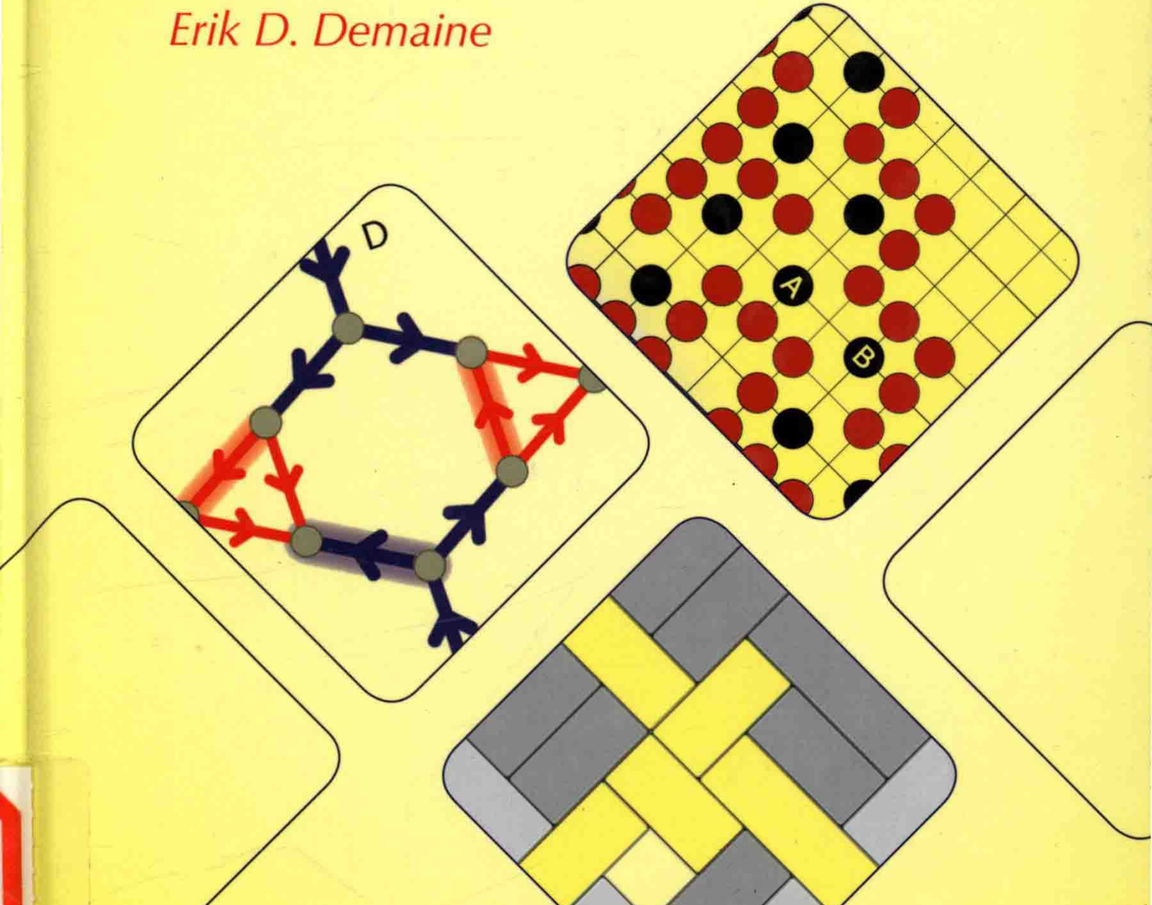
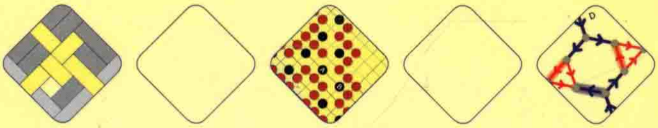
Games, Puzzles, & Computation

Robert A. Hearn
Erik D. Demaine



Games, Puzzles, & Computation

Robert A. Hearn · Erik D. Demaine



The authors show that there are underlying mathematical reasons that games and puzzles are challenging (which perhaps explain why they are so much fun). Complementarily, they also show that games and puzzles can serve as powerful models of computation—quite different from the usual models of automata and circuits—offering a new way of thinking about computation.

