Proceedings of the

First Workshop on Dynamic Graphs in Distributed Computing

(DGDC'16)

DISC 2016

Paris, France

Mobile ad-hoc networks, sensor networks, or peer-to-peer systems are some examples of distributed systems that share a very challenging property: they are dynamic in the sense that their underlying topology changes unpredictably with time. Due to the great development of such systems in many fields, distributed computing practitioners must take dynamicity into account. The challenge is that it is not sufficient to adapt existing solutions from static systems due to the intrinsic nature of these systems, in which dynamics is not an exception. Even basic assumptions like the availability of paths may sometimes become irrelevant in highly-dynamic networks, leading to reconsider the meaning of basic tasks like election, broadcast, or routing.

From an algorithmic point of view, the first step is to correctly model the dynamicity of the underlying topology. In some cases, it can be reformulated in terms of classical properties (e.g. scheduling). However, numerous works in the field of distributed computing (and beyond) recently considered graph-theoretical approaches to capture various definitions of dynamics and their impact on distributed computing.

The Workshop on Dynamic Graphs in Distributed Computing (DGDC) precisely focuses on new graph theoretical approaches to model dynamics and their implications in distributed computing. Topics of interest include, but are not limited to:

- New models or improvement of existing models of dynamic graphs
- Computability and/or complexity in distributed computing based on dynamic graph assumptions
- Relations among dynamic graph properties and concepts
- Property testing in dynamic graphs
- Modeling and/or verification of dynamic systems
- Fault-tolerance in dynamic systems

The first edition of DGDC was held in conjunction with the 30th International Symposium on Distributed Computing (DISC'16) in Paris, France on September 30, 2016. The program consisted of 8 talks, all but the first one being invited, whose summaries are recorded in these proceedings.

We want to warmly thank each of the speakers that accepted our invitation to present their work in DGDC'16. We are convinced that these talks make for an exciting program.

We also want to thank the 30th International Symposium on Distributed Computing (DISC'16) and its organizing committee for suggesting that we organize such a workshop.

Arnaud Casteigts, Université de Bordeaux, France
Swan Dubois, UPMC Sorbonne Universités & Inria, France

A Set Theoretic Approach to Dynamic Graphs

Arnaud Casteigts

Université de Bordeaux, France

Highly-dynamic networks seem chaotic at first. However, dealing with them does not mean working without assumption. The network dynamics, whether it be predictable or not, may still obey underlying rules that can be exploited. For instance, connectivity may not always be available, but there may exist paths over time and space (*a.k.a.* temporal paths or *journeys*) that enables a form of temporal connectivity. Other assumptions can be formulated which relate to properties over time. For instance, will all pairs of nodes interact at least once over the execution? Will some edges reappear recurrently (infinitely often)? Will this reappearance be bounded in time, or even periodic? Will the topology remain connected in the usual sense despite being dynamics? If not, will there recurrently exist journeys between every pair of nodes? Etc. Each property defines a particular class (i.e. a *set*) of dynamic graphs.

In this talk, I will present a dozen such classes and some links between them. Then I will advocate their use in the study of (dynamic adaptations of) classical problems in distributed computing. In particular, these classes prove useful in comparing the required assumptions that algorithms make on dynamics. Most elements of this talk are from the references below.

References

1. Bui-Xuan, B., Ferreira, A., Jarry, A.: Computing shortest, fastest, and foremost journeys in dynamic networks. Int. J. Found. Comput. Sci. **14**(2) (2003) 267–285
2. Casteigts, A., Flocchini, P., Mans, B., Santoro, N.: Shortest, fastest, and foremost broadcast in dynamic networks. Int. J. Found. Comput. Sci. **26**(4) (2015) 499–522
3. Casteigts, A., Flocchini, P., Quattrociocchi, W., Santoro, N.: Time-varying graphs and dynamic networks. IJPEDS **27**(5) (2012) 387–408
4. Casteigts, A., Chaumette, S., Ferreira, A.: Characterizing topological assumptions of distributed algorithms in dynamic networks. In: 16th International Colloquium onStructural Information and Communication Complexity (SIROCCO 2009). (2009) 126–140
5. Dubois, S., Kaaouachi, M., Petit, F.: Enabling minimal dominating set in highly dynamic distributed systems. In: 17th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2015). (2015) 51–66

Expressivity of TVGs (or Why Dynamic Graphs should be modelled with TAs?)

Emmanuel Godard

Université Aix-Marseille, France

Time-varying graphs model in a natural way infrastructure-less highly dynamic systems, such as wireless ad-hoc mobile networks, robotic swarms, vehicular networks, etc. In these systems, a path from a node to another might still exist over time, rendering computing possible, even though at no time the path exists in its entirety. Some of these systems allow waiting (i.e., provide the nodes with store-carry-forward-like mechanisms such as local buffering) while others do not.

