Lecture Notes in Computer Science **9988**

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

More information about this series at http://www.springer.com/series/7407

Jukka Suomela (Ed.)

Structural Information and Communication Complexity

23rd International Colloquium, SIROCCO 2016
Helsinki, Finland, July 19–21, 2016
Revised Selected Papers

Springer

Editor
Jukka Suomela
Aalto University
Espoo
Finland

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-48313-9 ISBN 978-3-319-48314-6 (eBook)
DOI 10.1007/978-3-319-48314-6

Library of Congress Control Number: 2016955506

LNCS Sublibrary: SL1 – Theoretical Computer Science and General Issues

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume contains the papers presented at SIROCCO 2016, the 23rd International Colloquium on Structural Information and Communication Complexity, held during July 19–21, 2016, in Helsinki, Finland.

This year we received 50 submissions in response to the call for papers. Each submission was reviewed by at least three reviewers; we had a total of 18 Program Committee members and 57 external reviewers. The Program Committee decided to accept 25 papers: 24 normal papers and one survey-track paper. Fabian Kuhn, Yannic Maus, and Sebastian Daum received the SIROCCO 2016 Best Paper Award for their work "Rumor Spreading with Bounded In-Degree." Selected papers will also be invited to a special issue of the *Theoretical Computer Science* journal.

In addition to the 25 contributed talks, the conference program included a keynote lecture by Yoram Moses, invited talks by Keren Censor-Hillel, Adrian Kosowski, Danupon Nanongkai, and Thomas Sauerwald, and the award lecture by Masafumi (Mark) Yamashita, the recipient of the 2016 SIROCCO Prize for Innovation in Distributed Computing.

I would like to thank all authors for their high-quality submissions and all speakers for their excellent talks. I am grateful to the Program Committee and all external reviewers for their efforts in putting together a great conference program, to the Steering Committee chaired by Andrzej Pelc for their help and support, and to everyone who was involved in the local organization for making it possible to have SIROCCO 2016 in sunny Helsinki.

Finally, I would like to thank our sponsors for their support: the Federation of Finnish Learned Societies, Helsinki Institute for Information Technology HIIT, and Helsinki Doctoral Education Network in Information and Communications Technology (HICT) provided financial support, Springer not only helped with the publication of these proceedings but also sponsored the best paper award, Aalto University provided administrative support and helped with the conference venue, and EasyChair kindly provided a free platform for managing paper submissions and the production of this volume.

September 2016 Jukka Suomela

Organization

Program Committee

Leonid Barenboim	Open University of Israel
Jérémie Chalopin	LIF, CNRS and Aix Marseille Université, France
Yuval Emek	Technion, Israel
Paola Flocchini	University of Ottawa, Canada
Pierre Fraigniaud	CNRS and Université Paris Diderot, France
Janne H. Korhonen	Reykjavik University, Iceland
Evangelos Kranakis	Carleton University, Canada
Christoph Lenzen	MPI for Informatics, Germany
Friedhelm Meyer auf der Heide	Heinz Nixdorf Institute and University of Paderborn, Germany
Danupon Nanongkai	KTH Royal Institute of Technology, Sweden
Calvin Newport	Georgetown University, USA
Gopal Pandurangan	University of Houston, USA
Merav Parter	MIT, USA
Peter Robinson	Queen's University Belfast, UK
Thomas Sauerwald	University of Cambridge, UK
Stefan Schmid	Aalborg University, Denmark
Jukka Suomela	Aalto University, Finland
Przemysław Uznański	ETH Zürich, Switzerland

Additional Reviewers

Abu-Affash, A. Karim
Alistarh, Dan
Amram, Gal
Assadi, Sepehr
Augustine, John
Blin, Lelia
Burman, Janna
Censor-Hillel, Keren
Chatterjee, Soumyottam
Cseh, Ágnes
Das, Shantanu
Delporte-Gallet, Carole
Devismes, Stephane
Di Luna, Giuseppe
Friedrichs, Stephan
Förster, Klaus-Tycho
Geissmann, Barbara
Gelashvili, Rati
Gelles, Ran
Giakkoupis, George
Godard, Emmanuel
Graf, Daniel
Halldorsson, Magnus M.
Jung, Daniel
Karousatou, Christina
Kling, Peter
Konrad, Christian
Konwar, Kishori
Kuszner, Lukasz
Kuznetsov, Petr

Labourel, Arnaud
Lempiäinen, Tuomo
Malatyali, Manuel
Mamageishvili, Akaki
Medina, Moti
Molla, Anisur Rahaman
Musco, Cameron
Navarra, Alfredo
Pacheco, Eduardo
Pemmaraju, Sriram
Podlipyan, Pavel
Purcell, Christopher
Rabie, Mikaël
Rajsbaum, Sergio
Ravi, Srivatsan
Santoro, Nicola
Sardeshmukh, Vivek B.
Schneider, Johannes
Scquizzato, Michele
Setzer, Alexander
Sourav, Suman
Su, Hsin-Hao
Tonoyan, Tigran
Trehan, Chhaya
Tschager, Thomas
Yamauchi, Yukiko
Yu, Haifeng

Laudatio

It is a pleasure to award the 2016 SIROCCO Prize for Innovation in distributed computing to Masafumi (Mark) Yamashita. Mark has presented many original ideas and important results that have enriched the theoretical computer science community and the distributed computing community, such as his seminal work "Computing on Anonymous Networks" (with T. Kameda), which introduced the notion of "view" and has inspired all the subsequent investigations on computability in anonymous networks, as well as his work on coteries, on self-stabilization, and on polling games, among others.

The prize is awarded for his lifetime achievements, but especially for introducing the computational universe of autonomous mobile robots to the algorithmic community and to the distributed community in particular. This has opened a new and exciting research area that has now become an accepted mainstream topic in theoretical computer science (papers on "mobile robots" now appear in all major theory conferences and journals) and clearly in distributed computing. The fascinating new area of research it opened is now under investigation by many groups worldwide.

