervals. Each layer is colored, as indicated by the different node colors (*black*, *grey*, *white*). The spanning tree on the connected dominating set is shown in solid lines, while interference edges (for the layer coloring with respect to maximal interference) are *dotted*. Communication links are not shown explicitly in this figure

Proof Let $G(V, E, d)$ be a powerline network, and $s \in V$ the broadcast source. Let R be the transmission schedule produced by COLORCAST, i.e., the binary variable $R[i, t] \in \{0, 1\}$ indicates whether node v_i transmits in round t. The proof is by contradiction: Assume such a collision occurs at some node w during round t. Let v_i and v_j two of the nodes responsible for this collision: $R[i, t] = R[j, t] = 1$. Since w experienced a collision we know that $(v_i, w) \in E_{com}^t \land (v_j, w) \in E_{com}^t$. Thus, $(v_i, w) \in E^{\top} \land (v_j, w) \in E^{\top}$ and therefore $(v_i, v_j) \in (E^{\top} \times E^{\top})$. First observe that if $R[i, t] = R[j, t] = 1$, because of the update of $round$ in Line 12, then necessarily v_i and v_j belong to the same layer. Let L be this layer, and Ξ the corresponding coloring obtained in Line 9. Because of Line 11, necessarily $\Xi(i) = \Xi(j)$. Since Ξ is a legal coloring of $G(L)$, we conclude that $(v_i, v_j) \notin E_i$. Contradiction. $\square$

Lemma 4 *Executing the* COLORCAST *schedule ensures that all nodes obtain the broadcast message.*

Proof Let $G(V, E, \ell)$ be a powerline network, and $s \in V$ the broadcast source. Let R be the schedule produced by Algorithm 1. We first show by induction on the layers L_i

that all the nodes of T will get the message. The inductive step is that *if all the nodes of layer L_i received the broadcast message at round k, then all the nodes of layer L_{i+1} have the message after round $k + \xi_i$.* Base case $i = 0$: by definition, the source s has the message and $L_0 = \{s\}$. Now assume that at some round $k < \infty$, all nodes from layer L_i have the message. Let $u \in L_{i+1}$, u has at least one parent p in T on layer L_i. Observe that from round k to round $k + \xi_i$, all nodes of layer L_i will forward the message ($\forall v \in L_i, k < R[v] \leq k + \xi_i$). Thanks to Lemma 3 we know there cannot be any collisions, and since $(u, p) \in T \subset G_{com}^{\perp}$, u will receive the message. Thus at round $k + \xi_i$ all the nodes of layer L_{i+1} have the message, which proves the induction. Since all nodes of T eventually get the message and send it, and since T is a dominating set of $G_{com}^{\perp}$, the other nodes obtain the message as well. $\square$

Theorem 3 COLORCAST *solves the broadcast problem in time ξ, where $\xi = \sum_i \xi_i$ is the sum of the chromatic numbers of the layers. This is at most $n - 1$.*

Proof From Lemma 4, we know that COLORCAST eventually solves the broadcast problem. Observe that at the end of the execution (Line 12), the *round* variable is maximal. This variable is initialized in Line 6 and only incremented in Line 12. Thus, $round = \sum_i \xi_i = \xi \leq |D|$. Since D is a connected dominating set of a connected topology, it dominates at least one node: $|D| \leq n - 1$. This bound is tight: consider a line topology L_n with n nodes, where $G_{com}^{\perp} = L_n$ yields a line and $G_{com}^{\top} = K_n$ yields a clique. In such a scenario, all nodes but the last one will have to transmit. $\square$

Remark 1 Note that Theorem 3 gives an absolute and deterministic bound on the broadcast complexity: the time is simply bounded by the chromatic number of the interference graph. This is attractive and unlike many existing algorithms, especially for unknown topologies and algorithms based on selectors (cf Sect. 7), which have hidden constants in their asymptotic bounds. Also, note that in general, the chromatic number can be significantly lower than the network size ($\xi < n$).

Remark 2 Another interesting corollary from Theorem 3 is the fact that even with unknown transmission ranges, the network leaks certain information on its structure which can be exploited. Even if ρ^t can assume arbitrary *non-zero* values, there are communication topologies which cannot be generated from a given PLC infrastructure graph: for instance, in a line network L_n, a communication topology connecting nodes 1 and n must also connect nodes 1 and j for any $j \leq n - 1$. This is the reason why in our model, more efficient algorithms exist than in the unknown topology case.

