An Introduction to Combinatorial Reconfiguration

Duc A. Hoang[1,2]

[1]VNU University of Science, Vietnam National University, Hanoi, Vietnam
hoanganhduc@hus.edu.vn

[2]Vietnam Institute for Advanced Study in Mathematics, Hanoi, Vietnam

March 12, 2024

Seminar at VIASM

Short Bio

- **Full Name (Vietnamese):** Hoàng Anh Đức
- **Name (in publications):** Duc A. Hoang
- **Education:**
  - **B.Math** @ VNU-HUS, Vietnam (K53, 2008–2013). Undergraduate Thesis Advisor: Phan Thị Hà Dương
  - **M.S.** and **Ph.D.** (Information Science) @ JAIST, Japan (2013–2018) under the advice of Prof. Ryuhei Uehara
- **Employment:**
  - **Lecturer** @ VNU-HUS, Vietnam (9/2018-12/2018)
  - **Postdoc** and then **Research Assistant** @ Kyutech, Japan under the direction of Prof. Toshiki Saitoh (04/2019–06/2021)
  - **Postdoc** @ KyotoU, Japan under the direction of Prof. Shin-ichi Minato (06/2021–01/2023)
  - **Lecturer** @ VNU-HUS, Vietnam (02/2023-present) and **Postdoc** @ VIASM (01/2024-present)
- **Web:** `https://hoanganhduc.github.io/`

Short Bio

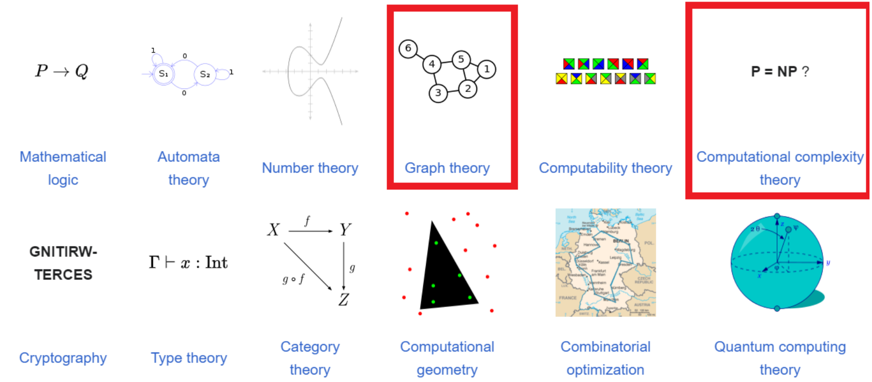


Figure: My research interests lie in the intersection of **Graph Theory** and **Computational Complexity Theory**. (Picture taken from Wikipedia at `https://en.wikipedia.org/wiki/Theoretical_computer_science/`)

Outline

1 Introduction
- Complexity Classes: P, NP, and PSPACE
- What is Combinatorial Reconfiguration?

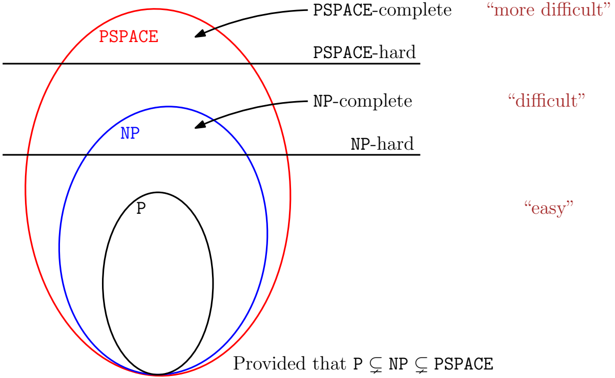
2 Motivation
- Understanding Solution Space
- Understanding Complexity Classes
- Understanding Complexity of Problem
- Robot Motion Planning
- Automated Restoring Power in Electrical Distribution Networks
- Reassigning Frequencies in Mobile Communication Networks
- And More ...

3 Online Resources
- For Motivating You Further
- Surveys and Wiki Page

Introduction

Complexity Classes: P, NP, and PSPACE

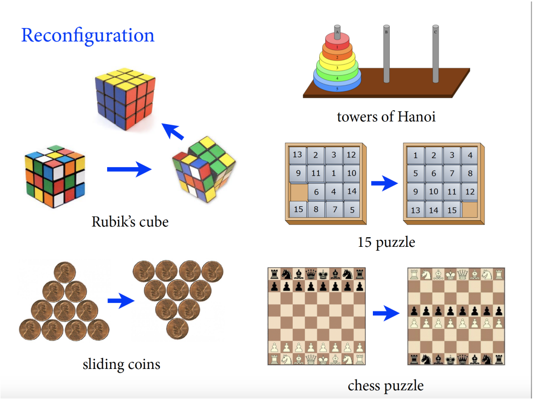
- We always talk about *decision problems* (output YES or NO)
- Complexity Classes
  - P: Problems can be "solved efficiently" in polynomial time
  - NP: Problems can be "verified efficiently" in polynomial time
  - PSPACE: Problems can be "solved efficiently" in polynomial space



What is Combinatorial Reconfiguration?

Reconfiguration Setting

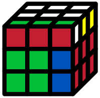
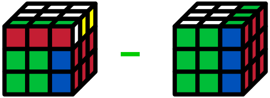
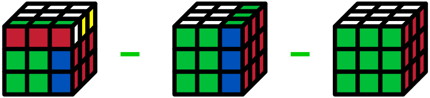
- A description of what *states* (≡ *configurations*) are
- One or more *allowed moves* between states (≡ *reconfiguration rule(s)*)



© Anna Lubiw @ CoRe 2019 (Aussois, France)

What is Combinatorial Reconfiguration?

