Causality, Influence, and Computation in Possibly Disconnected Synchronous Dynamic Networks*

Othon Michail[1,2], Ioannis Chatzigiannakis[1,2], and Paul G. Spirakis[1,2]

[1] Computer Technology Institute & Press "Diophantus" (CTI), Patras, Greece
[2] Computer Engineering and Informatics Department (CEID), University of Patras
Email: {michailo, ichatz, spirakis}@cti.gr, Fax: +30 2610 960490

Abstract. In this work, we study the *propagation of influence and computation in dynamic networks that are possibly disconnected at every instant*. We focus on a *synchronous message passing* communication model with *broadcast* and bidirectional links. To allow for bounded end-to-end communication we propose a set of minimal *temporal connectivity conditions* that bound from the above the time it takes for information to make progress in the network. We show that even in dynamic networks that are disconnected at every instant information may spread as fast as in networks that are connected at every instant. Further, we investigate *termination criteria* when the nodes know some upper bound on each of the temporal connectivity conditions. We exploit our termination criteria to provide efficient protocols (optimal in some cases) that solve the fundamental *counting* and *all-to-all token dissemination* (or *gossip*) problems. Finally, we show that any protocol that is correct in instantaneous connectivity networks can be adapted to work in temporally connected networks.

Keywords: dynamic graph, mobile computing, worst-case dynamicity, adversarial schedule, temporal connectivity, counting, information dissemination

1 Introduction

Distributed computing systems are more and more becoming dynamic. The static and relatively stable models of computation can no longer represent the plethora of recently established and rapidly emerging information and communication technologies. In recent years, we have seen a tremendous increase in the number of new mobile computing devices. Most of these devices are equipped with some sort of communication, sensing, and mobility capabilities. Even the Internet has become mobile. The design is now focused on complex collections of

* This work has in part been supported by the EU (ESF) and Greek national funds through the Operational Programme "Education and Lifelong Learning" (EdLL), under the title "Foundations of Dynamic Distributed Computing Systems" (FOCUS).

heterogeneous devices that should be robust, adaptive, and self-organizing, possibly moving around and serving requests that vary with time. Delay-tolerant networks are highly-dynamic, infrastructure-less networks whose essential characteristic is a possible absence of end-to-end communication routes at any instant. Mobility may be *active*, when the devices control and plan their mobility pattern (e.g. mobile robots), or *passive*, in opportinistic-mobility networks, where mobility stems from the mobility of the carries of the devices (e.g. humans carrying cell phones) or a combination of both (e.g. the devices have partial control over the mobility pattern, like for example when GPS devices provide route instructions to their carriers). Thus, it can vary from being completely predictable to being completely unpredictable. Gossip-based communication mechanisms, e-mail exchanges, peer-to-peer networks, and many other contemporary communication networks all assume or induce some sort of high dynamicity.

The formal study of dynamic communication networks is hardly a new area of research. There is a huge amount of work in distributed computing that deals with causes of dynamicity such as failures and changes in the topology that are rather slow and usually eventually stabilize (like, for example, in self-stabilizing systems [Dol00]). However the low rate of topological changes that is usually assumed there is unsuitable for reasoning about truly dynamic networks. Even graph-theoretic techniques need to be revisited: the suitable graph model is now that of a *dynamic graph* (a.k.a. *temporal graph* or *time-varying graph*) (see e.g. [CFQS11]), in which each edge has an associated set of time-labels indicating availability times. Even fundamental properties of classical graphs do not carry over to their temporal counterparts. For example, Kempe, Kleinberg, and Kumar [KKK00] found out that there is no analogue of Menger's theorem (see e.g. [Bol98] for a definition) for arbitrary temporal networks, which additionally renders the computation of the number of node-disjoint s-t paths **NP**-complete. Even the standard network diameter metric is no more suitable and has to be replaced by a dynamic/temporal version. In a dynamic star graph in which all leaf-nodes but one go to the center one after the other in a modular way, any message from the node that enters last the center to the node that never enters the center needs $n-1$ steps to be delivered, where n is the size (number of nodes) of the network; that is the *dynamic diameter* is $n-1$ while, one the other hand, the classical diameter is just 2 [AKL08] (see also [KO11]).

2 Related Work

Distributed systems with worst-case dynamicity were first studied in [OW05]. Their outstanding novelty was to assume a communication network that may change arbitrarily from time to time subject to the condition that each instance of the network is connected. They studied asynchronous communication and considered nodes that can detect local neighborhood changes; these changes cannot happen faster than it takes for a message to transmit. They studied *flooding* (in which one node wants to disseminate one piece of information to all nodes) and *routing* (in which the information need only reach a particular destination node t) in this setting. They described a uniform protocol for flooding that terminates in $O(Tn^2)$ rounds using $O(\log n)$ bit storage and message overhead, where T is the maximum time it takes to transmit a message. They conjectured that without identifiers (IDs) flooding is impossible to solve within the above resources. Finally, a uniform routing algorithm was provided that delivers to the destination in $O(Tn)$ rounds using $O(\log n)$ bit storage and message overhead.