The introduction of this area to the theory community was actually made in his SIROCCO paper [1]. The full version was then published in the *SIAM Journal on Computing* [2]. (This paper currently has more than 500 citations.)

The paper deals with the problem of coordination among autonomous robots moving on a plane. This and subsequent papers on this topic provided the first indications about which tasks can be accomplished using multiple deterministic, autonomous, and identical robots in a collaborative manner. The formal model for mobile robots introduced in the paper (called the Suzuki–Yamashita or SYM model) provides a nice abstraction that makes it easy to analyze algorithms but still captures many of the difficulties of coordination between the robots. Many of the recent results on distributed robotics are based on either this model or extensions of it. The paper provided the characterization (in terms of geometric pattern formation) of all tasks that can be performed by such teams of deterministic robots and provided some fundamental impossibility results including the impossibility of gathering two oblivious robots. A more recent work [3] extends the characterization to the model where robots are memory-less, thus showing the exact difference between oblivious robots and robots having memory.

The 2015 Award Committee[1]:
Thomas Moscibroda (Microsoft)
Guy Even (Tel Aviv University)
Magnús Halldórsson (Reykjavik University)
Shay Kutten (Technion) – Chair
Andrzej Pelc (Université du Québec en Outaouais)

[1] We wish to thank the nominators for the nomination and for contributing greatly to this text.

Selected Publications Related to Masafumi (Mark) Yamashita's Contribution:

1. Suzuki, I., Yamashita, M.: Distributed anonymous mobile robots. In: Proceedings of the 3rd International Colloquium on Structural Information and Communication Complexity, Siena, Italy, 6–8 June, pp. 313–330 (1996)
2. Suzuki, I., Yamashita, M.: Distributed anonymous mobile robots. SIAM J. Comput. **28**(4), 1347–1363 (1999)
3. Yamashita, M., Suzuki, I.: Characterizing geometric patterns formable by oblivious anonymous mobile robots. Theor. Comput. Sci. **411**(26–28), 2433–2453 (2010)
4. Dumitrescu, A., Suzuki, I., Yamashita, M.: Motion planning for metamorphic systems: feasibility, decidability, and distributed reconfiguration. IEEE Trans. Robot. **20**(3), 409–418 (2004)
5. Souissi, S., Defago, X., Yamashita, M.: Using eventually consistent compasses to gather memory-less mobile robots with limited visibility. ACM Trans. Auton. Adapt. Syst. **4**(1), #9 (2009)
6. Das, S., Flocchini, P., Santoro, N., Yamashita, M.: Forming sequences of geometric patterns with oblivious mobile robots. Distrib. Comput. **28**(2), 131–145 (2015)
7. Fujinaga, N., Yamauchi, Y., Ono, H., Shuji, K., Yamashita, M.: Pattern formation by oblivious asynchronous mobile robots. SIAM J. Comput. **44**(3), 740–785 (2015)

Towards a Theory of Formal Distributed Systems
(SIROCCO Prize Lecture)

Masafumi Yamashita

Department of Informatics, Kyushu University, Fukuoka, Japan
mak@inf.kyushu-u.ac.jp

In the title, the word *towards* means incomplete, immature or not ready for presenting, and the word *formal* means unrealistic, imaginary or useless. Please keep them in mind.

One might find similarity between two phenomena, seabirds competing for good nesting places in a small island and cars looking (or fighting) for parking space. Regardless of whether conscious or unconscious, they are solving a conflict resolution problem, which is a well-known problem in distributed computing (in computer science). This suggests us there are many (artificial or natural) systems that are in the face of solving distributed problems.

Lamport and Lynch [1] claimed "although one usually speak of a distributed system, it is more accurate to speak of a distributed *view* of a system," after defining the word *distributed* to mean spread across space. This claim seems to imply that every system is a distributed system at least from the view of atoms or molecules, and may be in the face of solving a distributed problem, when we concentrate on the distributed view, like seabirds and cars in the example above.

An abstract distributed view, which we call a *formal distributed system* (FDS), describes how system elements interact logically. Our final goal is to understand a variety of FDSs and compare them in terms of the solvability of distributed problems.

We first propose a candidate for the model of FDS in such a way that it can describe a wide variety of FDSs, and explain that many of the models of distributed systems (including ones suitable to describe biological systems) can be described as FDSs. Compared with other distributed system models, FDSs have two features: First, the system elements are modeled by points in d-dimensional space, where d can be greater than 3. Second incomputable functions can be taken as transition functions (corresponding to distributed algorithms).

We next explain some of our ongoing works in three research areas, localization, symmetry breaking and self-organization. In localization, we discuss the simplest problem of locating a single element with limited visibility to the center of a line segment. In symmetry breaking, we observe how elements in 3D space can eliminate some symmetries. Finally in self-organization, we examine why natural systems appear to have richer autonomous properties than artificial systems, despite that the latter would have stronger interaction mechanisms, e.g., unique identifiers, memory, synchrony, and so on.

Reference

1. Lamport, L., Lynch, N.: Distributed computing: models and methods, In: van Leeuwen, J. (ed.) Handbook of Theoretical Computer Science. Formal Models and Semantics, Chap. 18, vol. B, pp. 1157–1199. MIT Press/Elsevier (1990)

A Principled Way of Designing Efficient Distributed Protocols

(Keynote Lecture)

Yoram Moses

Technion - Israel Institute of Technology, Technion City, Haifa 32000, Israel
moses@ee.technion.ac.il

The focus of this invited talk is a demonstration of how knowledge-based reasoning can be used to design an efficient protocol in a stepwise manner. The ***Knowledge of Preconditions*** principle, denoted by (**K***o***P**), can be formulated as a theorem that applies in all the various distributed systems models [2]. Intuitively, it states that if some condition φ is a necessary condition for process i to perform action α, then, under every protocol that satisfies this constraint, process i must ***know*** φ when it performs α. We denote i knowing something by 'K_i'. **K***o***P** thus states that if φ is a necessary condition for i performing α, then $K_i\varphi$ is also a necessary condition for i performing α. Thus, for example, a process the enters the critical section (CS) in a mutual exclusion protocol must know that the CS is empty when it enters. Similarly, if an ATM must only provide cash to a customer that has a sufficient positive balance, then the ATM must know that the customer has such a balance.