Two major viewpoints: as a *process* or as a *graph*

Reconf. Process	**Reconf. Graph**	**Example** RUBIK'S CUBE
state	node	
move *[reconf. rule]*	edge	
seq. of moves *[reconf. sequence]*	path	

What is Combinatorial Reconfiguration?

Two major directions: *Algorithmic* and *Graph-Theoretic*

- **Algorithmic Questions**
  - REACHABILITY: Given two states S and T, is there a sequence of moves that *transforms S into T*?
  - SHORTEST TRANSFORMATION: Given two states S and T and some positive integer ℓ, is there a sequence of moves that *transforms S into T using at most ℓ moves*?
  - CONNECTIVITY: Is there a sequence of moves *between any pair of states*?
  - and so on
- **Graph-Theoretic Questions**
  - GRAPH PROPERTIES: Is the reconfiguration graph *connected*? *bipartite*? *Eulerian*? *Hamiltonian*?, and so on
  - GRAPH CLASSIFICATION: Does the reconfiguration graph *belong to some specific graph class* (e.g., planar graphs, perfect graphs, etc.)?
  - and so on

What is Combinatorial Reconfiguration?

The area was *first named* “Reconfiguration” in [Ito et al. 2011]

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

ELSEVIER

On the complexity of reconfiguration problems

Takehiro Ito [a,*], Erik D. Demaine [b], Nicholas J.A. Harvey [c], Christos H. Papadimitriou [d], Martha Sideri [e], Ryuhei Uehara [f], Yushi Uno [g]

[a] Graduate School of Information Sciences, Tohoku University, Aoba-yama 6-6-05, Sendai, 980-8579, Japan
[b] MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA
[c] Department of Combinatorics and Optimization, University of Waterloo, 200 University Ave. West, Waterloo, Ontario N2L 3G1, Canada
[d] Computer Science Division, University of California at Berkeley, Soda Hall 689, EECS Department, Berkeley, CA 94720, USA
[e] Department of Computer Science, Athens University of Economics and Business, Patision 76, Athens 10434, Greece
[f] School of Information Science, JAIST, Asahidai 1-1, Nomi, Ishikawa 923-1292, Japan
[g] Graduate School of Science, Osaka Prefecture University, 1-1 Gakuen-cho, Naka-ku, Sakai 599-8531, Japan

ARTICLE INFO

Article history:
Received 30 May 2010
Received in revised form 26 November 2010
Accepted 3 December 2010
Communicated by J. Díaz

Keywords:
Approximation
Graph algorithm
PSPACE-complete
Reachability on solution space

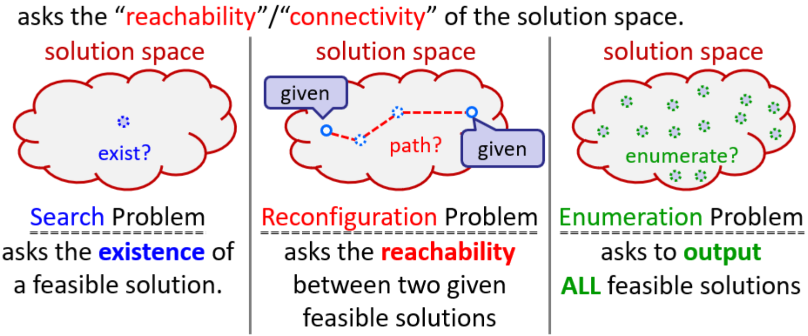
ABSTRACT

Reconfiguration problems arise when we wish to find a step-by-step transformation between two feasible solutions of a problem such that all intermediate results are also feasible. We demonstrate that a host of reconfiguration problems derived from NP-complete problems are PSPACE-complete, while some are also NP-hard to approximate. In contrast, several reconfiguration versions of problems in P are solvable in polynomial time.

Motivation

Understanding Solution Space

Reconfiguration is used in *studying the solution space of a computational problem*



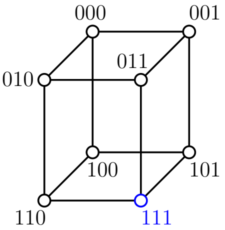
©Takehiro Ito @ CoRe 2017 (Banff, Canada)

Understanding Solution Space

SAT formula $\varphi = (x \wedge y) \vee z$

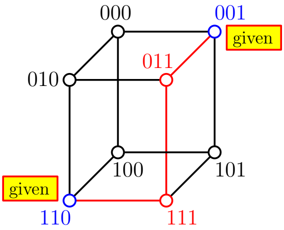
State $\equiv$ Feasible Solution: assignment of variables (x, y, z) makes φ true

Reconf. Rule: flip exactly one bit



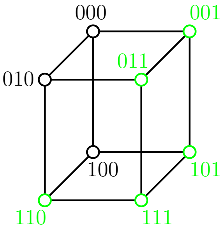
Search Problem

Check if **at least one** feasible solution exists among 2^n candidates for n variables



Reconfiguration Problem

Find **a sequence** of adjacent feasible solutions among 2^n candidates for n variables



Enumeration Problem

List **all** feasible solutions

among 2^n candidates for n variables

In **Reconfiguration** Problem, *we do NOT know which ones among the other $2^n - 2$ candidates are feasible solutions*