The first part of the book describes a simple, yet powerful, framework that the authors have developed for studying the connections between games, puzzles, and computation, called *constraint logic*. This framework is then applied to several real games and puzzles that people play, showing in each case that the game is computationally as difficult as any other game in its category. Finally, the appendices provide a substantial survey of all known results in the field of game complexity, serving as a reference guide for readers interested in the computational complexity of particular games, or interested in open problems about such complexities.

Advance Praise for *Games, Puzzles, and Computation*

"In this beautifully composed and fascinating volume, Hearn and Demaine make a compelling case that theirs is the right way to determine the computational complexity of a game. Highly recommended to those who like complexity theory or who wonder about the difficulty of playing games. If you're in both categories, tell your friends you'll be out of reach for the next two days!"

–Peter Winkler, author of *Mathematical Mind-Benders* and *Mathematical Puzzles*

ISBN 978-1-56881-322-6

9 781568 813226

A K Peters, Ltd.

Hearn
Demaine

Games, Puzzles, & Computation

AK PETERS

Games, Puzzles, and Computation

Robert A. Hearn
Erik D. Demaine

A K Peters, Ltd.
Wellesley, Massachusetts

Editorial, Sales, and Customer Service Office

A K Peters, Ltd.
888 Worcester Street, Suite 230
Wellesley, MA 02482
www.akpeters.com

Library of Congress Cataloging-in-Publication Data

Hearn, Robert A.
Games, puzzles, and computation / Robert A. Hearn, Erik D. Demaine.
p. cm.
Includes bibliographical references and index.
ISBN 978-1-56881-322-6 (alk. paper)
1. Problem solving—Mathematical models. 2. Games—Mathematical models. 3. Logic, Symbolic and mathematical. I. Demaine, Erik D., 1981– II. Title.

QA63.H35 2009
510–dc22

2009002069

Cover images: See Figures 1.2(a), 10.11(b), and C.12(a).

Printed in India
13 12 11 10 09 10 9 8 7 6 5 4 3 2 1

Games, Puzzles, and Computation

Acknowledgments

I would like to thank a few of the very many people who contributed directly or indirectly to my part in the making of this book: Michael Albert, Cyril Banderier, Eric Baum, Jake Beal, Elwyn Berlekamp, John Conway, Martin Demaine, Gary Flake, Aviezri Fraenkel, Greg Frederickson, Ed Fredkin, Martin Gardner, Shafi Goldwasser, J. P. Grossman, Richard Guy, Charles Hearn, Lerma Hearn, Michael Hoffmann, Michael Kleber, Tom Knight, Charles Leiserson, Norm Margolus, Albert Meyer, Marvin Minsky, Chet Murthy, Richard Nowakowsi, Ed Pegg, Ivars Peterson, Tom Rodgers, Aaron Seigel, Michael Sipser, Gerald Jay Sussman, John Tromp, Patrick Winston, David Wolfe, and Warren Wood.

This book arose out of my thesis work at MIT. Thus, it would not have been possible without Erik Demaine, who first interested me in tackling the complexity of sliding-block puzzles, and through whose mentoring and collaboration that initial result led to a stream of related results and eventually this book.

Special thanks are due the staff of A K Peters, most particularly Charlotte Henderson, who displayed amazing patience as deadlines slipped and who offered many valuable suggestions and improvements.

My deepest thanks go to my wife Liz, who is the reason I was at MIT in the first place. Finally, Liz, I've made something of all that fooling around with games and puzzles!

—R.A.H.

I would like to thank, at a broader level, the people who influenced the whole body of research. Most people studying the mathematics of games

and puzzles, and the two of us in particular, were heavily influenced by Martin Gardner. His 25 years of *Scientific American* articles and dozens of books showed the world how these fields could be combined. His influence continues today through the Gathering for Gardner meetings, organized by Tom Rodgers, giving a meeting place for many enthusiasts of games and puzzles and of mathematics and computer science. Other key meeting places have been provided by the combinatorial games community, in particular Elwyn Berlekamp, Richard Nowakowski, and David Wolfe. For me this began with the Second Combinatorial Games Theory Workshop and Conference in 2000, whose proceedings led to the book *More Games of No Chance.* Next came the Dagstuhl Seminar on Algorithmic Combinatorial Game Theory in 2002, which I helped organize, that specifically brought together people who work on algorithms and people who work on combinatorial games. These early meetings played an important role in getting this research area off the ground.