In this talk, we focus on the structure of the time-varying graphs modelling these highly dynamical environments. We examine the complexity of these graphs, with respect to waiting, in terms of their expressivity; that is in terms of the language generated by the feasible journeys (i.e., the paths over time). We prove that the set of languages $\mathcal{L}_{nowait}$ when no waiting is allowed contains all computable languages. On the other end, using algebraic properties of quasi-orders, we prove that $\mathcal{L}_{wait}$ is just the family of regular languages, even if the presence of edges is controlled by some arbitrary function of the time. In other words, we prove that, when waiting is allowed, the power of the accepting automaton drops drastically from being as powerful as a Turing machine, to becoming that of a Finite-State machine. This large gap provides a measure of the impact of waiting. We also study bounded waiting; that is when waiting is allowed at a node for at most d time units. We prove that $\mathcal{L}_{wait[d]} = \mathcal{L}_{nowait}$; that is, the complexity of the accepting automaton decreases only if waiting is unbounded.

We will also discuss what these results entails for the relationship between (general) TVGs and Timed Automaton.

An Overview of Recent Progress in Temporal Graphs: An Algorithmic Perspective*

Othon Michail

Department of Computer Science, University of Liverpool, Liverpool, UK
Computer Technology Institute and Press Diophantus (CTI), Patras, Greece

A *temporal graph* is, informally speaking, a graph that changes with time. When time is discrete and only the relationships between the participating entities may change and not the entities themselves, a temporal graph may be viewed as a sequence $G_1, G_2 \ldots, G_l$ of static graphs over the same (static) set of nodes V. Though static graphs have been extensively studied, for their temporal generalization we are still far from having a concrete set of structural and algorithmic principles. Recent research shows that many graph properties and problems become radically different and usually substantially more difficult when an extra time dimension is added to them. Moreover, there is already a rich and rapidly growing set of modern systems and applications that can be naturally modeled and studied via temporal graphs. This, further motivates the need for the development of a temporal extension of graph theory. In this talk, we will survey recent results on temporal graphs and temporal graph problems that have appeared in the Computer Science community.

References

1. Michail, O.: An introduction to temporal graphs: An algorithmic perspective. Internet Mathematics **12**(4) (2016) 239–280
2. Casteigts, A., Flocchini, P., Quattrociocchi, W., Santoro, N.: Time-varying graphs and dynamic networks. IJPEDS **27**(5) (2012) 387–408
3. Mertzios, G.B., Michail, O., Chatzigiannakis, I., Spirakis, P.G.: Temporal network optimization subject to connectivity constraints. In: 40th International Colloquium on Automata, Languages, and Programming (ICALP 2013). (2013) 657–668
4. Michail, O., Spirakis, P.G.: Traveling salesman problems in temporal graphs. Theor. Comput. Sci. **634** (2016) 1–23
5. Kempe, D., Kleinberg, J.M., Kumar, A.: Connectivity and inference problems for temporal networks. In: ACM Symposium on Theory of Computing (STOC 2000). (2000) 504–513
6. Kuhn, F., Lynch, N.A., Oshman, R.: Distributed computation in dynamic networks. In: 42nd ACM Symposium on Theory of Computing (STOC 2010). (2010) 513–522
7. Michail, O., Chatzigiannakis, I., Spirakis, P.G.: Causality, influence, and computation in possibly disconnected synchronous dynamic networks. J. Parallel Distrib. Comput. **74**(1) (2014) 2016–2026

* Supported in part by the School of EEE/CS of the University of Liverpool.

On Temporally Connected Graphs and the Issue of Label Redundancy

Eleni C. Akrida

Department of Computer Science, University of Liverpool, UK

A preliminary version of this work appeared in the 13^{th} Workshop on Approximation and Online Algorithms, WAOA 2015, under the title On Temporally Connected Graphs of Small Cost.

We study here design issues of small cost temporally connected temporal graphs, under various constraints. We mainly consider undirected graphs of n vertices, where each edge has an associated set of positive integers (labels) that demonstrate the discrete availability instances of the edge. A journey from vertex u to vertex v is a path from u to v where successive path edges have strictly increasing labels. A temporal graph is said to be temporally connected if there exists a (u, v)-journey for every pair of distinct vertices u, v. We present a simple polynomial-time algorithm that can be used to check whether a given temporal graph is temporally connected. We then consider the case in which a designer of temporal graphs can freely choose availability instances for all edges and aims for temporal connectivity with very small cost; the cost is the total number of availability instances used. We give a simple polynomial-time procedure which derives designs of cost linear in n. We also show that the above procedure is (almost) optimal when the underlying graph is a tree, by proving a lower bound on the cost for any tree. However, there are pragmatic cases where one is not free to design a temporally connected graph anew, but is instead *given* a temporal graph design with the claim that it is temporally connected, and wishes to make it more cost-efficient by removing labels without destroying temporal connectivity (redundant labels).