Understanding Solution Space

SAT RECONFIGURATION was *first studied* in [Gopalan et al. 2009] (**before** the area was first named)

SIAM J. COMPUT.
Vol. 38, No. 6, pp. 2330–2355

THE CONNECTIVITY OF BOOLEAN SATISFIABILITY: COMPUTATIONAL AND STRUCTURAL DICHOTOMIES*

PARIKSHIT GOPALAN†, PHOKION G. KOLAITIS‡, ELITZA MANEVA‡, AND CHRISTOS H. PAPADIMITRIOU§

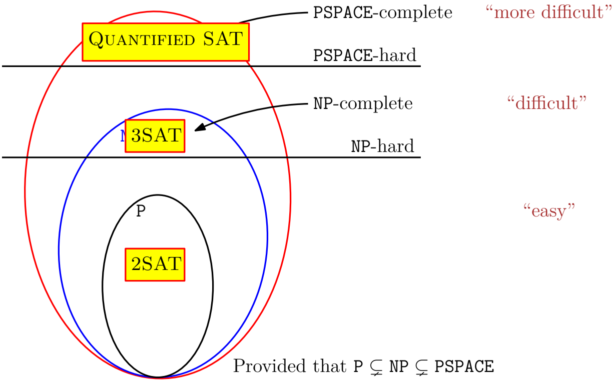
Abstract. Boolean satisfiability problems are an important benchmark for questions about complexity, algorithms, heuristics, and threshold phenomena. Recent work on heuristics and the satisfiability threshold has centered around the structure and connectivity of the solution space. Motivated by this work, we study structural and connectivity-related properties of the space of solutions of Boolean satisfiability problems and establish various dichotomies in Schaefer's framework. On the structural side, we obtain dichotomies for the kinds of subgraphs of the hypercube that can be induced by the solutions of Boolean formulas, as well as for the diameter of the connected components of the solution space. On the computational side, we establish dichotomy theorems for the complexity of the connectivity and *st*-connectivity questions for the graph of solutions of Boolean formulas. Our results assert that the intractable side of the computational dichotomies is PSPACE-complete, while the tractable side—which includes but is not limited to all problems with polynomial-time algorithms for satisfiability—is in P for the *st*-connectivity question, and in coNP for the connectivity question. The diameter of components can be exponential for the PSPACE-complete cases, whereas in all other cases it is linear; thus, diameter and complexity of the connectivity problems are remarkably aligned. The crux of our results is an expressibility theorem showing that in the tractable cases, the subgraphs induced by the solution space possess certain good structural properties, whereas in the intractable cases, the subgraphs can be arbitrary.

Key words. Boolean satisfiability, computational complexity, PSPACE, PSPACE-completeness, dichotomy theorems, graph connectivity

Understanding Complexity Classes

Reconfiguration *provides new insights into the understanding of complexity classes*

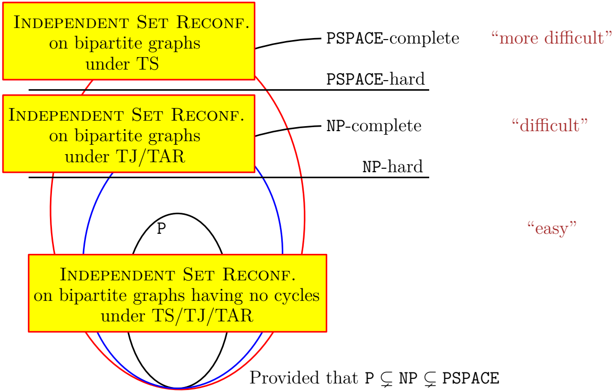
One can *"characterize" different complexity classes*
by *different* <u>*restricted*</u> *variants* of the *same problem*



Understanding Complexity Classes

Reconfiguration *provides new insights into the understanding of complexity classes*

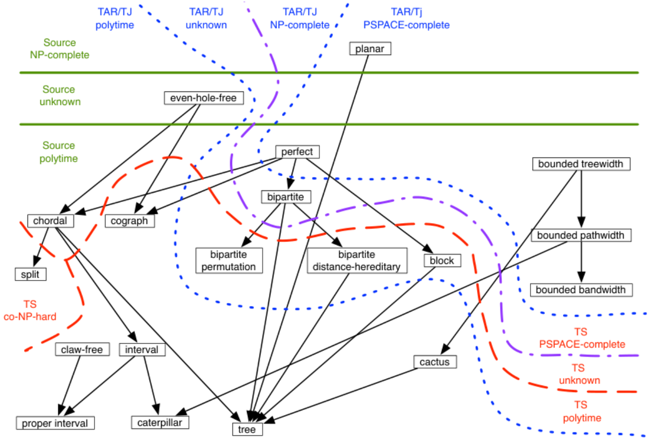
One can *"characterize" different complexity classes*
by *different reconfiguration variants* of the *same problem*



Provided that $\mathtt{P} \subsetneq \mathtt{NP} \subsetneq \mathtt{PSPACE}$

Understanding Complexity Classes

Independent Set Reconfiguration is *one of the most well-studied reconfiguration problems*



Taken from [Nishimura 2018], Figure 5

Understanding Complexity Classes

Before [Lokshtanov and Mouawad 2019], *most reconfiguration problems* are *either in* P *or* PSPACE*-complete*

The Complexity of Independent Set Reconfiguration on Bipartite Graphs

DANIEL LOKSHTANOV, University of Bergen, Norway
AMER E. MOUAWAD, American University of Beirut, Lebanon

We settle the complexity of the INDEPENDENT SET RECONFIGURATION problem on bipartite graphs under all three commonly studied reconfiguration models. We show that under the token jumping or token addition/removal model, the problem is NP-complete. For the token sliding model, we show that the problem remains PSPACE-complete.

