BRICS RS-07-1 D. Andersson: HIROIMONO is NP-complete

BRICS

Basic Research in Computer Science

HIROIMONO is NP-complete

Daniel Andersson

BRICS Report Series **RS-07-1**

ISSN 0909-0878 **January 2007**

See back inner page for a list of recent BRICS Report Series publications. Copies may be obtained by contacting:

BRICS
Department of Computer Science
University of Aarhus
IT-parken, Aabogade 34
DK–8200 Aarhus N
Denmark
Telephone: +45 8942 9300
Telefax: +45 8942 5601
Internet: BRICS@brics.dk

BRICS publications are in general accessible through the World Wide Web and anonymous FTP through these URLs:

`http://www.brics.dk`
`ftp://ftp.brics.dk`
This document in subdirectory `RS/07/1/`

HIROIMONO is $\mathcal{NP}$-complete

Daniel Andersson*

January 2, 2007

Abstract

In a Hiroimono puzzle, one must collect a set of stones from a square grid, moving along grid lines, picking up stones as one encounters them, and changing direction only when one picks up a stone. We show that deciding the solvability of such puzzles is $\mathcal{NP}$-complete.

1 Introduction

Hiroimono (拾い物, "things picked up") is an ancient Japanese class of tour puzzles. In a Hiroimono puzzle, we are given a square grid with stones placed at some grid points, and our task is to move along the grid lines and collect all the stones, while respecting the following rules:

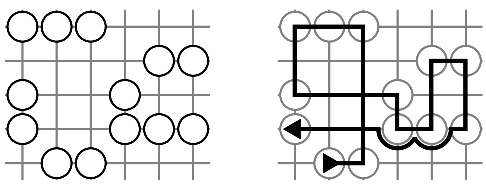
(1) We may start at any stone.

(2) When a stone is encountered, we must pick it up.

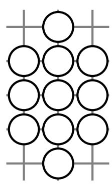
(3) We may change direction only when we pick up a stone.

(4) We may not make 180° turns.

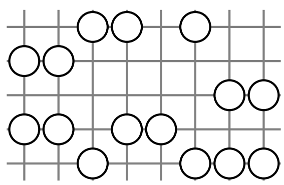
Example 1.



A puzzle and a way to solve it.



Unsolvable.



Exercise.

Although it is more than half a millennium old, Hiroimono, also known as Goishi Hiroi (碁石ひろい), appears in magazines, newspapers, and the World Puzzle Championship. Many other popular games and puzzles have been studied from a complexity-theoretic point of view and proved to give rise to hard computational problems, e.g. Tetris [3], Minesweeper [5], Sokoban [2], and Sudoku (also known as Number Place) [6]. We will show that this is also the case for Hiroimono.

Definition 1. HIROIMONO *is the problem of deciding for a given nonempty list of distinct points in* $\mathbb{Z}^2$ *representing a set of stones on the Cartesian grid, whether the corresponding Hiroimono puzzle is solvable under rules (1–4). The definition of* START-HIROIMONO *is the same, except that it replaces (1) with a rule stating that we must start at the first stone in the given list. Finally,* 180-HIROIMONO *and* 180-START-HIROIMONO *are derived from* HIROIMONO *and* START-HIROIMONO*, respectively, by lifting rule (4).*

Theorem 1. HIROIMONO*,* START-HIROIMONO*,* 180-HIROIMONO*, and* 180-START-HIROIMONO *are* $\mathcal{NP}$*-complete.*

Their membership is obvious. To show their hardness, we will construct a reduction from 3-SAT [4].