Computation under worst-case dynamicity was further and extensively studied in a series of works by Kuhn *et al.* in the synchronous case. In [KLO10], the network was assumed to be T*-interval connected* meaning that any time-window of length T has a static connected spanning subgraph (persisting throughout the window). Among others, *counting* (in which nodes must determine the size of the network) and *all-to-all token dissemination* (in which n different pieces of information, called tokens, are handed out to the n nodes of the network, each node being assigned one token, and all nodes must collect all n tokens) were solved in $O(n^2/T)$ rounds using $O(\log n)$ bits per message, almost-linear-time randomized approximate counting was established for $T = 1$, and two lower bounds on token dissemination were given. Several variants of *coordinated consensus* in 1-interval connected networks were studied in [KOM11]. [Hae11] is a recent work that presents information spreading algorithms in worst-case dynamic networks based on *network coding*. An *open* setting (modeled as high churn) in which nodes constantly join and leave has very recently been considered in [APRU12]. For an excellent introduction to distributed computation under worst-case dynamicity see [KO11]. Two very thorough surveys on dynamic networks are [Sch02,CFQS11].

Another notable model for dynamic distributed computing systems is the *population protocol* model [AAD$^+$06]. In that model, the computational agents are passively mobile, interact in ordered pairs, and the connectivity assumption is a *strong global fairness condition* according to which all events that may always occur, occur infinitely often. These assumptions give rise to some sort of structureless interacting automata model. The usually assumed *anonymity* and *uniformity* (i.e. n is not known) of protocols only allow for commutative computations that eventualy stabilize to a desired configuration. Most computability issues in this area have now been established. Constant-state nodes on a complete interaction network (and several variations) compute the *semilinear predicates* [AAER07]. Semilinearity persists up to $o(\log \log n)$ local space but not more than this [CMN$^+$11]. If constant-state nodes can additionally leave and update fixed-length pairwise marks then the computational power dramatically increases to the commutative subclass of $\mathbf{NSPACE}(n^2)$ [MCS11a]. For a very recent introductory text see [MCS11b].

3 Contribution

In this work, we study worst-case dynamic networks that are *free of any connectivity assumption about their instances.* Our dynamic network model is formally defined in Section 4.1. We only impose some *temporal connectivity* conditions on the adversary guaranteeing that *another causal influence occurs within every time-window of some given length*, meaning that, in that time, another node first hears of the state that some node u had at some time t (see Section 4.2 for a formal definition of *causal influence*). Note that our temporal connectivity conditions are minimal assumptions that allow for bounded end-to-end communication in any dynamic network including those that have disconnected instances. Based on this basic idea, we define several novel generic metrics for capturing the speed of information spreading in a dynamic network. In particular, we define the *outgoing influence time* (oit) as the maximal time until the state of a node *influences* the state of another node, the *incoming influence time* (iit) as the maximal time until the state of a node *is influenced by* the state of another node, and the *connectivity time* (ct) as the maximal time until the two parts of any cut of the network become connected. These metrics are defined in Section 5, where also several results that correlate these metrics are presented.

In Section 5.1, we present a simple but very fundamental dynamic graph based on alternating matchings that has oit 1 (equal to that of instantaneous connectivity networks) but at the same time is *disconnected in every instance.* In Section 6, we exhibit another dynamic graph additionally guaranteeing that edges take maximal time to reappear. That graph is based on a geometric edge-coloring method due to Soifer for coloring a complete graph of even order n with $n-1$ colors [Soi09]. Similar results have appeared before but to the best of our knowledge only in probabilistic settings [CMM$^+$08,BCF09].

In Section 7, we turn our attention to terminating computations and, in particular, we investigate termination criteria in networks in which an upper bound on the ct or the oit is known. By "termination criterion" we essentially mean any locally verifiable property that can be used to determine whether a node has heard from all other nodes. Note that we do not allow to the nodes any further knowledge on the network; for instance, nodes *do not* know the dynamic diameter of the network. In particular, in Section 7.1, we study the case in which an upper bound T on the ct is known and we present an optimal termination criterion that only needs time linear in the dynamic diameter and in T. Then, in Section 7.2, we study the case in which an upper bound K on the oit is known. We first present a termination criterion that needs time $O(K \cdot n^2)$. Additionally, we establish that even the optimal termination criterion for the ct case does not work in the oit case. These criteria share the fundamental property of hearing from the past. We then develop a new technique that gives an optimal termination criterion (time linear in the dynamic diameter and in K) by hearing from the future (by this we essentially mean that a node is interested for its outgoing influences instead for its incoming ones). Additionally, we exploit throughout the paper our termination criteria to provide protocols that solve the fundamental *counting* and *all-to-all token dissemination* (or *gossip*) problems; in the former nodes must determine the size of the network n and in the latter each node of the network is provided with a unique piece of information, called *token*, and all nodes must collect all n tokens. Then, we show that any protocol that is correct in 1-interval connected networks can be adapted to work in networks in which an upper bound on the oit, the iit, or the ct is known.

Finally, in Section 8, we conclude and discuss some interesting future research directions.

4 Preliminaries

4.1 The Dynamic Network Model

A *dynamic network* is modeled by a *dynamic graph* $G = (V, E)$, where V is a set of n nodes (or processors) and $E : \mathbb{N} \to \mathcal{P}(E')$ (wherever we use $\mathbb{N}$ we mean $\mathbb{N}_{\geq 1}$) is a function mapping a round number $r \in \mathbb{N}$ to a set $E(r)$ of bidirectional links drawn from $E' = \{\{u, v\} : u, v \in V\}$. [3] Intuitively, a dynamic graph G is an infinite sequence $G(1), G(2), \ldots$ of *instantaneous graphs*, whose edge sets are subsets of E' chosen by a *worst-case adversary.* A *static network* is just a special case of a dynamic network in which $E(i + 1) = E(i)$ for all $i \in \mathbb{N}$. The set V is assumed throughout this work to be *static*, that is it remains the same throughout the execution.