The talk illustrates the design of an unbeatable protocol for Consensus based on the **K***o***P**, along the lines of [1]. Based on the Validity property in the specification. In Consensus, a process can decide 0 only if some initial value is 0. The **K***o***P** immediately implies that following every correct protocol for Consensus, a process must *know* of an initial value of 0 when it decides 0. We consider binary Consensus, in which values are 0 or 1. We seek the optimal rule for deciding 1 in a protocol in which deciding on 0 is favored, by having every process that knows of a 0 decide 0. The Agreement property of Consensus implies that a process cannot decide 1 at a point when other processes decide 0. It follows by **K***o***P** that a process that decides 1 must know that nobody is deciding 0. In particular, it must know that no active process knows of a 0. A combinatorial analysis of when a process knows that nobody knows of a 0 is performed, yielding a natural condition that can be easily computed. The outcome is an elegant and efficient protocol that strictly dominates all known protocols for Consensus in the synchronous crash-failure model, which cannot be strictly dominated.

A video of a similar invited talk given in February 2016 appears in IHP talk.

References

1. Castañeda, A., Gonczarowski, Y.A., Moses, Y.: Unbeatable consensus. In: Kuhn, F. (ed.) DISC 2014. LNCS, vol. 8784, pp. 91–106. Springer, Heidelberg (2014). Full version available on arXiv
2. Moses,Y.: Relating knowledge and coordinated action: the knowledge of preconditions principle. In: Proceedings of the 15th Conference on Theoretical Aspects of Rationality and Knowledge, pp. 207–216 (2015)

Invited Talks

The Landscape of Lower Bounds for the Congest Model

Keren Censor-Hillel

Technion, Department of Computer Science, Haifa, Israel
ckeren@cs.technion.ac.il

Introduction. We address the classic Congest model of distributed computing [8], in which n nodes of a network communicate in synchronous rounds, during each of which they send messages of $O(\log n)$ bits on the available links. We focus on solving *global* graph problems, which require $\Omega(D)$ rounds of communication even in the LOCAL model in which messages can be of unbounded size. While in the LOCAL model D rounds suffice for solving these problems by gathering all information at a single node and solving the problem on its local processor, the Congest model imposes additional bandwidth restrictions, making such problems harder. Below we discuss some known lower bounds for global problems in Congest, glimpse into some new results, and discuss open questions.

Computing the Diameter. One of the lead examples of a global graph problem is that of computing the diameter. In the Congest model, the diameter can be computed in $O(n)$ rounds [7, 9], and a beautiful lower bound of $\Omega(n/\log n)$, which we describe next, is known even for small values of D [5, 7].

In a nutshell, the lower bound is obtained through a reduction from the wellknown 2-party communication complexity problem of set-disjointness, in which Alice and Bob receive input vectors $\bar{x}, \bar{y}$ of length k, respectively, and need to output whether there is an index $1 \leq i \leq k$ for which $x_i = y_i = 1$. The reduction is obtained by constructing a graph of n nodes, with two sets of nodes that are connected by a complete matching and some additional edges within each set. Alice and Bob are each responsible for one of the two sets, in terms of simulating the distributed algorithm for the nodes within that set. Any message that needs to be sent within a set is simulated locally, and communication is only needed for messages that cross the cut between the two sets.

The crux is that Alice and Bob add edges within their sets according to their input vectors, where a 0 input for index i corresponds to adding the corresponding edge. This is done in a way that promises that the diameter of the resulting graph determines the answer to the set-disjointness problem. The parameters are taken such that $k = \Theta(n^2)$, and since set-disjointness is known to require $\Omega(k)$ bits of communication, and the size of the cut between the two sets of nodes is of size $\Theta(n)$ and the message size is of log n bits, the end result is a lower bound of $\Omega(n/\log n)$ rounds.

Keren Censor-Hillel—Supported in part by the Israel Science Foundation (grant 1696/14).

Recently, Abboud et al. [1] introduce a new construction that allows obtaining a similar near-linear lower bound for computing the diameter. The main technical contribution is a *bit-gadget*, which allows the cut between the sets of Alice and Bob to be of size only $\Theta(\log n)$ and allows taking $k = \Theta(n)$, giving a lower bound of $\Omega(n/\log^2 n)$. While this is worse than the previously mentioned bound by a logarithmic factor, the strength of the bit-gadget is in reducing the size of the cut and having a sparse construction, which then allows improving the state-of-the-art for additional problems: It gives the first near-linear lower bounds for a $(3/2-\epsilon)$-approximation for the diameter, for computing or approximating the radius, for approximating all eccentricities, and for verifying certain types of spanners. These can also be made to work for constant degree graphs.

Constructing a Minimum Spanning Tree (MST). To exemplify another type of lower bounds for Congest that uses set-disjointness albeit in a different manner, consider the problem of finding an MST.