Our main technical result is that computing the maximum number of redundant labels is APX-hard, i.e., there is no PTAS unless $P = NP$. On the positive side, we show that in dense graphs with random edge availabilities, there is asymptotically almost surely a very large number of redundant labels. A temporal design may, however, be minimal, i.e., no redundant labels exist. We show the existence of minimal temporal designs with at least $n \log n$ labels.

How to Explore a Fast Changing World?

Zvi Lotker

Ben Gurion University, Israel

Motivated by real world networks and use of algorithms based on random walks on these networks we study the simple random walks on *dynamic* undirected graphs with fixed underlying vertex set, i.e., graphs which are modified by inserting or deleting edges at every step of the walk. We are interested in the expected time needed to visit all the vertices of such a dynamic graph, the *cover time*, under the assumption that the graph is being modified by an oblivious adversary. It is well known that on connected static undirected graphs the cover time is polynomial in the size of the graph. On the contrary and somewhat counter-intuitively, we show that there are adversary strategies which force the expected cover time of a simple random walk on connected dynamic graphs to be exponential. We relate this result to the cover time of static directed graphs. In addition we provide a simple strategy, the *lazy* random walk, that guarantees polynomial cover time regardless of the changes made by the adversary.

Broadcasting in Dynamic Radio Networks

Mohamad Ahmadi

Department of Computer Science, University of Freiburg, Germany

An essential characteristic of radio networks is the unreliability of communication links in packet delivery. A small obstacle in the communication environment or even small changes in the weather conditions may cause links to fluctuate in strength (e.g., [1]). Moreover, the mobility of wireless devices in the network might even lead to a dynamic communication topology. Hence, when establishing a theoretical foundation for the design and analysis of practical radio network algorithms, it seems necessary to consider models which explicitly allow unreliable or unpredictable behavior. To capture such dynamic behavior, in this talk, we consider the dual graph model [2] and the more general dynamic network models studied in [3, 4]. In these models, the set of nodes in the network is fixed while the set of edges, representing the communication topology, changes over time. We measure the dynamic connectivity of the network by its interval connectivity, the largest T such that in every time interval of length T, the stable edges induce a connected graph.

We discuss the broadcasting problem, where a piece of information has to be sent from a source node to all the nodes of a network. This is a fundamental problem in distributed computing and in particular also one of the key problems to study in the context of dynamic networks. We show that in dynamic networks, the complexity of broadcasting or also other related information dissemination problems depends on the strength of the adversary providing the dynamic network topology, as well as on the stability and dynamic connectivity of the network. We will first discuss upper and lower bounds for the single-message global broadcast for various combinations of dynamic connectivity and adversary strength [5, 6, 2, 4]. We will then also show techniques to extend these solutions to be applicable to solve multi-message broadcast [7].

References

1. Srinivasan, K., Kazandjieva, M.A., Agarwal, S., Levis, P.: The beta-factor: measuring wireless link burstiness. In: 6th International Conference on Embedded Networked Sensor Systems (SenSys 2008). (2008) 29–42
2. Kuhn, F., Lynch, N.A., Newport, C.C., Oshman, R., Richa, A.W.: Broadcasting in unreliable radio networks. In: ACM Symposium on Principles of Distributed Computing (PODC 2010). (2010) 336–345
3. Kuhn, F., Lynch, N.A., Oshman, R.: Distributed computation in dynamic networks. In: 42nd ACM Symposium on Theory of Computing (STOC 2010). (2010) 513–522
4. Ahmadi, M., Ghodselahi, A., Kuhn, F., Molla, A.R.: The cost of global broadcast in dynamic radio networks. In: 19th Int. Conf. on Principles of Distributed Systems (OPODIS 2015). (2015)
5. Clementi, A.E.F., Monti, A., Pasquale, F., Silvestri, R.: Broadcasting in dynamic radio networks. J. Comput. Syst. Sci. **75**(4) (2009) 213–230
6. Ghaffari, M., Lynch, N.A., Newport, C.C.: The cost of radio network broadcast for different models of unreliable links. In: ACM Symposium on Principles of Distributed Computing (PODC 2013). (2013) 345–354
7. Ahmadi, M., Kuhn, F.: Multi-message broadcast in dynamic radio networks. In: 12th Int. Symp. on Algorithms and Experiments for Wireless Sensor Networks. (2016)

Counting on Anonymous Dynamic Networks: Bounds and Algorithms

Giuseppe Antonio Di Luna

School of Electrical Engineering and Computer Science, University of Ottawa, Canada

Counting is a fundamental problem of every distributed system as it represents a basic building block to implement high level abstractions [1–3]. We focus on deterministic counting algorithms, that is we assume that no source of randomness is available to processes. We consider a dynamic system where processes do not leave the computation while there is an adversary that continuously changes the communication graph connecting such processes. The adversary is only constrained to maintain at each round a connected topology, i.e. 1-interval connectivity $\mathcal{G}(1-IC)$ [4]. In such environment, it has been shown, [5], that counting cannot be solved without a leader. Therefore, we assume that all processes are anonymous but the distinguished leader.