It has been exciting to go on this particular adventure with Bob Hearn. We started working together on the complexity of games in 2001 when I arrived at MIT, and our collaboration has been productive. Bob has been excitedly pushing the frontiers of the interplay between games, puzzles, and computation ever since we discovered Nondeterministic Constraint Logic, and I am happy that the culminated research is now embodied as both his PhD thesis and this book.

Finally, I would like to thank my father, Martin Demaine, whose passion for life and learning in general, and for games and puzzles in particular, ultimately brought me here. We have been sharing and collaborating throughout my life, all the way to this research, and beyond.

—E.D.D.

Contents

1 Introduction

This book is about games people play and puzzles people solve, viewed from the perspective of computer science—in particular computational complexity. Over the years, we have found increasingly deep connections between games, puzzles, and computation. These connections are interesting to us from multiple perspectives. As game players and puzzle solvers, we find underlying mathematical reasons that games and puzzles are challenging, which perhaps explain why they are so much fun. As computer scientists, we find that games and puzzles serve as powerful models of computation, quite different from the usual models of automata and circuits, offering a new way of thinking about computation.

This book has three main parts, and different parts may be of interest to different readers.

Part I (Games in General) describes a framework we have developed for studying the connections between games, puzzles, and computation, called *constraint logic*. This framework defines one simple prototypical game/puzzle that can be interpreted in a variety of settings. We can vary the number of players: one-player puzzles, two-player games, multiplayer team games, or, at the other extreme, zero-player automata. We can also vary how many moves for which the game lasts, or whether the players can hide information (like cards) from each other. In each such category of games, we prove that the corresponding form of constraint logic is the computationally most difficult game in that category, making it a natural point of reference from the computer-science perspective. This part of the book is fairly technical, building a mathematical foundation for particular constraint logics and establishing their computational complexity. Readers

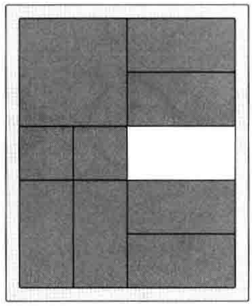


Figure 1.1. Dad's Puzzle, one of the earliest (c. 1909) and most popular sliding block puzzles [71]. The solver must slide the nine rectangular pieces within the 4×5 box to get the large square into the lower-left corner. The shortest solution takes a whopping 59 moves.

uninterested in the details, however, can simply read the summaries in the two short opening chapters, 2 and 3.

Part II (Games in Particular) applies the constraint-logic framework to real games and puzzles that people play. The approach is to take a real game or puzzle and show that it is computationally as hard as the corresponding form of constraint logic, making the real game/puzzle also computationally most difficult in its category. The intuition is that most "interesting" games are the most difficult in their class, so as a result we end up with many "equally difficult" games (when held up to the fairly course grain of computational complexity theory). What is interesting is that many real games and puzzles can be closely modeled within the constraint-logic framework, making it fairly easy to establish these complexity results.

Constraint logic started out as a tool for understanding the complexity of sliding-block puzzles, such as the puzzle shown in Figure 1.1. Our pursuit was motivated by a problem posed by Martin Gardner [71]: "These puzzles are very much in want of a theory. Short of trial and error, no one knows how to determine if a given state is obtainable from another given state. ..." The first application of the constraint-logic framework, which we will see in Section 9.3, shows that these puzzles have no such general theory, in a computational sense: no efficient procedure can tell whether a given state is obtainable from another, assuming standard beliefs in computational complexity. From there, the theory of constraint logic grew to increasing generality, capturing more and more types of real games and culminating in this book.