We assume that nodes in V have unique identities (ids) drawn from some namespace $\mathcal{U}$ (we assume that ids are represented using $O(\log n)$ bits) and that they do not know the topology or the size of the network, apart from some minimal necessary knowledge to allow for terminating computations (usually an upper bound on the time it takes for information to make some sort of progress). Any such assumed knowledge will be clearly stated. Moreover, nodes have unlimited local storage (though they usually use a reasonable portion of it).

Communication is *synchronous message passing* [Lyn96,AW04], meaning that it is executed in discrete steps controlled by a global clock that is available to the nodes and that nodes communicate by sending and receiving messages (usually of length that is some reasonable function of n, like e.g. $\log n$). We use the terms *round*, *time*, and *step* interchangeably to refer to the discrete steps of the system. Naturally, real rounds begin to count from 1 (e.g. "first round") and we reserve time 0 to refer to the initial state of the system. We assume that the message transmission model is *anonymous broadcast*, in which, in every round r, each node u generates a single message $m_u(r)$ to be delivered to all its current neighbors in $N_u(r) = \{v : \{u, v\} \in E(r)\}$ without knowing $N_u(r)$.

In every round, the adversary first chooses the edges for the round; for this choice it can see the internal states of the nodes at the beginning of the round. At the same time and independently of the adversary's choice of edges each node generates its message for the current round. Note that a node does not have any information about the internal state of its neighbors when generating its messages. In deterministic algorithms, nodes are only based on their current internal state to generate their messages and this implies that the adversary can infer the messages that will be generated in the current round before choosing

[3] By $\mathcal{P}(S)$ we denote the *powerset* of the set S, that is the set of all subsets of S.

the edges. In this work, we only consider deterministic algorithms. Each message is then delivered to the sender's neighbors, as chosen by the adversary; the nodes transition to new states, and the next round begins.

4.2 Spread of Influence in Dynamic Graphs (Causal Influence)

Probably the most important notion associated with a dynamic network/graph is the *causal influence*, which formalizes the notion of one node "influencing" another through a chain of messages originating at the former node and ending at the latter (possibly going through other nodes in between). We denote by (u,t) the state of node u at time t and usually call it the *t-state of u*. The pair (u,t) is also called a *time-node*. We use $(u,r) \rightsquigarrow (v,r')$ to denote the fact that node u's state in round r influences node v's state in round r'. Formally:

Definition 1 ([Lam78]). *Given a dynamic graph $G=(V,E)$ we define an order $\rightarrow \subseteq (V \times \mathbb{N}_{\geq 0})^2$, where $(u,r) \rightarrow (v,r+1)$ iff $u=v$ or $\{u,v\} \in E(r+1)$. The* causal order *$\rightsquigarrow \subseteq (V \times \mathbb{N}_{\geq 0})^2$ is defined to be the reflexive and transitive closure of $\rightarrow$.*

Obviously, for a dynamic distributed system to operate as a whole there must exist some upper bound on the time needed for information to spread through the network. This is the weakest possible guarantee since without it global computation is impossible. An abstract way to talk about information spreading is via the notion of the *dynamic diameter*. The *dynamic diameter* (also called *flooding time*, e.g., in [CMM$^+$08,BCF09]) of a dynamic graph, is an upper bound on the time required for each node to causally influence (or, equivalently, to be causally influenced by) every other node; formally, the dynamic diameter is the minimum $D \in \mathbb{N}$ s.t. for all times $t \geq 0$ and all $u,v \in V$ it holds that $(u,t) \rightsquigarrow (v,t+D)$. A small dynamic diameter allows for fast dissemination of information. In this work, we do not allow nodes to know the dynamic diameter of the network. We only allow some minimal knowledge (that will be explained in the sequel) based on which nodes may infer bounds on the dynamic diameter.

A class of dynamic graphs with small dynamic diameter is that of *T-interval connected* graphs. Formally, a dynamic graph $G=(V,E)$ is said to be *T-interval connected*, for $T \geq 1$, if, for all $r \in \mathbb{N}$, the static graph $G_{r,T} := (V, \bigcap_{i=r}^{r+T-1} E(r))$ is connected [KLO10]; that is, in every time-window of length T, a connected spanning subgraph is preserved.

Let us also define two very useful sets. We define by $\mathrm{past}_{(u,t')}(t) := \{v \in V : (v,t) \rightsquigarrow (u,t')\}$ [KOM11] the *past set of a time-node (u,t') from time t* and by $\mathrm{future}_{(u,t)}(t') := \{v \in V : (u,t) \rightsquigarrow (v,t')\}$ the *future set of a time-node (u,t) at time t'*, for times $0 \leq t \leq t'$. In words, $\mathrm{past}_{(u,t')}(t)$ is the set of nodes whose t-state (i.e. their state at time t) has causally influenced the t'-state of u and $\mathrm{future}_{(u,t)}(t')$ is the set of nodes whose t'-state has been causally influenced by the t-state of u. If $v \in \mathrm{future}_{(u,t)}(t')$ we say that at time t' node v *has heard of/from* the t-state of node u. If it happens that $t=0$ we say simply that v has heard of u. Note that $v \in \mathrm{past}_{(u,t')}(t)$ iff $u \in \mathrm{future}_{(v,t)}(t')$.

For a distributed system to be able to perform global computation, nodes need to be able to determine for all times $0 \le t \le t'$ whether $\mathrm{past}_{(u,t')}(t) = V$. If nodes know n, then a node can easily determine at time t' whether $\mathrm{past}_{(u,t')}(t) = V$ by counting all different t-states that it has heard of so far. If it has heard the t-states of all nodes then the equality is satisfied. If n is not known then various techniques may be applied (which is the subject of this work). By *termination criterion* we mean any locally verifiable property that can be used to determine whether $\mathrm{past}_{(u,t')}(t) = V$.