*Department of Computer Science, University of Aarhus, koda@daimi.au.dk

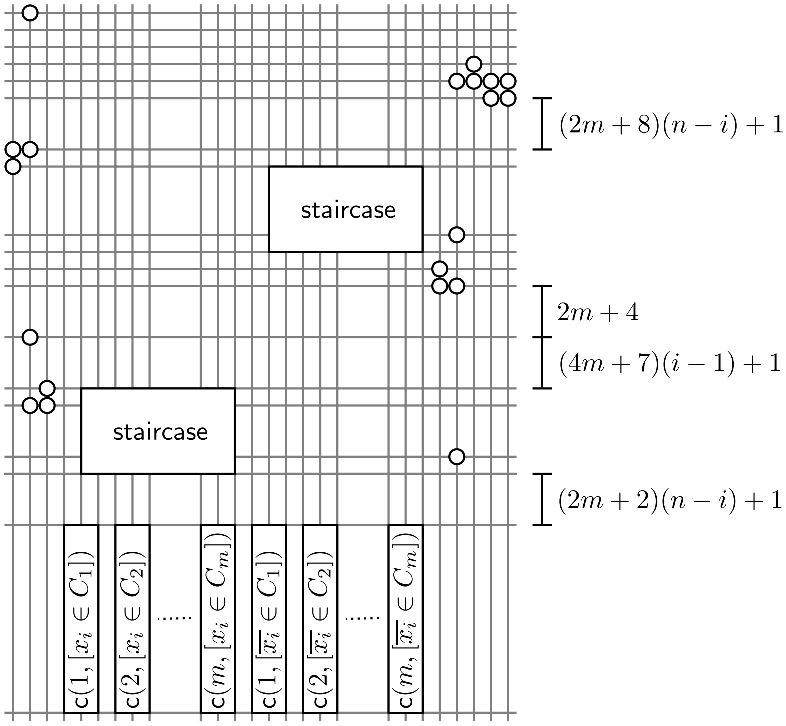
2 Reduction

Suppose that we are given as input a CNF formula $\phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ with variables $x_1, x_2, \ldots, x_n$ and with three literals in each clause. We output the puzzle p defined below.

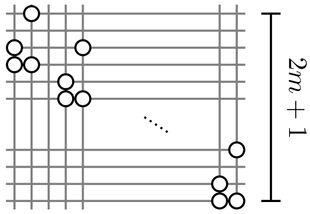
Remark. *Although formally, the problem instances are ordered lists of integer points, we will in our puzzle specifications leave out irrelevant details such as orientation, absolute position, and ordering after the first stone* $\odot$.

Definition 2.