6 Simulations

We evaluate our algorithm on the topology of a real urban electrical distribution grid of a town in Switzerland (population approx. 20k, area approx. 14 km^2), see Fig. 5. The electrical infrastructure consists of 93 nodes (primary substations and ring main units) connected by 107 edges. Typically, the distances between two neighboring ring main units are between 200m and 2000m in this area of Switzerland. Hence, we use the PLC infrastructure information provided by the utility as the graph $G = (V, E)$, and choose the lengths $d(e)$ uniformly at random between 200 and 2000. This corresponds to a PLC infrastructure without elements that disconnect PLC links (like open switches and transformers), and hence, we study a scenario with the maximum number of possible collisions. We generate 100 different graphs $G = (V, E, \ell)$ in this manner.

We investigate the effect of a multiplicative link quality function, i.e., $f(d(e), \rho^t(e)) = d(e) \cdot (1 + \rho^t(e))$ for a static scenario where the noise level may differ between edges, but it does not vary over time: $\rho^t(e) = \rho(e) \in [0, \rho_{max}]$ (the time complexity of the algorithm is independent of the current unknown conditions). We quantify the influence of varying ρ_{max} on the diameter and time complexity of broadcast algorithms with values derived from realistic scenarios. The powerline communication characteristics described in [31] and the model presented there lead to a bit error rate (BER) of close to 0 up to a distance of 2000m, after which there is a sharp increase. Based on this, we interpret $f(d, \rho)$ as a virtual distance in meters and assume that within a communication range of 2000m, nodes can communicate with each other and thus we do not consider longer edges.

To have a benchmark for the performance of COLORCAST, we also implemented the simple randomized DECAY algorithm (cf Algorithm 2) described in [4], which has a time complexity of $O((D + \log (n/\epsilon)) \cdot \log \Delta)$ with high probability, where D is the network diameter and ϵ is a parameter. While the maximal degree Δ can reach n, but can also be much lower, we run DECAY both with $\Delta = n$ and with $\Delta = \max \text{degree} (G^{com}_{\rho_{min}})$. In order to avoid penalizing DECAY for the fact that it is a randomized algorithm, we set ϵ to 1. This has the drawback that in our simulations, DECAY does not always reach all nodes. However, in our experiments this event occurred in less than 4 % of all cases, and we believe that the algorithm is well suited as a benchmark in this setting.

6.1 Influence of ρ_{max}

Figure 6 (left) plots the average duration of a broadcast on the topology of Fig. 5, for different ρ_{max} values. We compare COLORCAST with DECAY parametrized with two different estimates for Δ: maximum degree D of $G^{com}_{\rho_t}$ or number

---

Algorithm 2 DECAY(Δ, n, ϵ)

1: set k to $\log \Delta$; wait until receiving a message m
2: **for** $\log (n/\epsilon)$ times **do**
3:    wait until $time \mod k = 0$
4:    **repeat**
5:      transmit m, set $coin$ to 0 or 1 with equal probability
6:    **until** $coin = 0$ or sent k times

---

of nodes n. While the time complexity of COLORCAST does not vary with the actual ρ_t, the performance of DECAY is affected. Therefore, we run DECAY on the same topology with three different assignments and average them: (i) In the first scenario, ρ_t is set to 0 for all edges; (ii) in the second scenario, ρ is chosen uniformly at random between 0 and ρ_{max} for each edge; (iii) in the third scenario, edges are subject to ρ_{max}. Since COLORCAST only relies on ρ_{max} and not on the actual virtual distance of the edges, its performance is not affected by these different scenarios.

COLORCAST clearly outperforms DECAY on this medium-sized electrical distribution grid even in the scenario with most uncertainty and despite the randomized approach of DECAY which theoretically allows for asymptotically lower runtimes on average. When ρ_{max} is chosen uniformly at random, this increases the effective diameter from 12.37 to 17.38 and the average degree shrinks from 11.3 to 5.1 for maximal noise levels. However, as can be seen by the confidence interval for DECAY, even when the noise level is maximal on all edges, COLORCAST always completes broadcast faster.