Remark 1. Note that any protocol that allows the nodes to determine whether $\mathrm{past}_{(u,t')}(t) = V$ can be used to solve the counting and all-to-all token dissemination problems. The reason is that if a node knows at round r that it has been causally influenced by the initial states of all other nodes then it can solve counting by writing $|\mathrm{past}_{(u,r)}(0)|$ on its output and all-to-all dissemination by writing $\mathrm{past}_{(u,r)}(0)$ (provided that all nodes send their initial states and all nodes constantly broadcast all initial states that they have heard of so far).

5 Our Metrics

As already stated, in this work we aim to deal with dynamic networks that are allowed to have disconnected instances. To this end, we define some novel generic metrics that are particularly suitable for capturing the speed of information propagation in such networks.

5.1 The Influence Time

Recall that the guarantee on propagation of information resulting from instantaneous connectivity ensures that any time-node (u,t) influences another node *in each step* (if an uninfluenced one exists). From this fact, we extract two novel generic influence metrics that capture the maximal time until another influence (outgoing or incoming) of a time-node occurs.

We now formalize our first influence metric. We define the *outgoing influence time* (oit) as the minimum $k \in \mathbb{N}$ s.t. for all $u \in V$ and all times $t, t' \ge 0$ s.t. $t' \ge t$ it holds that

$$|\mathrm{future}_{(u,t)}(t'+k)| \ge \min\{|\mathrm{future}_{(u,t)}(t')| + 1, n\}.$$

Intuitively, the oit is the maximal time until the t-state of a node influences the state of another node (if an uninfluenced one exists) and captures the speed of information spreading.

Our second metric, the *incoming influence time* (iit), is similarly defined as the minimum $k \in \mathbb{N}$ s.t. for all $u \in V$ and all times $t, t' \ge 0$ s.t. $t' \ge t$ it holds that $|\mathrm{past}_{(u,t'+k)}(t)| \ge \min\{|\mathrm{past}_{(u,t')}(t)| + 1, n\}$.

We can now say that the oit of a T-interval connected graph is 1 and that the iit can be up to $n-2$. However, is it necessary for a dynamic graph to be T-interval connected in order to achieve unit oit? First, let us make a simple but useful observation:

Proposition 1. *If a dynamic graph $G = (V, E)$ has* oit *(or* iit*) 1 then every instance has at least $\lceil n/2 \rceil$ edges.*

Proposition 1 is easily generalized as: if a dynamic graph $G = (V, E)$ has oit (or iit) k then for all times t it holds that $|\bigcup_{i=t}^{t+k-1} E(i)| \geq \lceil n/2 \rceil$. The reason is that now any node must have a neighbor in any k-window of the dynamic graph (and not necessarily in every round).

Now, inspired by Proposition 1, we define a minimal dynamic graph that at the same time satisfies oit 1 and always disconnected instances:

The Alternating Matchings Dynamic Graph. Take a ring of an even number of nodes $n = 2l$, partition the edges into 2 disjoint perfect matchings A and B (each consisting of l edges) and alternate round after round between the edge sets A and B.

Proposition 2. *The Alternating Matchings dynamic graph has* oit *1 and any node needs precisely $n/2$ rounds to influence all other nodes.*

In the alternating matchings construction any edge reappears every second step but not faster than this. We now formalize the notion of the *fastest edge reappearence* (fer) of a dynamic graph.

Definition 2. *The* fastest edge reappearence *(*fer*) of a dynamic graph $G = (V, E)$ is defined as the minimum $p \in \mathbb{N}$ s.t., $\exists e \in \{\{u, v\} : u, v \in V\}$ and $\exists t \in \mathbb{N}$, $e \in E(t) \cap E(t + p)$.*

Clearly, the fer of the alternating matchings dynamic graph described above is 2, because no edge ever reappears in 1 step and all and always reappear in 2 steps. In Section 6, by invoking a geometric edge-coloring method, we generalize this basic contruction to a more involved dynamic graph with oit 1, always disconnected instances, and fer equal to $n-1$. Note that the fer is always bounded from above by a function of n.

5.2 The Connectivity Time

We now propose another natural and practical metric for capturing the temporal connectivity of a possibly disconnected dynamic network that we call the *connectivity time* (ct).

Definition 3. *We define the* connectivity time *(*ct*) of a dynamic network $G = (V, E)$ as the minimum $k \in \mathbb{N}$ s.t. for all times $t \in \mathbb{N}$ the static graph $(V, \bigcup_{i=t}^{t+k-1} E(i))$ is connected.*

In words, the ct of a dynamic network is the maximal time of keeping the two parts of any cut of the network disconnected. That is to say, in every ct-window of the network an edge appears in every (V_1, V_2)-cut. Note that, in the extreme case in which the ct is 1, every instance of the dynamic graph is connected and we thus obtain a 1-interval connected graph. On the other hand, greater ct allows for different cuts to be connected at different times in the ct-round interval and the resulting dynamic graph can very well have disconnected instances. For an illustrating example, consider again the alternating matchings graph from Section 5.1. Draw a line that crosses two edges belonging to matching A partitioning the ring into two parts. Clearly, these two parts communicate every second round (as they only communicate when matching A becomes available), thus the ct is 2 and every instance is disconnected. We now provide a result associating the ct of a dynamic graph with its oit.