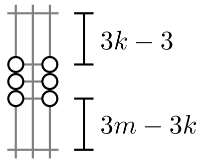
$\mathsf{choice}(i) :=$



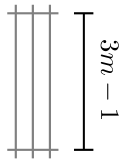
$\mathsf{staircase} :=$



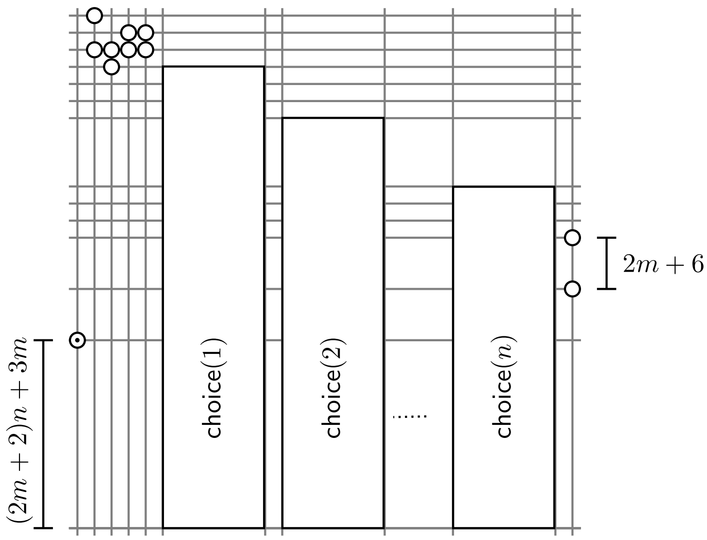
$\mathsf{c}(k, 1) :=$



$\mathsf{c}(k, 0) :=$



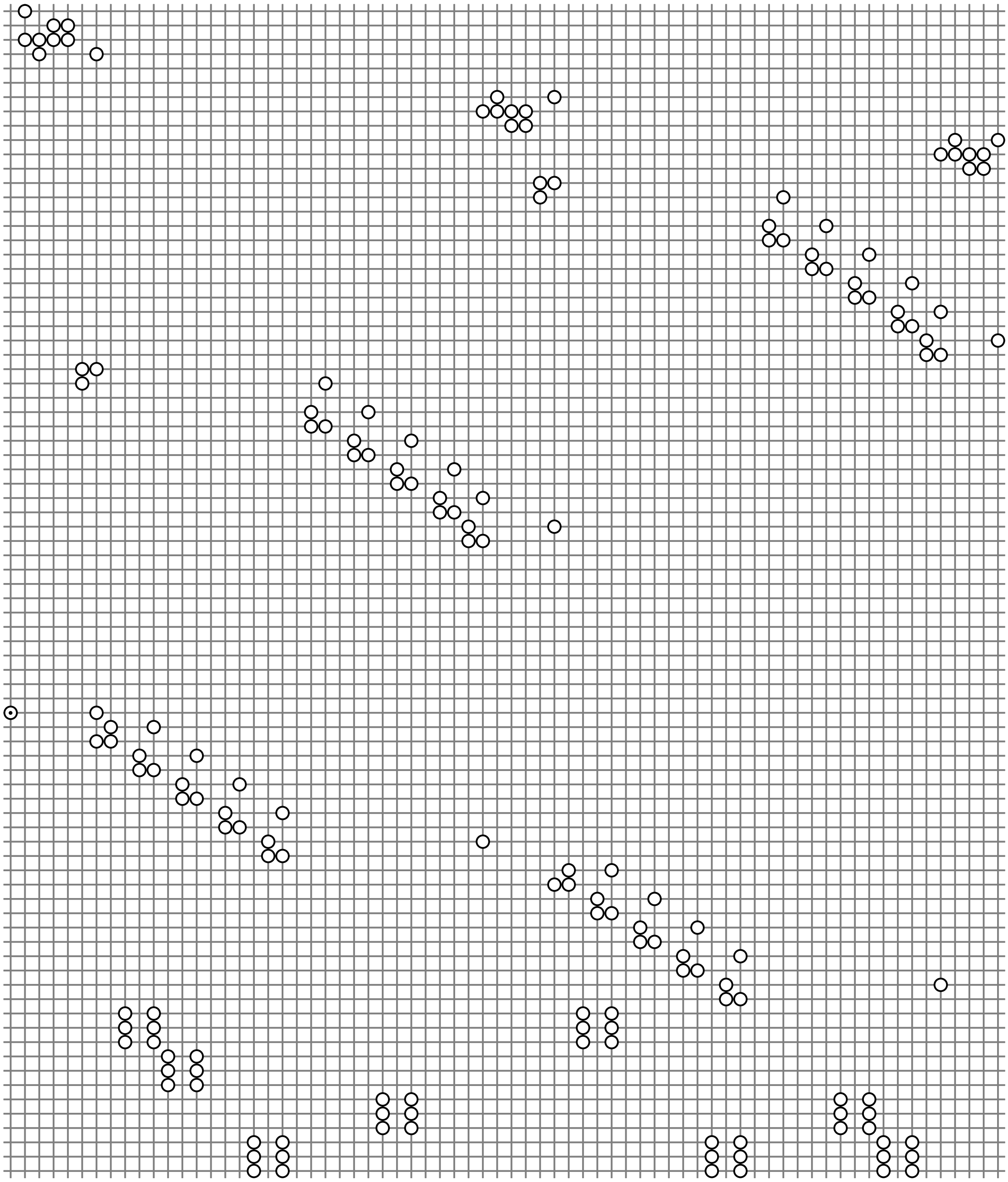
$\mathsf{p} :=$



Intuitively, the two staircase-components in choice(i) represent the possible truth values for x_i, and the c($k, 1$)-components, which are horizontally aligned, represent the clause C_k.