6.2 Impact of scale

In order to study the impact of larger network sizes, we iteratively attach two copies of the basic network to each other. The two copies are connected by adding links between three
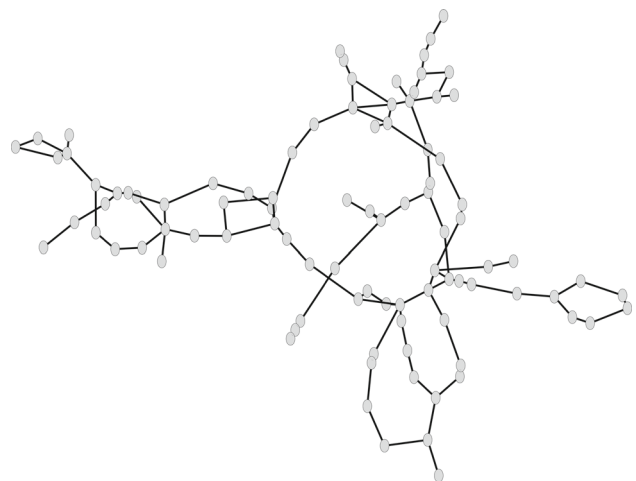

Fig. 5 Topology of a medium voltage distribution grid of a town in Switzerland
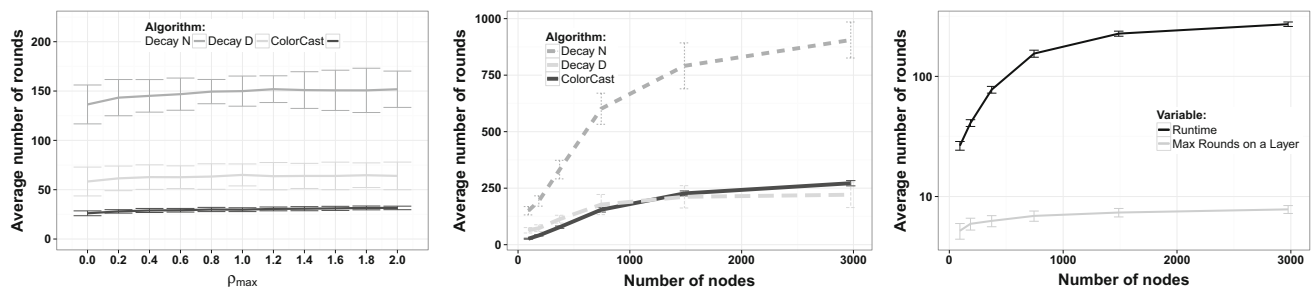
Fig. 6 Impact of ρ_{max} (*top*) and the network size (*center*) on the broadcast time for COLORCAST and DECAY. Number of colors per layer (*bottom*)

randomly chosen pairs of nodes of the two copies. This is reasonable, since larger distribution grids often have the same density as smaller distribution grids. Thus the average degree of the PLC infrastructure does not grow with the size of the network. In this manner, we construct 100 networks with $n = 93 \cdot 2^k$ nodes, for k between 0 and 5; thus, the largest networks contain 2976 nodes.

Figure 6 (*center*) studies how the broadcast time depends on the network size n. According to our construction of networks, the diameter grows roughly linearly with the size of the network, while the average degree grows from 11.3 to 16.6. With larger network sizes, the factors in the O-notation matter less and DECAY starts to exhibit better performance than COLORCAST for networks with more than around 1000 nodes. However, the difference in number of rounds is not very large, i.e., for networks with 2976 nodes, COLORCAST needs 263 rounds on average (std 14) while DECAY using the maximum degree for Δ finishes after 204 rounds (std 42). In general, DECAY is subject to a high variance (vertical bars represent the runtime's standard deviation over 100 runs), the different topologies influence COLORCAST' performance only slightly. This, in combination with the fact that DECAY cannot guarantee that the message reaches all nodes in all cases and fails to do so in up to 4% of all runs as well as the fact that Medium Voltage distribution grids are not arbitrarily large, indicates that COLORCAST is an attractive algorithm for PLC networks.

Figure 6 (*right*) sheds more light on these performance results: it shows the size distribution of the layers over the 100 different runs on each topology. A low number of colors facilitates a parallel traversal of the layers, and hence ensures a quick propagation of the broadcast message. Since the majority of layers are traversed in less than 10 rounds, regardless of the topology size, COLORCAST is efficient.

7 Related work

For an excellent overview of the broadcast problem in various radio network models (with an emphasis on time complexity), we refer the reader to the surveys [18] and [30]. Existing

broadcast models differ in the collision detection model, and the extent to wich the network topology is known.