Proposition 3. *(i) $\mathsf{oit} \leq \mathsf{ct}$ but (ii) there is a dynamic graph with oit 1 and $\mathsf{ct} = \Omega(n)$.*

Proof. (i) We show that for all $u \in V$ and all times $t, t' \in \mathbb{N}$ s.t. $t' \geq t$ it holds that $|\text{future}_{(u,t)}(t'+\mathsf{ct})| \geq \min\{|\text{future}_{(u,t)}(t')|+1, n\}$. Assume $V \backslash \text{future}_{(u,t)}(t') \neq \emptyset$ (as the other case is trivial). In at most ct rounds at least one edge joins $\text{future}_{(u,t)}(t')$ to $V \backslash \text{future}_{(u,t)}(t')$. Thus, in at most ct rounds $\text{future}_{(u,t)}(t')$ increases by at least one.

(ii) Recall the alternating matchings on a ring dynamic graph from Section 5.1. Now take any set V of a number of nodes that is a multiple of 4 (this is just for simplicity and is not necessary) and partition it into two sets V_1, V_2 s.t. $|V_1| = |V_2| = n/2$. If each part is an alternating matchings graph for $|V_1|/2$ rounds then every u say in V_1 influences 2 new nodes in each round and similarly for V_2. Clearly we can keep V_1 disconnected from V_2 for $n/4$ rounds without violating $\mathsf{oit} = 1$. □

6 Fast Propagation of Information Under Continuous Disconnectivity

In Section 5.1, we presented a simple example of an always-disconnected dynamic graph, namely, the alternating matchings dynamic graph, with optimal oit (i.e. unit oit). Note that the alternating matchings dynamic graph may be conceived as simple as it has small fer (equal to 2). We pose now, and answer to the positive, an interesting question: Is there an always-disconnected dynamic graph with unit oit and fer as big as $n-1$?

Let us define a very useful dynamic graph coming from the area of edge-coloring.

Definition 4. *We define the following dynamic graph S based on an edge-coloring method due to Soifer [Soi09]: $V(S) = \{u_1, u_2, \ldots, u_n\}$ where $n = 2l$, $l \geq 2$. Place u_n on the center and $u_1, \ldots, n_{n-1}$ on the vertices of a $(n-1)$-sided polygon. For each time $t \geq 1$ make available only the edges $\{u_n, u_{m(0)}\}$ for $m(j) := (t-1+j \bmod n-1)+1$ and $\{u_{m(-i)}, u_{m(i)}\}$ for $i = 1, \ldots, n/2-1$; that is make available one edge joining the center to a polygon-vertex and all edges perpendicular to it.*

Theorem 1. *For all $n = 2l$, $l \geq 2$, there is a dynamic graph of order n, with* oit *equal to 1,* fer *equal to $n-1$, and in which every instance is a perfect matching. This is Soifer's graph.*

Note that Theorem 1 is optimal w.r.t. fer as *it is impossible to achieve at the same time unit* oit *and* fer *strictly greater than $n-1$*. To see this, notice that if no edge is allowed to reappear in less than n steps then any node must have no neighbors once every n steps.

7 Termination and Computation

We now turn our attention to termination criteria that we exploit to solve the fundamental counting and all-to-all token dissemination problems. Keep in mind that nodes have no *a priori* knowledge of the size of the network.

7.1 Nodes Know an Upper Bound on the ct: An Optimal Termination Criterion

We here assume that all nodes know some upper bound T on the ct. We will give an optimal condition that allows a node to determine whether it has heard from all nodes in the graph. This condition results in an algorithm for counting and all-to-all token dissemination which is optimal, requiring $O(D+T)$ rounds in any dynamic network with dynamic diameter D. The core idea is to have each node keep track of its past sets from time 0 and from time T and terminate as long as these two sets become equal. This technique is inspired from [KOM11], where a comparison between the past sets from time 0 and time 1 was used to obtain an optimal termination criterion in 1-interval connected networks.

Theorem 2 (Repeated Past). *Node u knows at time t that* $\mathrm{past}_{(u,t)}(0) = V$ *iff* $\mathrm{past}_{(u,t)}(0) = \mathrm{past}_{(u,t)}(T)$.

Proof. If $\mathrm{past}_{(u,t)}(0) = \mathrm{past}_{(u,t)}(T)$ then we have that $\mathrm{past}_{(u,t)}(T) = V$. The reason is that $|\mathrm{past}_{(u,t)}(0)| \geq \min\{|\mathrm{past}_{(u,t)}(T)| + 1, n\}$. To see this, assume that $V \backslash \mathrm{past}_{(u,t)}(T) \neq \emptyset$. At most by round T there is some edge joining some $w \in V \backslash \mathrm{past}_{(u,t)}(T)$ to some $v \in \mathrm{past}_{(u,t)}(T)$. Thus, $(w, 0) \rightsquigarrow (v, T) \rightsquigarrow (u, t) \Rightarrow w \in \mathrm{past}_{(u,t)}(0)$. In words, all nodes in $\mathrm{past}_{(u,t)}(T)$ belong to $\mathrm{past}_{(u,t)}(0)$ and at least one node not in $\mathrm{past}_{(u,t)}(T)$ (if one exists) must belong to $\mathrm{past}_{(u,t)}(0)$.