In the talk we will discuss bounds and algorithms for counting in the aforementioned framework. Our bounds are obtained investigating networks where the distance between the leader and an anonymous process is persistent across rounds and is at most h, we denote such networks as $\mathcal{G}(PD)_h$ [6]. Interestingly we will show that counting in $\mathcal{G}(PD)_2$ requires $\Omega(\log|V|)$ rounds even when the bandwidth is unlimited. This implies that counting in networks with constant dynamic diameter requires a number of rounds that is function of the network size. We will discuss other results concerning the accuracy of counting algorithms and the robustness of the aforementioned bound. For the possibility side we will show an optimal counting algorithm for $\mathcal{G}(PD)_h$ networks and a counting algorithm for $\mathcal{G}(1-IC)$ networks.

References

1. Kempe, D., Dobra, A., Gehrke, J.: Gossip-based computation of aggregate information. In: 44th Symposium on Foundations of Computer Science (FOCS 2003). (2003) 482–491
2. Massoulié, L., Merrer, E.L., Kermarrec, A., Ganesh, A.J.: Peer counting and sampling in overlay networks: random walk methods. In: ACM Symposium on Principles of Distributed Computing (PODC 2006). (2006) 123–132
3. Mosk-Aoyama, D., Shah, D.: Computing separable functions via gossip. In: ACM Symposium on Principles of Distributed Computing (PODC 2006). (2006) 113–122
4. Kuhn, F., Lynch, N.A., Oshman, R.: Distributed computation in dynamic networks. In: 42nd ACM Symposium on Theory of Computing (STOC 2010). (2010) 513–522
5. Michail, O., Chatzigiannakis, I., Spirakis, P.G.: Brief announcement: Naming and counting in anonymous unknown dynamic networks. In: 26th International Symposium on Distributed Computing (DISC 2012). (2012) 437–438
6. Luna, G.A.D., Baldoni, R.: Brief announcement: Investigating the cost of anonymity on dynamic networks. In: ACM Symposium on Principles of Distributed Computing (PODC 2015). (2015) 339–341

Reaching Agreement in a Dynamic Distributed System

Peter Robinson

Department of Computer Science, Royal Holloway, University of London, UK

Modern-day networks are inherently dynamic. For example, mobile nodes in wireless networks move in and out of each others transmission range or, in the context of data centers, crashed nodes need to be replaced by new machines without disrupting the services provided by the network.

In this talk, we are going to consider distributed algorithms for dynamic networks where the topology can change from round to round and even the participants in the network can vary over time. Instances of such networks include P2P networks, mobile ad-hoc networks, and social networks. We assume that the evolution of the network is under the control of an adversary. The nodes are not aware of these changes in advance and start out having only local knowledge of the network topology.

We introduce the framework of [1] for modeling dynamic networks and discuss distributed algorithms for agreement problems under various adversarial assumptions [2]. Then, we study the more difficult agreement problem of electing a leader in a dynamic network with Byzantine nodes and present the algorithm of [3], which solves several subproblems that are of independent interest: In particular, we show how to implement an almost-everywhere public coin with constant bias in a dynamic network with Byzantine nodes and provide a mechanism for enabling honest nodes to store information reliably in the network.

All of our algorithms are scalable, as they require messages of $O(polylog(n))$ bits and run in $O(polylog(n))$ rounds, where n is the number of nodes in the network.

References

1. Augustine, J., Pandurangan, G., Robinson, P., Upfal, E.: Distributed agreement in dynamic peer-to-peer networks. J. Comput. Syst. Sci. **81**(7) (2015) 1088–1109
2. Augustine, J., Pandurangan, G., Robinson, P.: Fast byzantine agreement in dynamic networks. In: ACM Symposium on Principles of Distributed Computing (PODC 2013). (2013) 74–83
3. Augustine, J., Pandurangan, G., Robinson, P.: Fast byzantine leader election in dynamic networks. In: 29th International Symposium on Distributed Computing (DISC 2015). (2015) 276–291

DISC 2016

Paris, France

Proceedings of the First Workshop on Dynamic Graphs in Distributed Computing (DGDC'16)