Our model can be seen as an instance of a radio network model as well, in the sense that the transmissions of nodes are subject to topological constraints and collisions can happen when more than one node transmit at the same time. Thus, many existing broadcast algorithms can also be executed on our model. However, in contrast to most radio models, the underlying graph describes a PLC network with a different signal propagation compared to radio networks. However, we believe that the study of imperfect transmission ranges is of independent interest and the model can apply to other networks, such as radio networks as well. One interesting such radio model concerns Quasi-unit disk graphs (QUDGs), since they allow for a grey zone for links of lengths between d_{min} and 1 where communication may or may not be possible. Most UDG algorithms can be adapted for QUDG at the cost of an additional factor of $1/d_{min}^2$ [26]. This is acceptable for large d_{min}, but implies an increase of two orders of magnitude or more for $d_{min} < 0.1$.

To the best of our knowledge, the time complexity of the broadcast problem in networks with collisions under uncertain transmission ranges has not been considered so far. Several results from radio models with or without knowledge on the topology also apply on our networks.

7.1 Broadcast

A seminal work on the time-complexity of broadcast in multi-hop networks is by Bar-Yehuda et al. [4] who show an exponential gap between deterministic and randomized algorithms in the same collision model we adopt. Moreover they present a simple distributed randomized oblivious algorithm for unknown directed networks that broadcasts a message in time $O(D \log n + \log^2 n)$, a result which holds in our model too; however, as discussed, our setting generally allows for faster solutions. In the light of Kushilevitz et al.'s [27] randomized $\Omega(D \log(n/D))$ lower bound and Alon et al.'s [2] $\Omega(\log^2 n)$ lower bound for the broadcast problem in radio networks, the algorithm of [4] is almost optimal. For a slightly weaker collision model assuming that two or more transmis-

sions in the reception range always result in a collision that is indistinguishable from noise and silence the lower bound can be matched, as demonstrated in [11,22]. Another randomized algorithm reaches time $O(n \log n \log \log n)$ [12].

Many efficient broadcast algorithms for known topologies rely on the notion of a *wave front*: the algorithmic wit is focused on the frontier of nodes having the message (the potential senders) and their immediate vicinity (the potential receivers). Informally, in our model with unknown transmission ranges, it is *a priori* impossible to plan where the wavefront will be at a given time; moreover, since the communication graph is directed (and possibly only weakly connected), the *a posteriori* knowledge of which nodes received the message may remain local as well. When restricted to deterministic algorithms on unknown networks, lower bounds of $\Omega(n \log n / \log(n/D))$ and $\Omega(n \log D)$ are presented in [22] and [10] respectively. Deterministic algorithms for this scenario achieve broadcast in time $O(n \log n)$ [22], $O(n \log^2 D)$ [11] and $O(D\Delta \log^\alpha n)$ [10], where Δ is the maximum degree of the network and $\alpha \geq 2$ unless n is known ($\alpha = 2$) or n and Δ are known ($\alpha = 1$). If nodes know their neighbors, a broadcast time of $O(n)$ [3] is possible. In our model nodes know the possible communication topology of the whole network, however they do not know their effective neighbors. For scenarios where nodes are aware of the actual topology, a message can be broadcast in asymptotically optimal time $D + O(\log^2 n)$ [16] with a randomized algorithm or deterministically in time $D + O(\log^3 / \log \log n)$ [8] or $O(D + \log^2 n)$ [23].

An interesting related result to our article is due to Gasieniec et al. [15]. The authors attend to deterministic broadcasting in geometric/Euclidean radio networks whose nodes have complete knowledge of the network. Each node can transmit within some range r assigned to it; these ranges are nonuniform and they are drawn from the predefined interval $[r_{\min}, r_{\max}]$. A lower bound of $\delta + \Omega(\min\{\log(r_{\max}/r_{\min}), \log(n - \delta)\})$ is derived; on the positive side, an upper bound of $O(\delta \log^2(r_{\max}/r_{\min}))$ is shown. The main differences to our model are the fact that nodes know their range and the network growth is bounded.

While many broadcast results are limited to undirected networks, our model with unknown powers is inherently directed. In this regard, an interesting related result on directed communication networks has recently been shown by Kuhn et al. [25]. Finally, there also exists work on approximation algorithms for broadcast [13].