For the other direction, assume that there exists $v \in \mathrm{past}_{(u,t)}(0) \backslash \mathrm{past}_{(u,t)}(T)$. This does not imply that $\mathrm{past}_{(u,t)}(0) \neq V$ but it does imply that even if $\mathrm{past}_{(u,t)}(0) = V$ node u cannot know it has heard from everyone. Note that u heard from v at some time $T' < T$ but has not heard from v since then. It can be the case that arbitrarily many nodes were connected to no node until time $T-1$ and from time T onwards were connected only to node v (v in some sense conceals these nodes from u). As u has not heard from the T-state of v it can be the case that it has not heard at all from arbitrarily many nodes, thus it cannot decide on the count. $\square$

We now give a time-optimal $O(D+T)$-round algorithm for counting and all-to-all token dissemination that is based on Theorem 2.

Protocol *A*. All nodes constantly forward all 0-states and T-states of nodes that they have heard of so far (the ids of the nodes accompanied with 0 and T timestamps, respectively) and a node halts as soon as $\text{past}_{(u,t)}(0) = \text{past}_{(u,t)}(T)$ and outputs $|\text{past}_{(u,t)}(0)|$ for counting or $\text{past}_{(u,t)}(0)$ for all-to-all dissemination.

7.2 Known Upper Bound on the oit: Another Optimal Termination Criterion

Now we assume that all nodes know some upper bound K on the oit.

7.2.1 Inefficiency of Hearing the Past We begin by proving that if a node u has at some point heard of l nodes, then u hears of another node in $O(Kl^2)$ rounds (if an unknown one exists).

Theorem 3. *In any given dynamic graph with oit upper bounded by K, take a node u and a time t and denote $|\text{past}_{(u,t)}(0)|$ by l. It holds that $|\{v : (v,0) \rightsquigarrow (u, t+Kl(l+1)/2)\}| \geq \min\{l+1, n\}$.*

Proof. Consider a node u and a time t and define $A_u(t) := \text{past}_{(u,t)}(0)$ (we only prove it for the initial states of nodes but easily generalizes to any time), $I_u(t') := \{v \in A_u(t) : A_v(t') \backslash A_u(t) \neq \emptyset\}$, $t' \geq t$, that is $I_u(t')$ contains all nodes in $A_u(t)$ whose t'-states have been influence by nodes not in $A_u(t)$ (these nodes know new info for u), $B_u(t') := A_u(t) \backslash I_u(t')$, that is all nodes in $A_u(t)$ that do not know new info, and $l := |A_u(t)|$. The only interesting case is for $V \backslash A_u(t) \neq \emptyset$. Since the oit is at most K we have that at most by round $t+Kl$, (u,t) influences some node in $V \backslash B_u(t)$ say via some $u_2 \in B_u(t)$. By that time, u_2 leaves B_u. Next consider $(u, t+Kl+1)$. In $K(l-1)$ steps it must influence some node in $V \backslash B_u$ since now u_2 is not in B_u. Thus, at most by round $t+Kl+K(l-1)$ another node, say e.g. u_3, leaves B_u. In general, it holds that $B_u(t'+K|B_u(t')|) \leq \max\{|B_u(t')|-1, 0\}$. It is not hard to see that at most by round $j = t + K(\sum_{1 \leq i \leq l} i)$, B_u becomes empty, which by definition implies that u has been influenced by the initial state of a new node. In summary, u is influenced by another initial state in at most $K(\sum_{1 \leq i \leq l} i) = kl(l+1)/2$ steps. □

The good thing about the upper bound of Theorem 3 is that it associates the time for a new incoming influence to arrive at a node only with an upper bound on the oit, which is known, and the number of existing incoming influences which is also known, and thus the bound is locally computable at any time. So, there is a straightforward translation of this bound to a termination criterion and further to an $O(Kn^2)$ algorithm for counting and all-to-all dissemination.

Note that the upper bound of Theorem 3 is loose. The reason is that if a dynamic graph has oit upper bounded by K then in $O(Kn)$ rounds all nodes have causally influenced all other nodes and clearly the iit can be at most $O(Kn)$. In fact, it is not hard to construct a dynamic graph that achieves this worst possible gap between the iit and the oit. On the other hand, the bound of Theorem 3 is optimal in the following sense: a node cannot obtain a better upper bound based solely on K and l.

We now show that even the criterion of Theorem 2, that is optimal if an upper bound on the ct is known, does not work in dynamic graphs with known an upper bound K on the oit. In particular, we show that for any time $t' \in \mathbb{N}$ which can only depend on K (otherwise it is fixed) there is a dynamic graph with oit upper bounded by K, a node u, and a time $t \in \mathbb{N}$ s.t. $\mathrm{past}_{(u,t)}(0) = \mathrm{past}_{(u,t)}(t')$ while $\mathrm{past}_{(u,t)}(0) \neq V$. In words, for any such t' it can be the case that while u has not been yet causally influenced by all initial states its past set from time 0 may become equal to its past set from time t', which violates the termination criterion of Theorem 2.