We next describe the key idea of the $\Omega(\sqrt{n/\log n}+D)$-round lower bound of [11]. This bound is given for the problem of subgraph connectivity, which can be easily be shown to reduce to finding an MST. A base graph is given and some of its edges are marked to be in the subgraph H, according to the inputs of Alice and Bob. It is shown that H is connected iff the inputs are not disjoint. To simulate the required distributed algorithm, Alice and Bob need to exchange information on certain edges of the graph in a dynamic way. That is, there is no static partition of the nodes between the 2 players which makes the complexity depend on the size of the cut, but rather the assignment of the nodes to be simulated changes from round to round and is not a partition. Thus, while the cut between Alice and Bob's nodes in each round is large, the *used cut* is $O(\log n)$, and choosing $k = O(n^{1/2})$ gives almost the claimed lower bound (for ease of description, this is a slightly weakened simplification of the lower bound). In our context, the interesting thing here is that although this is also a reduction from set-disjointness, the framework is entirely different from the distance computation lower bounds.

Constructing Additive Spanners. Recently, another type of Congest lower bounds has been introduced, for constructing additive spanners. Previous work obtains various spanners in the Congest model [2, 3, 10], and a lower bound of $\Omega(D)$ is given in [10].

A $+\beta$-pairwise spanner of G is a subgraph S for which, given $P \subseteq V$, for every $u, v \in P$, it holds that $d_S(u,v) \leq d_G(u,v) + \beta$. In addition to algorithms for purely additive spanners, [4] give lower bounds, of which we describe the $\Omega(p/n \log n)$ lower bound for constructing (+2)-pairwise spanners with $|P| = p$. Consider here $p = n^{3/2}$. Define the (p, m)-partial-complement problem as follows. Alice receives a set x of p elements in $\{1, \ldots, m\}$ and Bob needs to output a set y of $m/2$ elements in $\{1, \ldots, m\} \setminus x$. First, it is proven that (p, m)-partial-complement requires $\Omega(p)$ bits of communication. Then, a distributed algorithm for constructing a +2-spanner is simulated on the graph that consists of an Erdös graph with girth 6 and $\Theta(n^{3/2})$ edges that is simulated by Bob, whose nodes are connected by a complete matching to an equal size independent set of nodes that are simulated by Alice. The only unknown is the set P, given only to Alice. To decide on an edge of the graph to be omitted from the constructed

spanner, Bob must know that the corresponding pair on Alice's side is not in P, otherwise its removal increases the distance between these nodes from 3 to 7, violating the +2 stretch requirement. Since Bob must remove $\Theta(n^{3/2})$ edges, this implies solving the (p, m)-partial-complement problem, hence requires $\Omega(p/n \log n)$ rounds. This gives a lower bound of a new flavor, where the graph is known to both players, and the uncertainty only comes from the unknown set of pairs.

Discussion. There are many additional lower bounds that are not described here.

Many specific questions are still open in the above various settings and problems. One example is that, while our lower bounds for distance computations apply to sparse graphs, they are far from being planar. It is known that an MST can be computed in $O(D \log D)$ rounds in planar graphs [6], which raises the question of whether distance computations can be performed faster than the general lower bound as well. Specifically, can the diameter of planar graphs be computed in $o(n/\text{polylog } n)$ rounds?

References

1. Abboud, A., Censor-Hillel, K., Khoury, S.: Near-linear lower bounds for distributed distance computations, even in sparse networks. In: Gavoille, C., Ilcinkas, D. (eds.) DISC 2016. LNCS, vol. 9888, pp. 29–42. Springer, Heidelberg (2016)
2. Baswana, S., Kavitha, T., Mehlhorn, K., Pettie, S.: Additive spanners and (alpha, beta)-spanners. ACM Trans. Algorithms **7**(1), 5 (2010)
3. Baswana, S., Sen, S.: A simple and linear time randomized algorithm for computing sparse spanners in weighted graphs. Random Struct. Algorithms **30**(4), 532–563 (2007)
4. Censor-Hillel, K., Kavitha, T., Paz, A., Yehudayoff, A.: Distributed construction of purelyadditive spanners. In: Gavoille, C., Ilcinkas, D. (eds.) DISC 2016. LNCS, vol. 9888, pp. 129–142. Springer, Heidelberg (2016)
5. Frischknecht, S., Holzer, S., Wattenhofer, R.: Networks cannot compute their diameter in sublinear time. In: SODA, pp. 1150–1162 (2012)
6. Ghaffari, M., Haeupler, B.: Distributed algorithms for planar networks II: low-congestion shortcuts, mst, and min-cut. In: SODA, pp. 202–219 (2016)
7. Holzer, S., Wattenhofer, R.: Optimal distributed all pairs shortest paths and applications. In: PODC, pp. 355–364 (2012)
8. Peleg, D.: Distributed computing: a locality-sensitive approach. In: Society for Industrial and Applied Mathematics (2000)
9. Peleg, D., Roditty, L., Tal, E.: Distributed algorithms for network diameter and girth. In: Czumaj, A., Mehlhorn, K., Pitts, A., Wattenhofer, R. (eds.) ICALP 2012. LNCS, vol. 7392, pp. 660–672. Springer, Heidelberg (2012)
10. Pettie, S.: Distributed algorithms for ultrasparse spanners and linear size skeletons. Distrib. Comput. **22**(3), 147–166, (2010)
11. Sarma, A.D., Holzer, S., Kor, L., Korman, A., Nanongkai, D., Pandurangan, G., Peleg, D., Wattenhofer, R.: Distributed verification and hardness of distributed approximation. SIAM J. Comput. **41**(5), 1235–1265 (2012)

What Makes a Distributed Problem Truly Local?

Adrian Kosowski

Inria and IRIF, CNRS — Université Paris Diderot, 75013 Paris, France
adrian.kosowski@inria.fr

Abstract. In this talk we attempt to identify the characteristics of a task of distributed network computing, which make it easy (or hard) to solve by means of fast local algorithms. We look at specific combinatorial tasks within the LOCAL model of distributed computation, and rephrase some recent algorithmic results in a framework of constraint satisfaction. Finally, we discuss the issue of efficient computability for relaxed variants of the LOCAL model, involving the so-called non-signaling property.