7.2 Relationship to abstract MAC

A recent line of research studies communication primitives such as broadcast and consensus on top of an Abstract MAC layer. Instead of dealing with MAC layer issues such as collisions and backoff in the design of new algorithms, basic guarantees on the delivery of messages by the MAC layer are assumed and algorithm complexities are analyzed based on these assumptions. E.g., given bounds for the maximum time F_{ack} of local broadcast (the MAC layer announces a message has been transmitted successfully) and the maximum time F_{prog} a device receives some message if neighbors are broadcasting, a simple broadcast algorithm that forwards a message whenever it receives it is analysed [17]. The authors show that this algorithm terminates fast, even when the availability of some links is unknown. More precisely, the following bounds hold: $O(DF_{prog} + kF_{ack})$ for known topologies [20], $O(DF_{prog} + kr F_{ack})$ for r-restricted uncertainty and $\Theta((D + k)F_{ack})$ for QUDG model [17].

The abstract MAC layer facilitates the design and analysis of network layer algorithms and is especially useful when working with devices where the MAC layer mechanisms can only be influenced by parameter tuning, but not by changing their algorithms. However, for domains with applications running on networks with very low bandwidth and static network infrastructure, it can be beneficial to address MAC layer primitives that are optimized for specific applications. Especially when determinism is key and hard deadlines are to be met, it is important that the MAC layer offers more than an indication that the medium seemed to be unoccupied before starting the transmission and that some near-by nodes might have received the message. Moreover it allows a more fine-grained analysis that gives bounds which are specific to the structure of a given network and do not depend on a potentially very high estimate value of F_{ack}. In other words, our work complements the line of research on the opportunities of Abstract MAC layer guarantees. More precisely, the algorithm COLORCAST it is a step in the opposite direction, i.e., it is shown how MAC layer transmission schedules with deterministic guarantees can be computed.

7.3 Relationship to dynamic graphs

Our work studies how to deal with uncertainty in terms of transmission ranges, and our model is related to the notion of temporal and dynamic graphs which have recently received much attention, both from an arbitrary graph as well as a geometric graph perspective [1,21,24]. Our graphs can be seen as a mixture of arbitrary and geometric graphs where propagation is dictated by physical lines. Also the dynamics of our communication graphs are constrained by the physical lines: the changes over time are due to varying noise levels on links.

8 Conclusion

We understand our work as a first step toward *network models* for powerline communication. As a case study, we consider

the broadcast problem in this model and give upper and lower time complexity bounds. We analyze the COLORCAST algorithm which not only achieves a good worst-case performance (strictly better than existing known and unknown topology approaches), but also performs well in our simulations. Interestingly, COLORCAST does not try to infer the transmission range of a given node. Consequently, the algorithm also works under changing transmission ranges and the assumption that transmission ranges are fixed is not necessary for our algorithm. Moreover, the algorithm automatically deals with directed links.

While we have derived lower bounds showing the inherent hardness of the broadcast problem under uncertain transmission ranges, the complexity results of the protocols presented in this article do not match these lower bounds. The obvious open research question regards the design of faster broadcast protocols in our model, or a proof that this is not possible.