CCS Concepts: • **Theory of computation → Graph algorithms analysis**; *Complexity classes*; *Problems, reductions and completeness*;

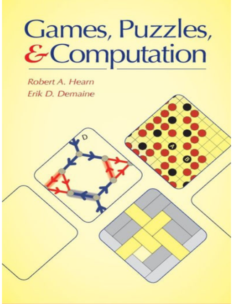
Additional Key Words and Phrases: Independent set, vertex cover, reconfiguration, solution space, bipartite graphs

ACM Reference format:
Daniel Lokshtanov and Amer E. Mouawad. 2018. The Complexity of Independent Set Reconfiguration on Bipartite Graphs. *ACM Trans. Algorithms* 15, 1, Article 7 (October 2018), 19 pages.
https://doi.org/10.1145/3280825

Understanding Complexity of Problem

Reconfiguration *provides new powerful tools for studying the complexity of a problem*

One of such tools is the *Nondeterministic Constraint Logic (NCL)*, *first introduced* in [Hearn and Demaine 2005]



Available online at www.sciencedirect.com
SCIENCE @ DIRECT®
ELSEVIER
Theoretical Computer Science 343 (2005) 72–96
Theoretical Computer Science
www.elsevier.com/locate/tcs

PSPACE-completeness of sliding-block puzzles and other problems through the nondeterministic constraint logic model of computation

Robert A. Hearn*, Erik D. Demaine

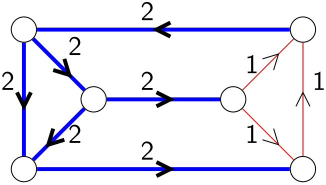
MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar Street, Cambridge, MA 02139, USA

Abstract

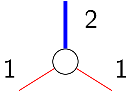
We present a nondeterministic model of computation based on reversing edge directions in weighted directed graphs with minimum in-flow constraints on vertices. Deciding whether this simple graph model can be manipulated in order to reverse the direction of a particular edge is shown to be PSPACE-complete by a reduction from Quantified Boolean Formulas. We prove this result in a variety of special cases including planar graphs and highly restricted vertex configurations, some of which correspond to a kind of passive constraint logic. Our framework is inspired by (and indeed a generalization of) the "Generalized Rush Hour Logic" developed by Flake and Baum [Theoret. Comput. Sci. 270(1–2) (2002) 895].

Understanding Complexity of Problem

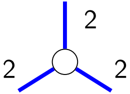
- **Input:**
  - Each *state/configuration* involves a graph having red (weight 1) and blue (weight 2) edges where each edge is oriented such that *(*) the sum of weights of in-coming arcs at each vertex is at least 2*
  - **Reconfiguration Rule:** Each *move* involves re-orienting an edge such that (*) is satisfied
- **Question:** Is there a sequence of moves that transforms one given configuration into another? (PSPACE-complete even on *planar graphs* having only *two types of vertices*)



(a) An NCL configuration



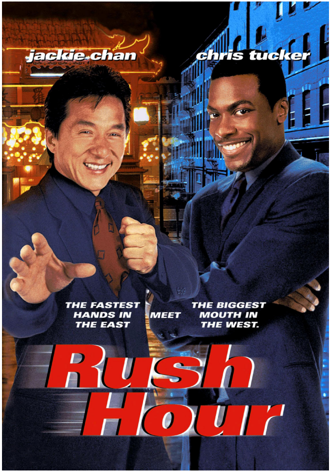
(b) AND vertex



(c) OR vertex

Understanding Complexity of Problem

RUSH HOUR (the puzzle, not the movie) *is* PSPACE*-complete*



Understanding Complexity of Problem

[Flake and Baum 2002]

Reduce from QUANTIFIED SAT. Use 3 "primitive devices" and more complicated "gadgets" built from the "devices"

ELSEVIER

Theoretical Computer Science 270 (2002) 895–911

Theoretical Computer Science

www.elsevier.com/locate/tcs

Mathematical Games

Rush Hour is PSPACE-complete, or "Why you should generously tip parking lot attendants"

Gary William Flake *, Eric B. Baum

NEC Research Institute, 4 Independence Way, Princeton, NJ 08540, USA

Received June 1999; revised February 2001; accepted February 2001
Communicated by A. Fraenkel

Abstract

Rush Hour is a children's game that consists of a grid board, several cars that are restricted to move either vertically or horizontally (but not both), a special target car, and a single exit on the perimeter of the grid. The goal of the game is to find a sequence of legal moves that allows the target car to exit the grid. We consider a slightly generalized version of the game that uses an $n \times n$ grid and assume that we can place the single exit and target car at any location we choose on initialization of the game.

In this work, we show that deciding if the target car can legally exit the grid is PSPACE-complete. Our constructive proof uses a lazy form of dual-rail reversible logic such that movement of "output" cars can only occur if logical combinations of "input" cars can also move. Emulating this logic only requires three types of devices (two switches and one crossover); thus, our proof technique can be easily generalized to other games and planning problems in which the same three primitive devices can be constructed.