Clearly, we can construct p from ϕ in polynomial time.

Example 2. *If* $\phi = (x_1 \vee x_2 \vee x_2) \wedge (x_1 \vee x_1 \vee x_1) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_2}) \wedge (x_1 \vee x_2 \vee \overline{x_2})$, *then* p =



.

The implementation that generated this example is accessible online [1].

3 Correctness

From Definition 1, it follows that

$$\mathsf{START\text{-}HIROIMONO} \begin{array}{c} \subsetneq \\ \subsetneq \end{array} \begin{array}{c} \mathsf{HIROIMONO} \\ \mathsf{180\text{-}START\text{-}HIROIMONO} \end{array} \begin{array}{c} \subsetneq \\ \subsetneq \end{array} \mathsf{180\text{-}HIROIMONO}.$$

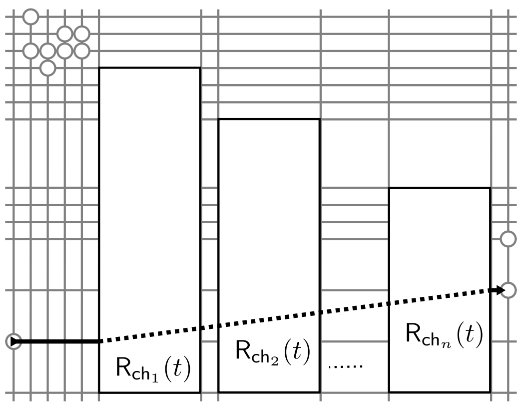
Thus, to prove that the map $\phi \mapsto \mathsf{p}$ from the previous section is indeed a correct reduction from 3-SAT to each of the four problems above, it suffices to show that $\phi \in$ 3-SAT $\Rightarrow \mathsf{p} \in$ START-HIROIMONO and $\mathsf{p} \in$ 180-HIROIMONO $\Rightarrow \phi \in$ 3-SAT.

3.1 Satisfiability implies solvability

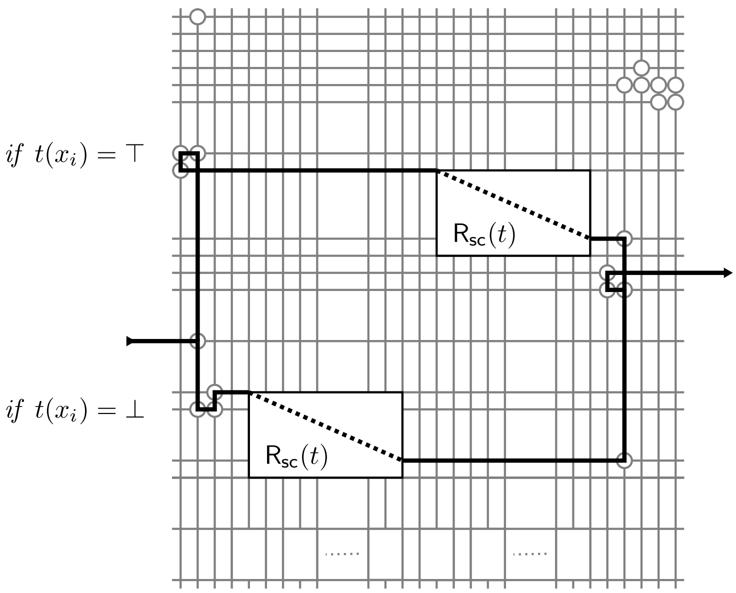
Suppose that ϕ has a satisfying truth assignment t^*. We will solve p in two stages. First, we start at the leftmost stone ⊙ and go to the lower rightmost stone along the path $\mathsf{R}(t^*)$, where we for any truth assignment t, define $\mathsf{R}(t)$ as follows:

Definition 3.