References

1. Abshoff, S., Benter, M., Cord-Landwehr, A., Malatyali, M., auf der Heide, F. Meyer: Token dissemination in geometric dynamic networks. In: Flocchini, P., Gao, J., Kranakis, E., auf der Heide, F. Meyer (eds.) 9th International Symposium on Algorithms and Experiments for Sensor Systems, Wireless Networks and Distributed Robotics. Algorithms for Sensor Systems, Lecture Notes in Computer Science, vol. 8243. Springer, Berlin, Heidelberg, pp. 22–34. (2014)
2. Alon, N., Bar-Noy, A., Linial, N., Peleg, D.: A lower bound for radio broadcast. J. Comput. Syst. Sci. **43**(2), 290–298 (1991)
3. Awerbuch, B.: A new distributed depth-first-search algorithm. Inform. Process. Lett. **20**(3), 147–150 (1985)
4. Bar-Yehuda, R., Goldreich, O., Itai, A.: On the time-complexity of broadcast in multi-hop radio networks: an exponential gap between determinism and randomization. J. Comput. Syst. Sci. **45**, 104–126 (1992)
5. Ben Saad, L., Chauvenet, C., Tourancheau, B. et al.: Simulation of the RPL Routing Protocol for IPv6 Sensor Networks: two cases studies. In: SENSORCOMM, (2011)
6. Bumiller G.: Power-line Physical Layer Emulator for Protocol Development. In: ISPLC, (2004)
7. Burkhart, M., Von Rickenbach, P., Wattenhofer, R., Zollinger, A.: Does topology control reduce interference? In: Proceedings of the 5th ACM International Symposium on Mobile ad hoc Networking and Computing, pp. 9–19. ACM, (2004)
8. Cicalese, F., Manne, F., Xin, Q.: Faster centralized communication in radio networks. In: Algorithms and Computation. (2006)
9. Clark, B.N., Colbourn, C.J., Johnson, D.S.: Unit disk graphs. Discrete Math. **86**(1–3), 165–177 (1990)
10. Clementi, A., Monti, A., Silvestri, R.: Selective families, superimposed codes, and broadcasting on unknown radio networks. In: Proc. 12th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 709–718. Society for Industrial and Applied Mathematics, (2001)
11. Czumaj, A., Rytter, W.: Broadcasting algorithms in radio networks with unknown topology. J. Algorithms 492–501 (2003)
12. De Marco, G.: Distributed broadcast in unknown radio networks. In: Proceedings of 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 208–217 (2008)
13. Elkin, M., Kortsarz, G.: Polylogarithmic inapproximability of the radio broadcast problem. In: Proceedings of APPROX, vol. 3122. pp. 105–116 (2004)
14. Fussen, M., Wattenhofer, R., Zollinger, A.: Interference arises at the receiver. In: Wireless Networks, Communications and Mobile Computing, 2005 International Conference on, vol. 1, pp. 427–432. IEEE, (2005)
15. Gasieniec, L., Kowalski, D.R., Lingas, A., Wahlen, M.: Efficient broadcasting in known geometric radio networks with non-uniform ranges. In: Distributed Computing, pp. 274–288. Springer, (2008)
16. Gasieniec, L., Peleg, D., Xin, Q.: Faster communication in known topology radio networks. In: PODC (2005)
17. Ghaffari, M., Kantor, E., Lynch, N., Newport, C.: Multi-message broadcast with abstract mac layers and unreliable links. In: PODC (2014)
18. Ghaffari, M., Lynch, N., Newport, C.: The cost of radio network broadcast for different models of unreliable links. In: PODC (2013)
19. Gungor, V.C., Sahin, D., Kocak, T., Ergut, S., Buccella, C., Cecati, C., Hancke, G.P.: Smart grid technologies: communication technologies and standards. IEEE Trans. Ind. Inform. **7**(4), 529–539 (2011)
20. Khabbazian, M., Kuhn, F., Kowalski, D.R., Lynch, N.: Decomposing broadcast algorithms using abstract mac layers. In: Proceedings of the 6th International Workshop on Foundations of Mobile Computing, pp. 13–22. ACM (2010)
21. Kostakos, V.: Temporal graphs. Phys. A Stat. Mech. Appl. **388**(6), 1007–1023 (2009)
22. Kowalski, D., Pelc, A.: Broadcasting in undirected ad hoc radio networks. Distrib. Comput. **18**(1), 43–57 (2005)
23. Kowalski, D.R., Pelc, A.: Optimal deterministic broadcasting in known topology radio networks. Distrib. Comput. **19**(3), 185–195 (2007)
24. Kuhn, F., Lynch, N., Oshman, R.: Distributed computation in dynamic networks. In: Proc. ACM Symposium on Theory of Computing (STOC), pp. 513–522 (2010)
25. Kuhn, F., Oshman, R.: The complexity of data aggregation in directed networks. In: Proceedings of 25th International Conference on Distributed Computing (DISC), pp. 416–431 (2011)
26. Kuhn, F., Wattenhofer, R., Zollinger, A.: Ad-hoc networks beyond unit disk graphs. In: Proceedings of the 2003 joint workshop on Foundations of mobile computing, pp. 69–78. ACM (2003)
27. Kushilevitz, E., Mansour, Y.: An $\omega(d \log(n/d))$ lower bound for broadcast in radio networks. In: Proceedings of 12th Annual ACM Symposium on Principles of Distributed Computing (PODC), pp. 65–74 (1993)
28. Malacasa E., Morabito, G.: Characterization of PLC Communication Channel: a Networking Perspective. In: WSPLC (2009)
29. Patel, A., Aparicio, J., Tas, N., Loiacono, M., Rosca, J.: Assessing communications technology options for smart grid applications. In: IEEE SmartGridComm (2011)
30. Peleg, D.: Time-efficient broadcasting in radio networks: a review. In: ICDCT (2007)
31. Pignolet, Y.-A., Rinis, I., Dzung, D., Karaagac, A.: Multi-Interface extensions for PLC / wireless simulator. In: WSPLC (2012)
32. Versolatto, F., Tonello, A., Analysis of the PLC channel statistics using a bottom-up random simulator. In: ISPLC (2010)