In distributed network computing, autonomous computational entities are represented by the nodes of an undirected *system graph*, and exchange information by sending messages along its edges. A major line of research in this area concerns the notion of *locality*, and asks how much information about its neighborhood a node needs to collect in order to solve a given computational task. In particular, in the seminal LOCAL model [19], the complexity of a distributed algorithm is measured in term of number of *rounds*, where in each round all nodes synchronously exchange data along network links, and subsequently perform individual computations. A t-round algorithm is thus one in which every node exchanges data with nodes at distance at most t (i.e., at most t hops away) from it.

Arguably, the most important class of local computational tasks concerns *symmetry breaking*, and several forms of such tasks have been considered, including the construction of proper *graph colorings* [3–9, 11, 15, 17, 18, 22], of *maximal independent sets* (MIS) [1, 4, 5, 14, 16, 18], as well as edge-based variants of these problems (cf. e.g. [21]). In this talk we address the following question: What makes some symmetry-breaking problems in the LOCAL model easier than others?

We note that the LOCAL model has two flavors, involving the design of deterministic and randomized algorithms, which are clearly distinct [8]. When considering randomized algorithms, for n-node graphs of maximum degree Δ, a hardness separation between the randomized complexities of the specific problems of MIS and $(\Delta + 1)$-coloring has recently been observed [11, 14]. No analogous separation is as yet known when considering deterministic solutions to these problems. We look at some partial evidence in this direction, making use of the recently introduced framework of *conflict coloring* representations [9] for local combinatorial problems. A conflict coloring representation captures a distributed task through a set of local constraints on edges

---

This talk includes results of joint work with: P. Fraigniaud, C. Gavoille, M. Heinrich, and M. Markiewicz.

of the system graph, thus constituting a special case of the much broader class of constraint satisfaction problems (CSP) with binary constraints. Whereas all local tasks are amenable to a conflict coloring formulation, one may introduce a natural *constraint density* parameter, which turns out to be inherently smaller for some problems than for others. For example, for the natural representation of the $(\Delta + 1)$-coloring task, the constraint density is $1/(\Delta + 1)$, while for any accurate representation of MIS, the constraint density is at least 1/2. We discuss implications of how low constraint density (notably, much smaller than $1/\Delta$) may be helpful when finding solutions to a distributed task, especially when applying the so-called *shattering method* [20] in a randomized setting, and more directly, when designing faster deterministic algorithms through a direct attack on the conflict coloring representation of the task [9].

We close this talk with a discussion of relaxed variants of the LOCAL model, inspired by the physical concept of non-signaling. In a computational framework, the *non-signaling property* can be stated as the following necessary (but not sufficient) property of the LOCAL model: for any $t > 0$, given two subsets of nodes S_1 and S_2 of the system graph, such that the distance between the nearest nodes of S_1 and S_2 is greater than t, in any t-round LOCAL algorithm, the outputs of nodes from S_1 must be (probabilistically) independent of the inputs of nodes from S_2. We point out that for a number of symmetry breaking tasks in the LOCAL model, the currently best known asymptotic lower bounds can be deduced solely by exploiting the non-signaling property. This is the case for problems such as MIS [10, 14] or 2-coloring of the ring [10]. On the other hand, such an implication is not true for, e.g., the $\Omega(\log^* n)$ lower bound on the number of rounds required to 3-color the ring [15] — this lower bound follows from different (stronger) properties of the LOCAL model [12, 13]. This leads us to look at the converse question: How to identify conditions under which non-signaling solutions to a distributed task can be converted into an algorithm in the LOCAL model? We note some progress in this respect for quantum analogues of the LOCAL model [2].

References

1. Alon, N., Babai, L., Itai, A.: A fast and simple randomized parallel algorithm for the maximal independent set problem. J. Algorithms **7**(4), 567–583 (1986)
2. Arrighi, P., Nesme, V., Werner, R.F.: Unitarity plus causality implies localizability. J. Comput. Syst. Sci. **77**(2), 372–378 (2011)
3. Barenboim, L.: Deterministic $(\Delta + 1)$-coloring in sublinear (in Δ) time in static, dynamic and faulty networks. In: Proceedings of the 34th ACM Symposium on Principles of Distributed Computing (PODC), pp. 345–354 (2015)
4. Barenboim, L., Elkin, M.: Distributed $(\Delta + 1)$-coloring in linear (in Δ) time. In: Proceedings of the 41th ACM Symposium on Theory of Computing (STOC), pp. 111–120 (2009)
5. Barenboim, L., Elkin, M.: Distributed graph coloring: fundamentals and recent developments. In: Synthesis Lectures on Distributed Computing Theory. Morgan & Claypool Publishers (2013)
6. Barenboim, L., Elkin, M., Kuhn, F.: Distributed $(\Delta + 1)$-coloring in linear (in Δ) time. SIAM J. Comput. **43**(1), 72–95 (2014)