Keywords: Games; PSPACE-completeness; Reversible logic; Motion planning; Dual-rail logic

[Hearn and Demaine 2005]

Reduce from NCL. Use 2 "gadgets"

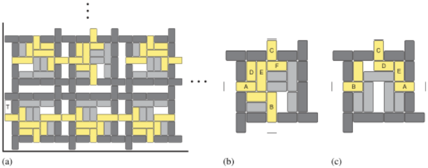


Fig. 14. Rush Hour layout and vertex gadgets. (a) Layout. (b) AND. (c) Protected OR.

generic crossover construction (Section 3.2), we do not need a crossover gadget. (We also do not need the miscellaneous wiring gadgets used in [4].)

Rush Hour layout. We tile the grid with our vertex gadgets, as shown in Fig. 14(a). One block (T) is the target, which must be moved to the bottom left corner; it is released when a particular port block slides into a vertex.

Dark-colored blocks represent the "cell walls", which unlike in our sliding-blocks construction are not shared. They are arranged so that they may not move at all. Light-colored blocks are "trigger" blocks, whose motion serves to satisfy the vertex constraints. Medium-colored blocks are fillers; some of them may move, but they do not disrupt the vertices' operation.

As in the sliding-blocks construction, edges are directed inward by sliding blocks out of the vertex gadgets; edges are directed outward by sliding blocks in. The layout ensures that no port block may ever slide out into an adjacent vertex; this helps keep the cell walls fixed.

Understanding Complexity of Problem

There are *interesting and nontrivial relations between the complexities of a source problem and its reconfiguration variant(s)*

Source	**Existence**	**Reconfiguration**
MATCHING	P [Edmonds 1965]	P [Ito et al. 2011]
3-COLORING	NP-*complete* [Stockmeyer 1973]	P [Bonsma and Cereceda 2009]
SHORTEST PATH	P	PSPACE-*complete* [Bonsma 2013]
INDEPENDENT SET on bipartite graphs	P [König-Egerváry Theorem 1931]	NP-*complete* under TJ/TAR [Lokshtanov and Mouawad 2019]

Understanding Complexity of Problem

There are *interesting and nontrivial relations between the complexities of a source problem and its reconfiguration variant(s)*

Graph	Independent Set	Independent Set Reconfiguration
general	NP-*complete* [Karp 1972]	PSPACE-*complete* [Ito et al. 2011]
perfect	P [Grötschel et al. 1981]	PSPACE-*complete* [Kamiński et al. 2012]
interval	P [Frank 1975]	P [Kamiński et al. 2012]; [Bonamy and Bousquet 2017]
Unknown	NP-*complete*	P

Robot Motion Planning

- Robots and obstacles are placed in an environment
- All robots are controlled by a central algorithm to perform some tasks (e.g., managing a warehouse or inventory) as a team
- Can we move robots to their final destinations without having collision with other robots or obstacles?
- Robot Motion $\equiv$ Token Reconfiguration [Hopcroft et al. 1984]

J. E. Hopcroft
Computer Science Department
Cornell University
Ithaca, NY 14853

J. T. Schwartz
Computer Science Department
Courant Institute of Mathematical Sciences
New York University
New York, NY 10012

M. Sharir
School of Mathematical Sciences
Tel Aviv University
Tel Aviv, Israel

On the Complexity of Motion Planning for Multiple Independent Objects; *PSPACE*-Hardness of the "Warehouseman's Problem"

Robot Motion Planning

After almost 40 years, the topic *remains actively studied*, e.g., see [Gupta et al. 2020]

Journal of Artificial Intelligence Research 69 (2020) 191-229 Submitted 12/2019; published 09/2020

The Parameterized Complexity of Motion Planning for Snake-Like Robots

Siddharth Gupta SIDDHART@POST.BGU.AC.IL
Guy Sa'ar SAAG@POST.BGU.AC.IL
Meirav Zehavi MEIRAVZE@BGU.AC.IL
Ben-Gurion University of the Negev, Israel

Abstract

We study the parameterized complexity of a variant of the classic video game Snake that models real-world problems of motion planning. Given a snake-like robot with an initial position and a final position in an environment (modeled by a graph), our objective is to determine whether the robot can reach the final position from the initial position without intersecting itself. Naturally, this problem models a wide-variety of scenarios, ranging from the transportation of linked wagons towed by a locomotor at an airport or a supermarket to the movement of a group of agents that travel in an "ant-like" fashion and the construction of trains in amusement parks. Unfortunately, already on grid graphs, this problem is PSPACE-complete. Nevertheless, we prove that even on general graphs, the problem is solvable in FPT time with respect to the size of the snake. In particular,

Automated Restoring Power in Electrical Distribution Networks

組合せ遷移
Combinatorial Reconfiguration

HOME MESSAGE PROJECT ACTIVITIES

[Research News] Development of Algorithm to Calculate Shortest Procedure for Power Restoration -- Applicable to Multi-Stage Power Restoration, with Application Expected in Distribution System Operations on Wider Areas

Date and time：2022.12.05

4th Asia Pacific Conference of the Prognostics and Health Management,
Tokyo, Japan, September 11 – 14, 2023 OS07-03

Algorithmic Study for Power Restoration in Electrical Distribution Networks

Jun Kawahara[1], Chuta Yamaoka[1], Takehiro Ito[2], Akira Suzuki[2], Daisuke Iioka[3], Shuhei Sugimura[4], Seiya Goto[4], and Takayuki Tanabe[4]

Automated Restoring Power in Electrical Distribution Networks

[Result of this research and its three characteristics]