Theorem 4. *For any time t' (which can only depend on the upper bound K on the oit) there is a dynamic graph with oit upper bounded by K, a node u, and a time $t \in \mathbb{N}$ s.t. $\mathrm{past}_{(u,t)}(0) = \mathrm{past}_{(u,t)}(t')$ while $\mathrm{past}_{(u,t)}(0) \neq V$.*

Proof. Let n be sufficiently large, that is $n \gg t'$, and for simplicity assume that n is a multiple of 4. As in Proposition 3-ii, we can keep two parts V_1, V_2 of the network, of size $n/2$ each, disconnected up to some time $\Omega(n)$. Let $u \in V_1$. At time $t'+1$ the adversary directly connects some node $v \in V_1$ to all $w \in V_1$. Now v knows the t'-states (and of course also the 0-states) of all nodes in V_1. Then at time $t'+2$ the adversary connects v only to u and to some node in V_2. Clearly, at time $t'+2$, u learns the t'-states of all nodes in V_1 (v inclusive) and it holds that $\mathrm{past}_{(u,t'+2)}(0) = \mathrm{past}_{(u,t'+2)}(t')$. Additionally, $|\mathrm{past}_{(u,t)}(0)| = n/2 \Rightarrow \mathrm{past}_{(u,t)}(0) \neq V$. □

7.2.2 Hearing the Future We now present an optimal protocol for counting and all-to-all dissemination in dynamic networks with known an upper bound K on the oit, that is based on the following termination criterion. By definition of oit, if $\mathrm{future}_{(u,0)}(t) = \mathrm{future}_{(u,0)}(t+K)$ then $\mathrm{future}_{(u,0)}(t) = V$. The reason is that if there exist uninfluenced nodes, then at least one such node must be influenced in at most K rounds, otherwise no such node exists and $(u,0)$ must have already influenced all nodes. So, a fundamental goal is to allow a node to know its future set. Note that this criterion has a very basic difference from all termination criteria that have so far been applied to worst-case dynamic networks: instead of keeping track of its past set(s) and waiting for new incoming influences a node now directly keeps track of its future set and is informed by other nodes of its progress. We assume, for simplicity, a unique leader l in the initial configuration of the system (we later drop this unnecessary assumption).

Protocol *Hear_from_known*. We denote by r the current round. Each node u keeps a list $Infl_u$ in which it keeps track of all nodes that first heard of $(l,0)$ (the initial state of the leader) by u (u was between those nodes that first acquained $(l,0)$ to nodes in $Infl_u$), a set A_u in which it keeps track of the $Infl_v$ sets that it is aware of initially set to $(u, Infl_u, 1)$, and a variable *timestamp* initially set to 1. Each node u broadcast in every round (u, A_u) and if it has heard of $(l, 0)$ also broadcasts $(l, 0)$. Upon reception of an id w that is not accompanied with $(l, 0)$, a node u that has already heard of $(l, 0)$ adds (w, r) to $Infl_u$ to recall that at round r it notified w of $(l, 0)$ (note that it is possible that other nodes also notify w of $(l, 0)$ at the same time without u being aware of them; all these nodes will write (w, r) in their lists). If it ever holds at a node u that $r > \max_{(v \neq u, r') \in Infl_u}\{r'\} + K$ then u adds (u, r) in $Infl_u$ (replacing any existing $(u, t) \in Infl_u$) to denote the fact that r is the maximum known time until which u has performed no further propagations of $(l, 0)$. If at some round r a node u modifies its $Infl_u$ set, it sets $timestamp \leftarrow r$. In every round, a node u updates A_u by storing in it the most recent $(v, Infl_v, timestamp)$ triple of each node v that it has heard of so far (its own $(u, Infl_u, timestamp)$ inclusive), where the "most recent" triple of a node v is the one with the greatest *timestamp* between those whose first component is v. Moreover, u clears multiple (w, r) records from the $Infl_v$ lists of A_u. In particular, it keeps (w, r) only in the $Infl_v$ list of the node v with the smallest id between those that share (w, r). Similarly, the leader collects all $(v, Infl_v, timestamp)$ triples in its own A_l set. Let *tmax* denote the maximum timestamp appearing in A_l, that is the maximum time for which the leader knows that some node was influenced by $(l, 0)$ at that time. Moreover denote by I the set of nodes that the leader knows to have been influenced by $(l, 0)$ [4]. If at some round r it holds at the leader that for all $u \in I$ there is a $(u, Infl_u, timestamp) \in A_l$ s.t. $timestamp \geq tmax + K$ and $\max_{(w \neq u, r') \in Infl_u}\{r'\} \leq tmax$ then the leader notifies the other nodes about termination for $K \cdot |I|$ rounds and then outputs $|I|$ or I depending on whether counting or all-to-all dissemination needs to be solved and halts.

The above protocol can be easily made to work without the assumption of a unique leader. The idea is to have all nodes begin as leaders and make all nodes prefer the leader with the smallest id that they have heard of so far. In particular, we can have each node keep an $Infl_{(u,v)}$ only for the smallest v that it has heard of so far. Clearly, in $O(D)$ rounds all nodes will have sticked to the node with the smallest id in the network.

Theorem 5. *Protocol* Hear_from_known *solves counting and all-to-all dissemination in $O(D + K)$ rounds by using messages of size $O(n \log Kn)$, in any dynamic network with dynamic diameter D, and with* oit *upper bounded by some K known to the nodes.*

We defer for the full paper a protocol (inspired from a technique from [MCS12]) that solves counting and all-to-all dissemination in $O(Dn^2 + K)$ rounds by using messages of size $O(\log D + \log n)$, in any dynamic network with dynamic diameter D, and with oit upper bounded by some K known to the nodes.

[4] Note that I can be extracted from A_l by $I = \{v \in V : \exists u \in V, \exists timestamp, r \in \mathbb{N}$ s.t. $(u, Infl_u, timestamp) \in A_l$ and $(v, r) \in Infl_u\}$.

Finally, it is not hard to prove that protocols that are correct in 1-interval connected networks carry over to networks in which an upper bound on the oit, iit, or ct is known, with only a small delay being introduced in the process.