7. Barenboim, L., Elkin, M., Pettie, S., Schneider, J.: The locality of distributed symmetry breaking. In: Proceedings of the 53rd IEEE Symposium on Foundations of Computer Science (FOCS), pp. 321–330 (2012)
8. Chang, Y-J., Kopelowitz, T., Pettie, S.: An exponential separation between randomized and deterministic complexity in the LOCAL model, In: Proceedings 57th IEEE Symposium on Foundations of Computer Science (FOCS) (2016, to appear). http://arxiv.org/abs/1602.08166.
9. Fraigniaud, P., Heinrich, M., Kosowski, A.: Local conflict coloring, In: Proceedings of the 57th IEEE Symposium on Foundations of Computer Science (FOCS) (2016, to appear). http://arxiv.org/abs/1511.01287
10. Gavoille, C., Kosowski, A., Markiewicz, M.: What can be observed locally? In: Keidar, I. (ed.) DISC 2009. LNCS, vol. 5805, pp. 243–257. Springer, Heidelberg (2009)
11. Harris, D.G., Schneider, J., Su, H-H.: Distributed ($\Delta + 1$)-coloring in sublog-arithmic rounds, In: Proceedings of the 48th Annual Symposium on the Theory of Computing (STOC), pp. 465–478 (2016)
12. Holroyd, A.E., Liggett, T.M.: Finitely dependent coloring. Submitted preprint. http://arxiv.org/abs/1403.2448
13. Holroyd, A.E. Liggett, T.M.: Symmetric 1-dependent colorings of the integers. Electron. Commun. Probab. **20**(31) (2015)
14. Kuhn, F., Moscibroda, T., Wattenhofer, R.: What cannot be computed locally! In: Proceedings of the 23rd ACM Symposium on Principles of Distributed Computing (PODC), pp. 300–309 (2004)
15. Linial, N.: Locality in distributed graph algorithms. SIAM J. Comput. 21(1), 193–201 (1992)
16. Luby. M. A simple parallel algorithm for the maximal independent set problem. SIAM J. Comput. **15**, 1036–1053 (1986)
17. Naor. M.: A lower bound on probabilistic algorithms for distributive ring coloring. SIAM J. Discrete Math. **4**(3), 409–412 (1991)
18. Panconesi, A. Srinivasan, A.: Improved distributed algorithms for coloring and network decomposition problems. In: Proceedings of the 24th ACM Symposium on Theory of Computing (STOC), pp. 581–592 (1992)
19. Peleg, D.: Distributed computing: a locality-sensitive approach. SIAM (2000). Philadelphia, PA
20. Schneider, J., Wattenhofer, R.: A new technique for distributed symmetry breaking. In: Proceedings of the 29th ACM Symposium on Principles of Distributed Computing (PODC), pp. 257–266 (2010)
21. Suomela, J.: Survey of local algorithms. ACM Comput. Surv. **45**(2), 24 (2013)
22. Szegedy, M., Vishwanathan, S.: Locality based graph coloring. In: Proceedings of the 25th ACM Symposium on Theory of Computing (STOC), pp. 201–207 (1993)

Some Challenges on Distributed Shortest Paths Problems, A Survey

Danupon Nanongkai

KTH Royal Institute of Technology, Stockholm, Sweden
danupon@gmail.com
https://sites.google.com/site/dannanongkai/

Abstract. In this article, we focus on the time complexity of computing distances and shortest paths on distributed networks (the CONGEST model). We survey previous key results and techniques, and discuss where previous techniques fail and where major new ideas are needed. This article is based on the invited talk given at SIROCCO 2016. The slides used for the talk are available at the webpage of SIROCCO 2016 (http://sirocco2016.hiit.fi/programme/#invited).

Keywords: Shortest paths · Graph algorithms · Distributed algorithms

Our focus is on solving the *single-source shortest paths* problem on undirected weighted distributed networks. The network is modeled by the CONGEST model, and the goal is for every node to know its distance to a given source node. The algorithm should run with the least number of *rounds* possible (known as *time complexity*). (See, e.g., [8] for detailed descriptions.) Through a series of studies (e.g. [1, 3, 4, 8, 10, 11, 12, 14]), we now know that

1. any distributed algorithm with polynomial approximation ratio needs $\tilde{\Omega}(\sqrt{n}+D)$ rounds [3][1], and
2. there is a deterministic $(1+\epsilon)$-approximation algorithm that takes $\tilde{O}(\epsilon^{O(1)}(\sqrt{n}+D))$ rounds [1, 8].

Here, n and D are the number of nodes and the network diameter, respectively, and $\tilde{\Omega}$ and $\tilde{O}$ hide $\log^{O(1)} n$ factors. The above results imply that we already know the best number of rounds an approximation algorithm can achieve, modulo some lower-order terms. The case of *exact* algorithm is, however, widely open. The best exact algorithm we know of takes $O(n)$ rounds, due to the distributed version of the Bellman-Ford algorithm. Beating this bound is the first open problem we highlight:

Open problem 1: Is there an algorithm that can solve the single-source shortest paths (or simply compute the distance between two given nodes) *exactly* in time that is *sublinear* in n, i.e. in $\tilde{O}(n^{1-\epsilon})$ rounds for some constant $\epsilon > 0$?

Note that whether we can solve graph problems exactly in sublinear time (in n) is interesting for many graph problems (e.g. the minimum cut problem [6, 13]).

[1] This lower bound holds for randomized algorithms and, in fact, even for quantum algorithms [5].

An equally interesting question is whether we can solve the *all-pairs* shortest paths problem exactly in linear-time (in n). We already know that we can get a $(1+\epsilon)$-approximate solution in such running time.

One challenge in answering the above open problems is to avoid computing *k-source h-hop distances*. The h-hop distance between nodes u and v, denoted by $dist^h(u,v)$, is the (weighted) length of the shortest path among paths between u and v containing at most h edges. In the k-source h-hop distances problem, we are given k source nodes $s_1, s_2, \ldots, s_k$ and want to make every node u knows its distance to every source node s_i. An $\tilde{O}(k+n)$ distributed algorithm for solving this problem was presented in [12] and was an important subroutine in subsequent algorithms (e.g. [1, 8]). The drawback of this subroutine is that it only provides $(1+\epsilon)$-approximate distances. Unfortunately, obtaining exact distances within the same running time is impossible, as Lenzen and Patt-Shamir [11] showed that such algorithm requires $\tilde{\Omega}(kh)$ rounds.