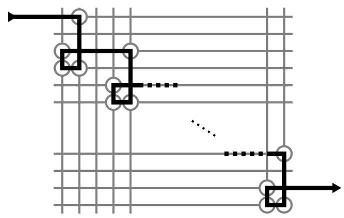
$\mathsf{R}(t) :=$



$\mathsf{R}_{\mathsf{ch}_i}(t) :=$



$\mathsf{R}_{\mathsf{sc}}(t) :=$



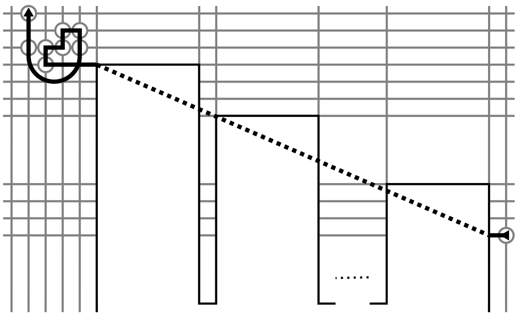
Definition 4. *Two stones on the same grid line are called* neighbors.

By the construction of p and R, we have the following:

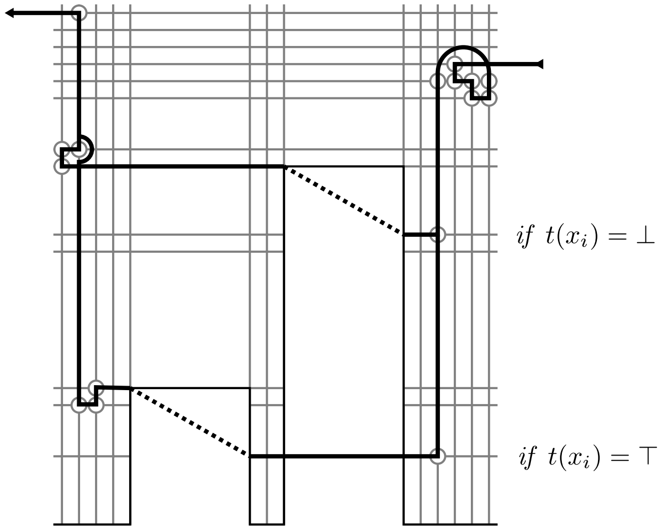
Lemma 1. *For any t and k, after $\mathsf{R}(t)$, there is a stone in a $\mathsf{c}(k,1)$-component with a neighbor in a* staircase*-component if and only if t satisfies C_k.*

In the second stage, we go back through the choice-components as follows:

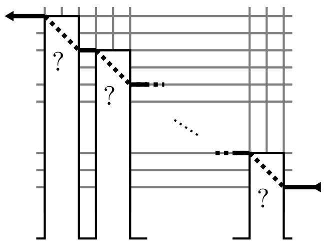
p



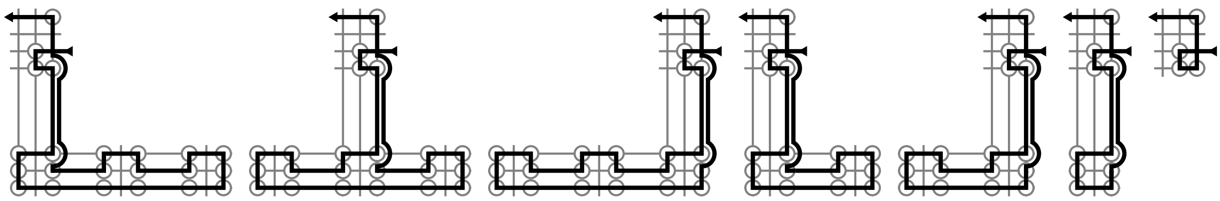
choice(i)