Development of algorithm to calculate switching procedures for power restoration

1. Determines whether multi-stage power restoration is required
   → First considers rapid power restoration with little effect on the surroundings
2. Recovers power outages even in a scale necessitating multi-stage power restoration
   → **Guarantees the availability**, maintaining power supply to the healthy sections
3. Calculates the **shortest switching procedure** for executing power restoration

Enables the provision of mathematical evidence to the power restoration process

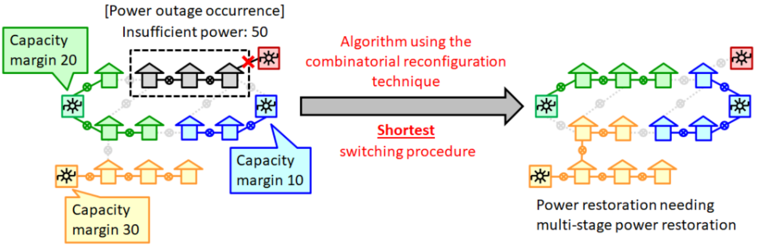
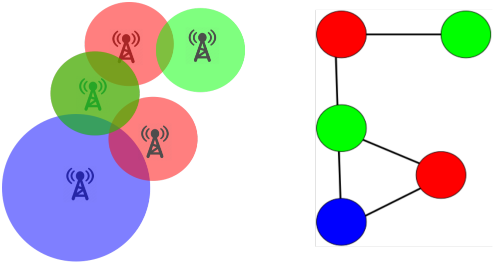


Figure 1: Results of this research and image of multi-stage power restoration.

Reassigning Frequencies in Mobile Communication Networks

- ($\star$) Two cell towers whose covering areas intersect must have different frequencies.
- Can we reassign the frequency of one tower at a time without affecting ($\star$)?



Frequency Re-Assignment $\equiv$ Vertex-Coloring Reconfiguration

Reassigning Frequencies in Mobile Communication Networks

The problem *was addressed* in [Han 2007] (using an approach that is different from “reconfiguration”)

ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Computers & Operations Research 34 (2007) 2939–2948

computers & operations research

www.elsevier.com/locate/cor

Frequency reassignment problem in mobile communication networks

Junghee Han*

College of Business Administration, Kangwon National University Hyoja-2Dong, Chunchon-Shi, Kangwon-Do, Republic of Korea

Available online 13 December 2005

Abstract

In this paper, we present a new frequency reassignment problem (FP) arising from the installation of new base stations for capacity expansion of a mobile telecommunication network, and develop two mathematical formulations along with some valid inequalities. Also, we develop a novel decomposition based heuristic procedure for solving large size problems. Computational results show that the developed valid inequalities are quite strong, and the developed heuristic procedure finds an optimal solution to the most test problems within reasonable time bound.

Keywords: Frequency reassignment; Integer programming; Valid inequality; Heuristic procedure

And More ...

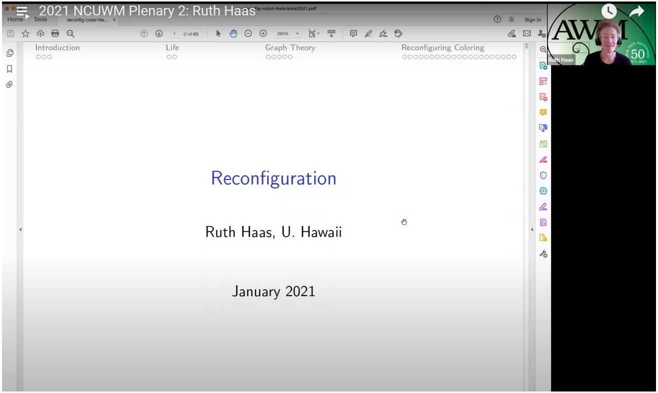
Reconfiguration has *emerged in several areas*, some of which are

- Computational geometry
  - Yan Alves Radtke et al. (2023). "Flip Graph Connectivity for Arrangements of Pseudolines and Pseudocircles". In: *Proceedings of SODA 2024*. SIAM, pp. 4849–4871. DOI: 10.1137/1.9781611977912.172
- Rerouting (shortest) paths
  - Kshitij Gajjar et al. (2022). "Reconfiguring Shortest Paths in Graphs". In: *Proceedings of AAAI 2022*. Vol. 36. 9, pp. 9758–9766. DOI: 10.1609/aaai.v36i9.21211
- Quantum complexity theory
  - Sevag Gharibian and Dorian Rudolph (2023). "Quantum Space, Ground Space Traversal, and How to Embed Multi-Prover Interactive Proofs into Unentanglement". In: *Proceedings of ITCS 2023*. Ed. by Yael Tauman Kalai. Vol. 251. LIPIcs. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 53:1–53:23. DOI: 10.4230/LIPIcs.ITCS.2023.53
- and so on