The third main part of this book, Appendix A (Survey of Games and Their Complexities), serves as a reference guide for readers interested in the computational complexity of particular games, or interested in open problems about such complexities. While Part II establishes the complexity of many games, it focuses on applications of the constraint-logic framework, and currently not all game-complexity results fit this framework. Appendix A surveys all known results, in addition to highlighting many open problems.

The rest of this introduction gives the reader some basic background on the two main concepts of this book—games/puzzles and complexity—followed by a more detailed overview of the constraint-logic framework.

1.1 What is a Game?

The term *game* means different things to different people in different fields. Our use intends to capture the kinds of games that people play, including board games like Chess, Checkers, and Go; card games like Poker and Bridge; one-player puzzles like Rush Hour, Peg Solitaire, and Sliding Blocks; and zero-player automata like John Conway's Game of Life.

Common to all of these games are four main features: positions, players, moves, and goals. Every game we consider has finitely many possible *positions*: board configurations, card distributions, piece arrangements, etc. In computer-science terminology, our games have a *bounded state*, a finite amount of information that defines the current situation. Some number of *players* manipulate the game position by individual *moves*. The players take turns in some order; the next player to move can be viewed as part of the game position. During each turn, the current player has a clear list of allowable moves (defined by the rules of the game) and picks one of them. The move transforms the game position into some other game position, in particular advancing to the next player in whatever order is determined by the game. Players may not be able to observe certain parts of the game position, allowing players to have *hidden states* such as cards in a hand, but this hidden state should not prevent a player from determining their allowable moves. Each player has a *goal*: to reach a game position with a particular property. The first player to reach their goal *wins*. We generally assume *optimal play*: players try to win as best they can given the available information. Although we do not directly consider games with randomness such as dice rolls in this book, we can model such phenomena by supposing that one player plays randomly instead of following optimal play (as in [131]).

This informal definition is related to several types of games studied in a variety of fields. To provide some context for our study of games, we summarize the related results and differences in these fields.

Combinatorial Game Theory. One closely aligned study of games is *combinatorial game theory*, as in the two classic books *Winning Ways* [8] and *On Numbers and Games* [27]; see also the more recent introduction *Lessons in Play* [3] and the research collections *Games of No Chance* I–III [128–130]. The bulk of this study considers two-player games of *perfect information*, where every player knows the entire state of the game and the moves available to each player—no hidden cards, random dice rolls, etc. Combinatorial game theory builds a beautiful theory of such perfect-information two-player games, revealing a rich mathematical structure. Perhaps most surprising is the connection to number systems: real numbers are special cases of Conway's "surreal numbers," which in turn are special cases of games, and basic addition carries over to the general case of games.

Perfect information has the attractive consequence that, in principle, a player could determine the optimal move to make by simulating the entire game execution, trying every possible move by each player (assuming the game is finite). Algorithmic combinatorial game theory aims to understand when there are better strategies than such brute force, and combinatorial game theory builds a useful collection of tools for understanding such optimal strategies in games. In many if not most interesting games, however, optimal game play is a difficult computational problem, and proving such results is our purpose in studying the complexity of games.

Economic Game Theory. A less related study of games is *(economic) game theory*, as pioneered by the work of John von Neumann [169] and John Nash [127]. Here, two or more selfish players participate in an economic event (game), often framed as a single round in which each player simultaneously chooses a strategy (or, often, a probability distribution of strategies), and the score (outcome) for each player is a given function of these strategies. In this context, there is generally no clear optimum strategy, either globally or for each player. There is, however, a clear set of optimal strategies for one player when given the strategies of other players, and if all players simultaneously follow such a strategy, the strategies are in *Nash equilibrium*. Nash [127] proved that all games have such an equilibrium, with the idea that players' strategies will eventually converge to one. On the other hand, theoretical computer scientists [23, 34] recently established that finding a Nash equilibrium is computationally intractable (formally, PPAD-complete), so players of normal computational power will in general take a long time to converge to a Nash equilibrium. More generally, economic game theory studies a wide variety of different notions of equilibria and the properties they possess.

The games we consider are both more specialized and more general than what is traditionally addressed by game theory: more specialized because we are concerned only with determining the winner of a game, and not