staircase

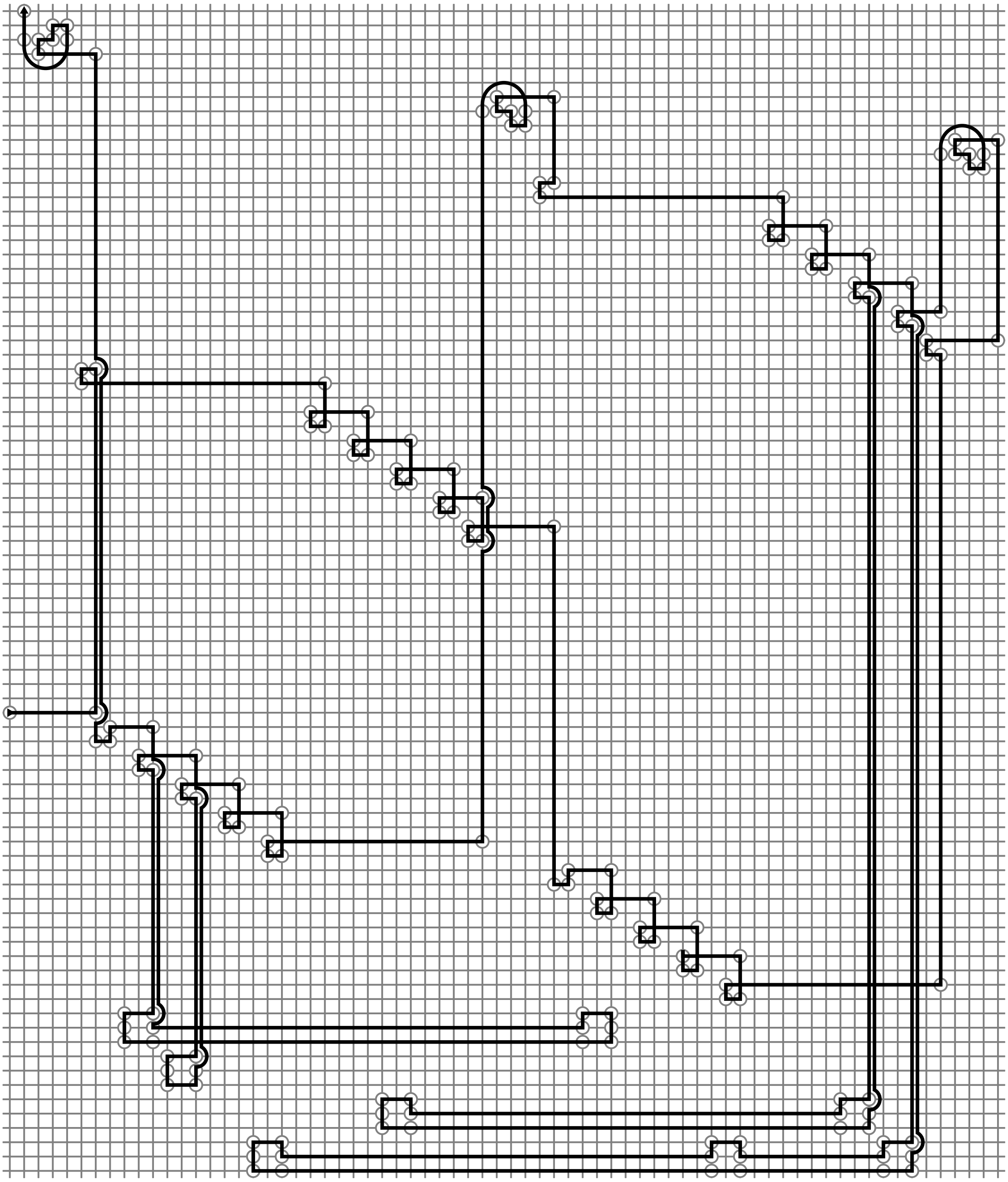


At each ?, we choose the first matching alternative of the seven following:



By Lemma 1, we will be able to “collect all the clauses”. Since this two-stage solution starts from the first stone ⊙ and does not make 180° turns, we have that $\mathsf{p} \in$ START-HIROIMONO.