Online Resources

For Motivating You Further

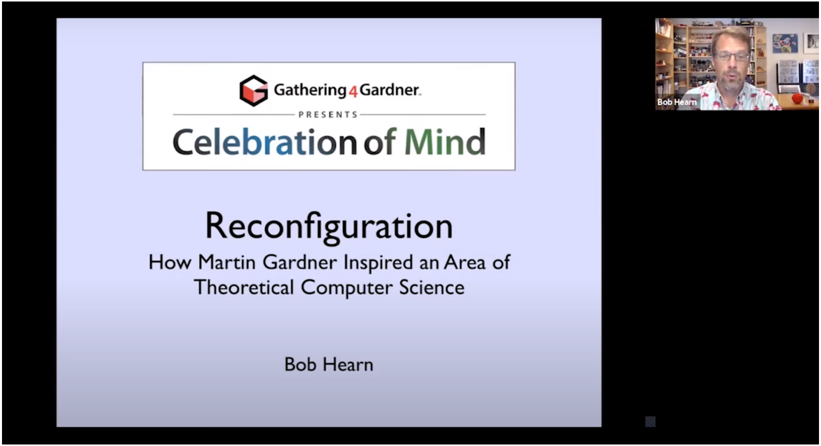
A nice and inspiring introduction to Reconfiguration in Graph Coloring (and other contexts) by Prof. Ruth Haas (U. Hawaii) at the NCUWM (Nebraska Conference for Undergraduate Women in Mathematics) 2021



https://www.youtube.com/watch?v=gApwRCEC89Q

For Motivating You Further

An inspiring talk in 2021 by Robert A. Hearn—one of the authors who introduced NCL [Hearn and Demaine 2005]



https://www.youtube.com/watch?v=4cWVjhBTDSY

For Motivating You Further

A more technical introduction at WALCOM (International Conference and Workshops on Algorithms and Computation) 2022 about Reconfiguration by Prof. Takehiro Ito (Tohoku Univ.)—one of the leading experts in this area

https://youtu.be/gwrIyuT3F8w?t=21308

For Motivating You Further

Mathematics and Art: Unifying Perspectives 18

Heather M. Russell and Radmila Sazdanovic

Contents

Introduction 498
Mathematics in Art 499
Mathematics as an Artistic Inspiration 499
Mathematics as an Artistic Tool and Medium 502
The Interplay of Art, Culture, and Mathematics 505
Artistic Ideas in Mathematics 509
Graphs and Their Visualizations 510
Examples of Graphs 513
Unifying Perspectives 520
Conclusion 522
Cross-References 523
References 523

Abstract

In this chapter, we explore the interconnection of mathematics and art. We discuss mathematics as a lens to understand artwork and investigate how mathematical thinking and mathematical tools contribute to the process of creating art. Turning then to the manifestation of art within mathematics, we introduce ideas and constructions from mathematical graph theory that can be appreciated

Heather M. Russell and Radmila Sazdanovic (2021). "Mathematics and Art: Unifying Perspectives". In: *Handbook of the Mathematics of the Arts and Sciences*. Ed. by Bharath Sriraman. Springer, pp. 497–525. DOI: 10.1007/978-3-319-57072-3_125

Surveys and Wiki Page

- **General Surveys**
  - Jan van den Heuvel (2013). "The Complexity of Change". In: *Surveys in Combinatorics*. Vol. 409. London Mathematical Society Lecture Note Series. Cambridge University Press, pp. 127–160. DOI: 10.1017/cbo9781139506748.005
  - Naomi Nishimura (2018). "Introduction to Reconfiguration". In: *Algorithms* 11.4, p. 52. DOI: 10.3390/a11040052
- **Surveys on Specific Problems**
  - C.M. Mynhardt and S. Nasserasr (2019). "Reconfiguration of Colourings and Dominating Sets in Graphs". In: *50 years of Combinatorics, Graph Theory, and Computing*. Ed. by Fan Chung et al. 1st. CRC Press, pp. 171–191. DOI: 10.1201/9780429280092-10
  - Nicolas Bousquet et al. (2022). "A survey on the parameterized complexity of the independent set and (connected) dominating set reconfiguration problems". In: *arXiv preprint*. arXiv: 2204.10526
- **Online Wiki:** `http://reconf.wikidot.com/` (I am one of the maintainers of this site)

CAME TO MY PARTY
YOU DID,
THANK YOU I MUST

References (1/6)

Bonamy, Marthe and Nicolas Bousquet (2017). "Token Sliding on Chordal Graphs". In: *Proceedings of the 43rd International Workshop on Graph-Theoretic Concepts in Computer Science, WG 2017*. Ed. by H. Bodlaender and G. Woeginger. Vol. 10520. Lecture Notes in Computer Science. Springer, pp. 136–149. DOI: 10.1007/978-3-319-68705-6_10.

Bonsma, Paul S. (2013). "The Complexity of Rerouting Shortest Paths". In: *Theoretical Computer Science* 510, pp. 1–12. DOI: 10.1016/j.tcs.2013.09.012.

Bonsma, Paul S. and Luis Cereceda (2009). "Finding Paths Between Graph Colourings: PSPACE-Completeness and Superpolynomial Distances". In: *Theoretical Computer Science* 410.50, pp. 5215–5226. DOI: 10.1016/j.tcs.2009.08.023.

Bousquet, Nicolas, Amer E. Mouawad, Naomi Nishimura, and Sebastian Siebertz (2022). "A survey on the parameterized complexity of the independent set and (connected) dominating set reconfiguration problems". In: *arXiv preprint*. arXiv: 2204.10526.

References (2/6)

Edmonds, Jack (1965). “Paths, trees, and flowers”. In: *Canadian Journal of mathematics* 17, pp. 449–467. DOI: 10.4153/CJM-1965-045-4.

Flake, Gary William and Eric B. Baum (2002). “Rush Hour is PSPACE-complete, or “Why you should generously tip parking lot attendants””. In: *Theoretical Computer Science* 270.1-2, pp. 895–911. DOI: 10.1016/S0304-3975(01)00173-6.

Frank, András (1975). “Some polynomial algorithms for certain graphs and hypergraphs”. In: *Proceedings of the 5th British Combinatorial Conference, 1975*. Utilitas Mathematica.