8 Conclusions

We studied for the first time worst-case dynamic networks that are free of any connectivity assumption about their instances. To enable a quantitative study we proposed some novel generic metrics that capture the speed of information propagation in a dynamic network. We proved that fast dissemination and computation are possible even under continuous disconnectivity. In particular, we presented optimal termination conditions and protocols based on them for the fundamental counting and all-to-all token dissemination problems.

There are many open problems and promising research directions related to this work. An asynchronous communication model in which nodes can broadcast when there are new neighbors would be a very natural extension of the synchronous model that we studied in this work. Note that in our work (and all previous work on the subject) information dissemination is only guaranteed under continuous broadcasting. How can the number of redundant transmissions be reduced in order to improve communication efficiency? Is there a way to exploit visibility to this end? Does predictability help? Finally, randomization will be certainly valuable in constructing fast and symmetry-free protocols. We strongly believe that these and other known open questions and research directions will motivate the further growth of this emerging field.

References

[AAD+06] D. Angluin, J. Aspnes, Z. Diamadi, M. J. Fischer, and R. Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed Computing*, pages 235–253, March 2006.

[AAER07] D. Angluin, J. Aspnes, D. Eisenstat, and E. Ruppert. The computational power of population protocols. *Distributed Computing*, 20[4]:279–304, November 2007.

[AKL08] C. Avin, M. Koucký, and Z. Lotker. How to explore a fast-changing world (cover time of a simple random walk on evolving graphs). In *Proceedings of the 35th international colloquium on Automata, Languages and Programming, Part I*, pages 121–132, Berlin, Heidelberg, 2008. Springer-Verlag.

[APRU12] J. Augustine, G. Pandurangan, P. Robinson, and E. Upfal. Towards robust and efficient computation in dynamic peer-to-peer networks. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 551–569. SIAM, 2012.

[AW04] H. Attiya and J. Welch. *Distributed computing: fundamentals, simulations, and advanced topics*, volume 19. Wiley-Interscience, 2004.

[BCF09] H. Baumann, P. Crescenzi, and P. Fraigniaud. Parsimonious flooding in dynamic graphs. In *Proceedings of the 28th ACM symposium on Principles of distributed computing*, PODC '09, pages 260–269. ACM, 2009.

[Bol98] B. Bollobás. *Modern Graph Theory*. Springer, corrected edition, July 1998.

[CFQS11] A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro. Time-varying graphs and dynamic networks. *Ad-hoc, Mobile, and Wireless Networks*, pages 346–359, 2011.

[CMM+08] A. E. Clementi, C. Macci, A. Monti, F. Pasquale, and R. Silvestri. Flooding time in edge-markovian dynamic graphs. In *Proceedings of the twenty-seventh ACM symposium on Principles of distributed computing*, PODC '08, pages 213–222, New York, NY, USA, 2008. ACM.

[CMN+11] I. Chatzigiannakis, O. Michail, S. Nikolaou, A. Pavlogiannis, and P. G. Spirakis. Passively mobile communicating machines that use restricted space. *Theor. Comput. Sci.*, 412[46]:6469–6483, October 2011.

[Dol00] S. Dolev. *Self-stabilization*. MIT Press, Cambridge, MA, USA, 2000.

[Hae11] B. Haeupler. Analyzing network coding gossip made easy. In *Proceedings of the 43rd annual ACM symposium on Theory of computing*, pages 293–302. ACM, 2011.

[KKK00] D. Kempe, J. Kleinberg, and A. Kumar. Connectivity and inference problems for temporal networks. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 504–513, 2000.

[KLO10] F. Kuhn, N. Lynch, and R. Oshman. Distributed computation in dynamic networks. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 513–522, New York, NY, USA, 2010. ACM.

[KO11] F. Kuhn and R. Oshman. Dynamic networks: models and algorithms. *SIGACT News*, 42:82–96, March 2011. Distributed Computing Column, Editor: Idit Keidar.

[KOM11] F. Kuhn, R. Oshman, and Y. Moses. Coordinated consensus in dynamic networks. In *Proceedings of the 30th annual ACM SIGACT-SIGOPS symposium on Principles of distributed computing*, pages 1–10, 2011.

[Lam78] L. Lamport. Time, clocks, and the ordering of events in a distributed system. *Commun. ACM*, 21[7]:558–565, 1978.

[Lyn96] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann; 1st edition, 1996.

[MCS11a] O. Michail, I. Chatzigiannakis, and P. G. Spirakis. Mediated population protocols. *Theor. Comput. Sci.*, 412[22]:2434–2450, May 2011.

[MCS11b] O. Michail, I. Chatzigiannakis, and P. G. Spirakis. *New Models for Population Protocols*. N. A. Lynch (Ed), Synthesis Lectures on Distributed Computing Theory. Morgan & Claypool, 2011.

[MCS12] O. Michail, I. Chatzigiannakis, and P. G. Spirakis. Brief Announcement: Naming and counting in anonymous unknown dynamic networks. To appear in the *26th International Symposium on Distributed Computing*, DISC '12, 2012.

[OW05] R. O'Dell and R. Wattenhofer. Information dissemination in highly dynamic graphs. In *Proceedings of the 2005 joint workshop on Foundations of mobile computing*, DIALM-POMC '05, pages 104–110, 2005.

[Sch02] C. Scheideler. Models and techniques for communication in dynamic networks. In *Proceedings of the 19th Annual Symposium on Theoretical Aspects of Computer Science*, STACS '02, pages 27–49, 2002.

[Soi09] A. Soifer. *The Mathematical Coloring Book: Mathematics of Coloring and the Colorful Life of its Creators*. Springer; 1st edition, 2009.