Example 3. *A solution to Example 2.*



3.2 Solvability implies satisfiability

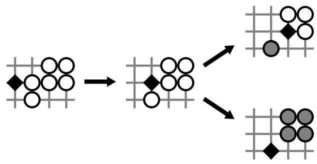
Suppose that $\mathsf{p} \in$ 180-HIROIMONO, and let s be any solution to p. We consider what happens as we solve p using s. Since the topmost stone and the leftmost stone each have only one neighbor, s must start at one of these and end at the other.

Definition 5. *A* situation *is a set of remaining stones and a current position. A* dead end D *is a nonempty subset of the remaining stones such that:*

- *There is at most one remaining stone outside of* D *that has a neighbor in* D.
- *No stone in* D *is on the same grid line as the current position.*

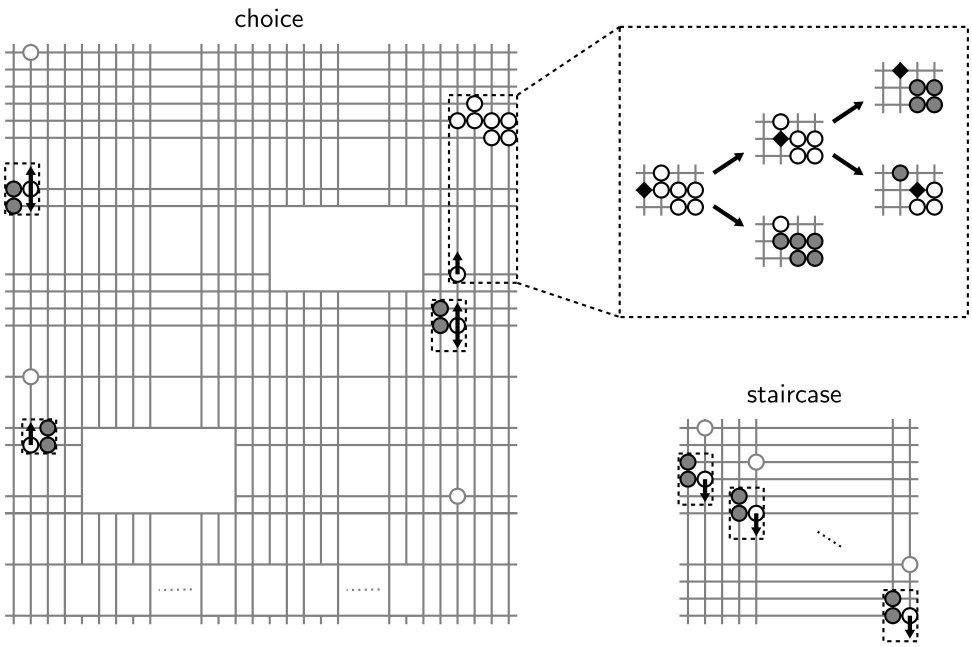
A hopeless situation *is one with two disjoint dead ends.*

Clearly, s cannot create hopeless situations. However, if we start at the topmost stone, then we will after collecting at most four stones find ourselves in a hopeless situation, as is illustrated by the following figure, where ◆ denotes the current position and ● denotes a stone in a dead end.



Thus, s must start at the leftmost stone and end at the topmost one.

We claim that there is an assignment t^* such that s starts with $\mathsf{R}(t^*)$. The following figure shows all the ways that we might attempt to deviate from the set of R-paths and the dead ends that would arise.



By Lemma 1, we have that if t^* from above fails to satisfy some clause C_k, then after $\mathsf{R}(t^*)$, the stones in the $\mathsf{c}(k,1)$-components will together form a dead end. This cannot happen, so t^* satisfies ϕ.