Gajjar, Kshitij, Agastya Vibhuti Jha, Manish Kumar, and Abhiruk Lahiri (2022). “Reconfiguring Shortest Paths in Graphs”. In: *Proceedings of AAAI 2022*. Vol. 36. 9, pp. 9758–9766. DOI: 10.1609/aaai.v36i9.21211.

Gharibian, Sevag and Dorian Rudolph (2023). “Quantum Space, Ground Space Traversal, and How to Embed Multi-Prover Interactive Proofs into Unentanglement”. In: *Proceedings of ITCS 2023*. Ed. by Yael Tauman Kalai. Vol. 251. LIPIcs. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 53:1–53:23. DOI: 10.4230/LIPIcs.ITCS.2023.53.

References (3/6)

Gopalan, Parikshit, Phokion G. Kolaitis, Elitza N. Maneva, and Christos H. Papadimitriou (2009). "The Connectivity of Boolean Satisfiability: Computational and Structural Dichotomies". In: *SIAM Journal on Computing* 38.6, pp. 2330–2355. DOI: 10.1137/07070440x.

Grötschel, M., L. Lovász, and A. Schrijver (1981). "The ellipsoid method and its consequences in combinatorial optimization". In: *Combinatorica* 1.2, pp. 169–197. DOI: 10.1007/BF02579273.

Gupta, Siddharth, Guy Sa'ar, and Meirav Zehavi (2020). "The Parameterized Complexity of Motion Planning for Snake-Like Robots". In: *Journal of Artificial Intelligence Research* 69, pp. 191–229. DOI: 10.1613/jair.1.11864.

Han, Junghee (2007). "Frequency reassignment problem in mobile communication networks". In: *Computers & Operations Research* 34.10, pp. 2939–2948. DOI: 10.1016/j.cor.2005.11.005.

References (4/6)

- Hearn, Robert A. and Erik D. Demaine (2005). "PSPACE-Completeness of Sliding-Block Puzzles and Other Problems through the Nondeterministic Constraint Logic Model of Computation". In: *Theoretical Computer Science* 343.1-2, pp. 72–96. DOI: 10.1016/j.tcs.2005.05.008.
- Heuvel, Jan van den (2013). "The Complexity of Change". In: *Surveys in Combinatorics*. Vol. 409. London Mathematical Society Lecture Note Series. Cambridge University Press, pp. 127–160. DOI: 10.1017/cbo9781139506748.005.
- Hopcroft, John E., Jacob Theodore Schwartz, and Micha Sharir (1984). "On the Complexity of Motion Planning for Multiple Independent Objects; PSPACE-Hardness of the "Warehouseman's Problem"". In: *The International Journal of Robotics Research* 3.4, pp. 76–88. DOI: 10.1177/027836498400300405.
- Ito, Takehiro, Erik D. Demaine, Nicholas J. A. Harvey, Christos H. Papadimitriou, Martha Sideri, Ryuhei Uehara, and Yushi Uno (2011). "On the Complexity of Reconfiguration Problems". In: *Theoretical Computer Science* 412.12-14, pp. 1054–1065. DOI: 10.1016/j.tcs.2010.12.005.

References (5/6)

Kamiński, Marcin, Paul Medvedev, and Martin Milanič (2012). "Complexity of independent set reconfigurability problems". In: *Theoretical Computer Science* 439, pp. 9–15. DOI: 10.1016/j.tcs.2012.03.004.

Karp, Richard M. (1972). "Reducibility Among Combinatorial Problems". In: *Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA*. Ed. by Raymond E. Miller and James W. Thatcher. The IBM Research Symposia Series. Plenum Press, New York, pp. 85–103. DOI: 10.1007/978-1-4684-2001-2_9.

Lokshtanov, Daniel and Amer E. Mouawad (2019). "The Complexity of Independent Set Reconfiguration on Bipartite Graphs". In: *ACM Transactions on Algorithms* 15.1, 7:1–7:19. DOI: 10.1145/3280825.

References (6/6)

Mynhardt, C.M. and S. Nasserasr (2019). "Reconfiguration of Colourings and Dominating Sets in Graphs". In: *50 years of Combinatorics, Graph Theory, and Computing*. Ed. by Fan Chung, Ron Graham, Frederick Hoffman, Ronald C. Mullin, Leslie Hogben, and Douglas B. West. 1st. CRC Press, pp. 171–191. DOI: 10.1201/9780429280092-10.

Nishimura, Naomi (2018). "Introduction to Reconfiguration". In: *Algorithms* 11.4, p. 52. DOI: 10.3390/a11040052.

Radtke, Yan Alves, Stefan Felsner, Johannes Obenaus, Sandro Roch, Manfred Scheucher, and Birgit Vogtenhuber (2023). "Flip Graph Connectivity for Arrangements of Pseudolines and Pseudocircles". In: *Proceedings of SODA 2024*. SIAM, pp. 4849–4871. DOI: 10.1137/1.9781611977912.172.

Russell, Heather M. and Radmila Sazdanovic (2021). "Mathematics and Art: Unifying Perspectives". In: *Handbook of the Mathematics of the Arts and Sciences*. Ed. by Bharath Sriraman. Springer, pp. 497–525. DOI: 10.1007/978-3-319-57072-3_125.

Stockmeyer, Larry (1973). "Planar 3-colorability is polynomial complete". In: *ACM Sigact News* 5.3, pp. 19–25. DOI: 10.1145/1008293.1008294.