Another open problem (raised before in [12]) is the *directed case* (referred to as the *asymmetric case* in [12]). This is when we think of each edge (u,v) as two *directed* edges, one from u to v and the other from v to u, and the weight of the two edges might be different. (Note that the directions and edge weight do not affect the communication between u and v.) Obviously, the lower bound of $\tilde{\Omega}(\sqrt{n}+D)$ [3] for the undirected case also holds for this case. Using the techniques in [12], we can get a $(1+\epsilon)$-approximation $\tilde{\Omega}(\sqrt{nD}+D)$-time algorithm. If we do not care about the approximation ratio, and simply want to know whether there is a directed path from the source to each node (this problem is called *single-source reachability*), then the running time can be slightly improved to $\tilde{\Omega}(\sqrt{n}D^{1/4}+D)$ [7]

Open problem 2: Is there an algorithm that can solve the *directed* single-source shortest paths (or just reachability) with any approximation ratio in $\tilde{O}(\sqrt{n}+D)$ rounds?

The main challenge in answering this open problem is to avoid the use of *sparse spanner* and related structures. A spanner is a subgraph that approximately preserves the distance between every pairs of nodes. Spanner and other relevant structures, such as emulator and hopset were used previously as the main tools to obtain tight upper bounds for the undirected case (see, e.g., [1, 8]). Unfortunately, similar structures do not exist on directed graphs. A sparse spanner, for example, do not exist for a complete bipartite graph with edges directed from left to right; removing any edge (u,v) from such graph will cause the distance from u to v to increase from one to infinity.

The last open problem we highlight is on *congested cliques*, i.e. when the network is fully-connected. For approximately solving the single-source shortest paths problem, we already have a satisfying algorithm with polylogarithmic time and $(1+\epsilon)$-approximation ratio [1, 8]. The best $(1+\epsilon)$-approximation algorithm for all-pairs shortest paths take $\tilde{O}(n^{0.15715})$ time [2]. For exact solutions, both single-source and all-pairs shortest paths have the best known running time of $\tilde{O}(n^{1/3})$ [2].

Open problem 3: Can we improve the running time of [2] for solving single-source shortest paths exactly and all-pairs shortest paths $(1+\epsilon)$-approximately on congested cliques?

The above problem is interesting because of its connection to algebraic techniques. Its answer might lead us to understand these techniques better. See [2, 9] for algebraic tools developed so far on congested cliques.

References

1. Becker, R., Karrenbauer, A., Krinninger, S., Lenzen, C.: Approximate undirected transshipment and shortest paths via gradient descent. CoRR abs/1607.05127 (2016)
2. Censor-Hillel, K., Kaski, P., Korhonen, J.H., Lenzen, C., Paz, A., Suomela, J.: Algebraic methods in the congested clique. In: Symposium on Principles of Distributed Computing, PODC, pp. 143–152 (2015)
3. Das Sarma, A., Holzer, S., Kor, L., Korman, A., Nanongkai, D., Pandurangan, G., Peleg, D., Wattenhofer R.: Distributed verification and hardness of distributed approximation. SIAM J. Comput. **41**(5) (2012). Announced at STOC 2011
4. Elkin, M.: An unconditional lower bound on the time-approximation trade-off for the distributed minimum spanning tree problem. SIAM J. Comput. **36**(2) (2006). Announced at STOC 2004
5. Elkin, M., Klauck, H., Nanongkai, D., Pandurangan, G.: Can quantum communication speed up distributed computation? In: Symposium on Principles of Distributed Computing, PODC (2014)
6. Ghaffari, M., Kuhn, F.: Distributed minimum cut approximation. In: Afek, Y. (ed.) DISC 2013. LNCS, vol. 8205, pp. 1–15. Springer, Heidelberg (2013)
7. Ghaffari, M., Udwani, R.: Brief announcement: distributed single-source reachability. In: ACM Symposium on Principles of Distributed Computing, PODC, pp. 163–165 (2015)
8. Henzinger, M., Krinninger, S., Nanongkai, D: A deterministic almost-tight distributed algorithm for approximating single-source shortest paths. In: Symposium on Theory of Computing, STOC (2016)
9. Le Gall, F.: Further algebraic algorithms in the congested clique model and applications to graph-theoretic problems. In: Gavoille, C., Ilcinkas, D. (eds.) DISC 2016. LNCS, vol. 9888, pp. 57–70. Springer, Heidelberg (2016)
10. 10. Lenzen, C., Patt-Shamir, B.: Fast routing table construction using small messages. In: Symposium on Theory of Computing, STOC (2013)
11. Lenzen, C., Patt-Shamir, B.: Fast partial distance estimation and applications. In: ACM Symposium on Principles of Distributed Computing, PODC, pp. 153–162 (2015)
12. Nanongkai, D.: Distributed approximation algorithms for weighted shortest paths. In: ACM Symposium on Theory of Computing, STOC (2014)
13. Nanongkai, D., Su, H.: Almost-tight distributed minimum cut algorithms. In: Kuhn, F. (ed.) DISC 2014. LNCS, vol. 8784, pp. 439–453. Springer, Heidelberg (2014)
14. Peleg, D., Rubinovich, V.: A near-tight lower bound on the time complexity of distributed minimum-weight spanning tree construction. SIAM J. Comput. **30**(5) (2000). Announced at FOCS 1999

A Survey on Smoothing Networks

Thomas Sauerwald

Computer Laboratory, University of Cambridge, USA

Abstract. In this talk we will consider smoothing networks (a.k.a. balancing networks) that accept an arbitrary stream of tokens on input and routes them to output wires. Pairs of wires can be connected by balancers that direct arriving tokens alternately to its two outputs. We first discuss some classical results and relate smoothing networks to their siblings, including sorting and counting networks. Then we will present some results on randomised smoothing networks, where balancers are initialised randomly. Finally, we will explore stronger notions of smoothing networks including a model where an adversary can specify the input and the initialisation of all balancers.