4 Acknowledgements

I thank Kristoffer Arnsfelt Hansen, who introduced me to Hiroimono and suggested the investigation of its complexity, and my advisor, Peter Bro Miltersen.

References

[1] D. Andersson. "Reduce 3-SAT to HIROIMONO." `http://purl.org/net/koda/sat2hiroi.php`.

[2] J. Culberson. "Sokoban is PSPACE-complete." In E. Lodi and L. Pagli, eds., *Proceedings of the International Conference on Fun with Algorithms (FUN '98)*, pages 65–76. Carleton Scientific, 1998.

[3] E. D. Demaine, S. Hohenberger, and D. Liben-Nowell. “Tetris is hard, even to approximate.” In T. Warnow and B. Zhu, eds., *Proceedings of the 9th Annual International Conference on Computing and Combinatorics (COCOON ’03)*, pages 351–363. Springer-Verlag, 2003.

[4] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1979.

[5] R. Kaye. “Minesweeper is NP-complete.” *Mathematical Intelligencer*, **22**(2):9–15, 2000.

[6] T. Yato and T. Seta. “Complexity and completeness of finding another solution and its application to puzzles.” *Information Processing Society of Japan SIG Notes*, 2002-AL-87-2, 2002.

Recent BRICS Report Series Publications

RS-07-1 **Daniel Andersson.** ***HIROIMONO is NP-complete*****. January 2007. 8 pp.**

RS-06-19 **Michael David Pedersen.** ***Logics for The Applied π Calculus*****. December 2006. viii+111 pp.**

RS-06-18 **Małgorzata Biernacka and Olivier Danvy.** ***A Syntactic Correspondence between Context-Sensitive Calculi and Abstract Machines*****. dec 2006. iii+39 pp. Extended version of an article to appear in TCS. Revised version of BRICS RS-05-22.**

RS-06-17 **Olivier Danvy and Kevin Millikin.** ***A Rational Deconstruction of Landin's J Operator*****. December 2006. ii+37 pp. Revised version of BRICS RS-06-4. A preliminary version appears in the proceedings of IFL 2005, LNCS 4015:55–73.**

RS-06-16 **Anders Møller.** ***Static Analysis for Event-Based XML Processing*****. October 2006. 16 pp.**

RS-06-15 **Dariusz Biernacki, Olivier Danvy, and Kevin Millikin.** ***A Dynamic Continuation-Passing Style for Dynamic Delimited Continuations*****. October 2006. ii+28 pp. Revised version of BRICS RS-05-16.**

RS-06-14 **Giorgio Delzanno, Javier Esparza, and Jiří Srba.** ***Monotonic Set-Extended Prefix Rewriting and Verification of Recursive Ping-Pong Protocols*****. July 2006. 31 pp. To appear in ATVA '06.**

RS-06-13 **Jiří Srba.** ***Visibly Pushdown Automata: From Language Equivalence to Simulation and Bisimulation*****. July 2006. 21 pp. To appear in CSL '06.**

RS-06-12 **Kristian Støvring.** ***Higher-Order Beta Matching with Solutions in Long Beta-Eta Normal Form*****. June 2006. 13 pp. To appear in** ***Nordic Journal of Computing*****, 2006.**

RS-06-11 **Kim G. Larsen, Ulrik Nyman, and Andrzej Wasowski.** ***An Interface Theory for Input/Output Automata*****. June 2006. 40 pp. Appears in Misra, Nipkow and Sekerinski, editors,** ***Formal Methods: 14th International Symposium*****, FM '06 Proceedings, LNCS 4085, 2006, pages 82–97.**

RS-06-10 **Christian Kirkegaard and Anders Møller.** ***Static Analysis for Java Servlets and JSP*****. June 2006. 23 pp. Full version of paper presented at SAS '06.**