References

1. Aspnes, J., Herlihy, M., Shavit, N.: Counting networks. J. ACM **41**(5), 1020–1048 (1994)
2. Batcher, K.E.: Sorting networks and their applications. In: American Federation of Information Processing Societies: AFIPS Conference Proceedings (1968) Spring Joint Computer Conference, pp. 307–314 (1968)
3. Busch, C., Mavronicolas, M.: A combinatorial treatment of balancing networks. J. ACM **43**(5), 794–839 (1996)
4. Dowd, M., Perl, Y., Rudolph, L., Saks, M.E.: The periodic balanced sorting network. J. ACM **36**(4), 738–757 (1989)
5. Friedrich, T., Sauerwald, T., Vilenchik. D.: Smoothed analysis of balancing networks. Random Struct. Algorithms **39**(1), 115–138 (2011)
6. Herlihy, M., Tirthapura, S.: Randomized smoothing networks. J. Parallel Distrib. Comput. **66**(5), 626–632 (2006)
7. Herlihy, M., Tirthapura, S.: Self-stabilizing smoothing and balancing networks. Distrib. Comput. **18**(5), 345–357 (2006)
8. Klugerman, M.: Small-depth counting networks and related topics, Ph.D. thesis, Massachusetts Institute of Technology, Department of Mathematics, September 1994
9. Klugerman, M., Plaxton, C.G.: Small-depth counting networks. In: Proceedings of the 24th Annual ACM Symposium on Theory of Computing (STOC 1992), pp. 417–428 (1992)
10. Mavronicolas, M. Sauerwald, T.: A randomized, o(log w)-depth 2 smoothing network. In: Proceedings of the 21st Annual ACM Symposium on Parallelism in Algorithms and Architectures (SPAA 2009), pp. 178–187 (2009)
11. Mavronicolas, M., Sauerwald, T.: The impact of randomization in smoothing networks. Distrib. Comput. **22**(5–6), 381–411 (2010)
12. Sauerwald, T., Sun, H.: Tight bounds for randomized load balancing on arbitrary network topologies. In: Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS 2012), pp. 341–350 (2012)

Contents

Message Passing

How Many Cooks Spoil the Soup? 3
Othon Michail and Paul G. Spirakis

Dynamic Networks of Finite State Machines 19
Yuval Emek and Jara Uitto

Setting Ports in an Anonymous Network: How to Reduce the Level of Symmetry?. 35
Ralf Klasing, Adrian Kosowski, and Dominik Pajak

Fooling Pairs in Randomized Communication Complexity 49
Shay Moran, Makrand Sinha, and Amir Yehudayoff

Public vs. Private Randomness in Simultaneous Multi-party Communication Complexity 60
Orr Fischer, Rotem Oshman, and Uri Zwick

Message Lower Bounds via Efficient Network Synchronization 75
Gopal Pandurangan, David Peleg, and Michele Scquizzato

Recent Results on Fault-Tolerant Consensus in Message-Passing Networks. . . 92
Lewis Tseng

Shared Memory

Asynchronous Coordination Under Preferences and Constraints 111
Armando Castañeda, Pierre Fraigniaud, Eli Gafni, Sergio Rajsbaum, and Matthieu Roy

Concurrent Use of Write-Once Memory. 127
James Aspnes, Keren Censor-Hillel, and Eitan Yaakobi

In the Search for Optimal Concurrency 143
Vincent Gramoli, Petr Kuznetsov, and Srivatsan Ravi

The *F*-Snapshot Problem 159
Gal Amram

t-Resilient Immediate Snapshot Is Impossible 177
Carole Delporte, Hugues Fauconnier, Sergio Rajsbaum, and Michel Raynal

Mobile Agents

Linear Search by a Pair of Distinct-Speed Robots 195
Evangelos Bampas, Jurek Czyzowicz, Leszek Gąsieniec, David Ilcinkas, Ralf Klasing, Tomasz Kociumaka, and Dominik Pająk

Deterministic Meeting of Sniffing Agents in the Plane. 212
Samir Elouasbi and Andrzej Pelc

Distributed Evacuation in Graphs with Multiple Exits 228
Piotr Borowiecki, Shantanu Das, Dariusz Dereniowski, and Łukasz Kuszner

Universal Systems of Oblivious Mobile Robots 242
Paola Flocchini, Nicola Santoro, Giovanni Viglietta, and Masafumi Yamashita

Collaborative Delivery with Energy-Constrained Mobile Robots 258
Andreas Bärtschi, Jérémie Chalopin, Shantanu Das, Yann Disser, Barbara Geissmann, Daniel Graf, Arnaud Labourel, and Matúš Mihalák

Communication Problems for Mobile Agents Exchanging Energy 275
Jurek Czyzowicz, Krzysztof Diks, Jean Moussi, and Wojciech Rytter

Data Dissemination and Routing

Asynchronous Broadcasting with Bivalent Beeps 291
Kokouvi Hounkanli and Andrzej Pelc

Sparsifying Congested Cliques and Core-Periphery Networks 307
Alkida Balliu, Pierre Fraigniaud, Zvi Lotker, and Dennis Olivetti

Rumor Spreading with Bounded In-Degree. 323
Sebastian Daum, Fabian Kuhn, and Yannic Maus

Whom to Befriend to Influence People 340
Manuel Lafond, Lata Narayanan, and Kangkang Wu

Approximating the Size of a Radio Network in Beeping Model 358
Philipp Brandes, Marcin Kardas, Marek Klonowski, Dominik Pająk, and Roger Wattenhofer

An Approximation Algorithm for Path Computation and Function Placement in SDNs 374
Guy Even, Matthias Rost, and Stefan Schmid

Transiently Consistent SDN Updates: Being Greedy is Hard. 391
Saeed Akhoondian Amiri, Arne Ludwig, Jan Marcinkowski, and Stefan Schmid

Author Index 407