BRICS

Basic Research in Computer Science

BRICS RS-00-10 Dantchev & Riis: A Tough Nut for Tree Resolution

A Tough Nut for Tree Resolution

Stefan Dantchev
Søren Riis

BRICS Report Series RS-00-10

ISSN 0909-0878 May 2000

BRICS
Department of Computer Science
University of Aarhus
Ny Munkegade, building 540
DK–8000 Aarhus C
Denmark
Telephone: +45 8942 3360
Telefax: +45 8942 3255
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/00/10/`

A tough nut for tree resolution

May 2000

Stefan Dantchev, Søren Riis

BRICS*

Department of Computer Science
University of Aarhus
Ny Munkegade, bldg. 540
DK-8000 Århus C
Denmark

Abstract

One of the earliest proposed hard problems for theorem provers is a propositional version of the *Mutilated Chessboard problem*. It is well known from recreational mathematics: Given a chessboard having two diagonally opposite squares removed, prove that it cannot be covered with dominoes. In Proof Complexity, we consider not ordinary, but $2n \times 2n$ mutilated chessboard. In the paper, we show a $2^{\Omega(n)}$ lower bound for *tree resolution*.

1 Introduction

The most well-studied tautologies, that are hard for resolution, are those created by translating matching principles in certain graphs into propositional formulas - [1], [5], [11]. It was Haken who first proved in [4] an exponential lower bound is proven for the pigeon-hole principle PHP_n^{n+1} stating that there is no perfect matching in the bipartite graph $K_{n+1,n}$. After that, the result has been simplified and improved ([2], [3], [5], [11],) as well as lower bounds have been proven for other matching principles [2]. In all these proves counting arguments have been used.

One of the problems on which these techniques have failed so far is the mutilated chessboard problem. It also has the distinction to be the earliest proposed hard problem for theorem provers [7]. The problem itself is: given a $2n \times 2n$ chessboard with two diagonally opposite squares missing, prove that it cannot

*Basic Research in Computer Science,
Centre of the Danish National Research Foundation.

be covered with dominoes. We can consider it as a matching problem: Squares are vertices of the graph, and there is an edge between every two neighboring squares. Thus one component of the bipartite graph consists of black squares and the other consists of white ones. Two missing squares are of the same color which implies one of the components in the graph has two more vertices than the other. That is why there is no perfect matching i.e. dominoes tiling of the mutilated chessboard.

In this paper, we prove an exponential lower bound for *tree* resolution proofs of the problem. Our technique is an adversary argument. We consider any tree resolution proof as a Prover-Adversary game. This kind of games is first introduced in [9] and [8] where Haken's proof is presented in this setting.

2 The Problem

In this section, we first remind what the resolution proof system is, and introduce the formal description of the Mutilated Chessboard problem, i.e. its encoding as a set of clauses.

2.1 Tree resolution

We first give some definitions. A *literal* is either a propositional variable or the negation of propositional variable. A *clause* is a set of literals. It is satisfied by a truth assignment if at least one of its literals is true under this assignment. A set of clauses is *satisfiable* if there exists a truth assignment satisfying all the clauses.

Resolution is a proof system designed to *refute* given set of clauses i.e. to prove that it is unsatisfiable. This is done by means of the resolution rule

$$\frac{C_1 \bigcup \{v\} \quad C_1 \bigcup \{\neg v\}}{C_1 \bigcup C_2}.$$

We derive a new clause from two clauses that contain a variable and its negation respectively. The goal is to derive the empty clause from the initial ones. Anywhere we say we *prove* some proposition, we mean that first its negation in a clausal form and then resolution is used to refute these clauses.

There is an obvious way to represent every resolution proof as a directed acyclic graph whose nodes are labelled by clauses. If we restrict the corresponding graph to be a tree, we obtain *tree* resolution. If this restriction is not present, we speak about general or *dag-like* resolution.

In our proof, we use the following simple fact (see, e.g. [6]): *Any tree resolution refutation* of a set of clauses can be considered as a *decision tree* that solves the corresponding search problem. *The search problem* for an unsatisfiable set of clauses is: given an arbitrary truth assignment of variables, find a clause which is falsified by it.

2.2 Mutilated Chessboard

We consider a $2n \times 2n$ chessboard, with two diagonally opposite white corners removed as shown on fig. 1.

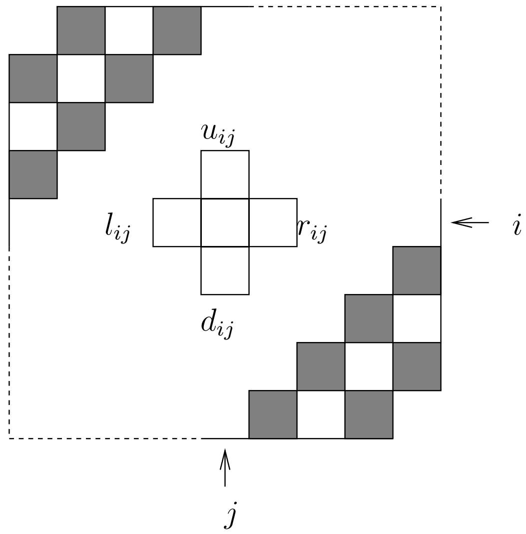


Figure 1:

We can encode the problem in clausal form as follows. We introduce (at most) four propositional variables $u_{ij}, r_{ij}, d_{ij}, l_{ij}$ for every square (i,j) as it is shown on fig. 1. The set of clauses consists of

1. $\{u_{ij}, r_{ij}, d_{ij}, l_{ij}\}$
2. $\{\neg u_{ij}, \neg r_{ij}\}$, $\{\neg u_{ij}, \neg d_{ij}\}$, $\{\neg u_{ij}, \neg l_{ij}\}$, $\{\neg r_{ij}, \neg d_{ij}\}$, $\{\neg r_{ij}, \neg l_{ij}\}$, $\{\neg d_{ij}, \neg l_{ij}\}$
3. $u_{ij} \leftrightarrow d_{i-1\,j}$, $r_{ij} \leftrightarrow l_{i\,,j+1}$, $d_{ij} \leftrightarrow u_{i+1\,j}$, $l_{ij} \leftrightarrow r_{i\,j-1}$

First two lines claim the square is covered exactly once. Third one connects it with all its neighbors. It is not written in as a set of clauses, but in fact it is as the propositional formula $a \leftrightarrow b$ is equivalent to the clauses $\{a, \neg b\}$ and $\{\neg a, b\}$. Of course, not all of these variables are well defined for the border squares (for instance $r_{i\,2n}$ does not exist as it would require the existence of $r_{i\,2n} \leftrightarrow l_{i\,2n+1}$). In those cases, we simply omit the corresponding variables and clauses.

Note that our encoding is somehow redundant. We could have associated a propositional variable to every neighboring squares, and we would have had as half as many variables. However, the presented encoding makes makes the proof technically easier as we will see later on. In particular, we can now think of any propositional variable as a pair of a square and a domino that covers it. Thus any query in the decision tree is of the form "Is it true that the domino covering the square (i,j) goes up/right/down/left?".

3 Lower bound

This section is organized as follows. First, the outline of the proof is given in 3.1. The proof itself is given in terms of a Prover-Adversary Game. It consists of two main lemmas. First of them is presented in 3.2. For the sake of simplicity a new game is introduced there. We call it Road Game, and it might be of independent interest. The second lemma is proven in 3.3 where the original problem is reduced to the Road Game.

3.1 Prover-Adversary Game

The main idea is to consider a tree resolution proof of the mutilated chessboard problem as a *Prover-Adversary Game*. This concept is introduced by Pudlák and Buss in [9]. In [8], the classical result of Haken [4] is presented in terms of such a game.

Let us define the game that corresponds to our problem precisely. As usual there are two players - *Prover* and *Adversary*. Adversary pretends that there is a complete tiling of the mutilated chessboard by dominoes. Prover tries to convict him of lying. She holds a *decision tree* that solves the *search problem*, and asks questions following the tree. As the mutilated chessboard is finite, Prover always wins meaning that he finds a contradiction in the (partial) assignment built by Adversary's answers. Thus the goal of Adversary is to maximize the size of the tree Prover needs in order to win. Adversary's strategy is based on the concept of *critical questions* first introduced in [10]. It has two important properties that correspond to the two main lemmas in our proof:

1. Prover must ask at least cn critical questions (for some constant c) in order to win.
2. Every time Adversary is asked a critical question, he has the freedom to choose between "yes" and "no" answers.

Roughly speaking, the first lemma claims that there is at least one long branch in the decision tree. The second one shows that this branch, in fact, blows up to an exponentially big subtree. This implies that the size of Prover's decision tree is at least 2^{cn}.

3.2 Road Game

This game is played on a $m \times m$ chessboard. Adversary claims that there exists an *infinite acyclic road* starting from the the bottommost right square of the board. At any round Prover chooses a squares and asks "Is this square on the road?" Adversary answers either "yes" or "no". In the former case, he should construct the entire path from the bottommost right square to the one he has just been asked about. As the chessboard is finite, Prover can always win after having asked m^2 questions. Thus the goal of Adversary is to survive as many rounds as possible. We will prove that he can answer consistently $\Omega(m)$ questions no matter what strategy Prover uses.

Let us now present the Road Game more formally. We have four kinds of tiles as shown on fig. 2. The tile a corresponds to the "no" answer, while b and c

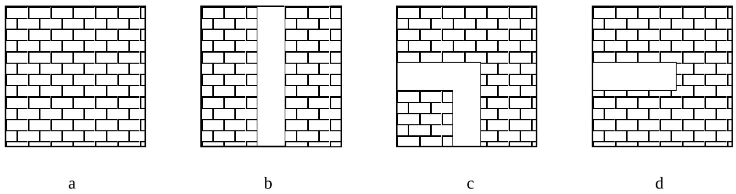


Figure 2: Tiles for the new game

(and all their rotations) correspond to "yes" answers. The tile d is a special one. It is used only once before the game has even started. We put it as shown on fig. 3 in order to emphasize the starting point of the road. It is now clear how the

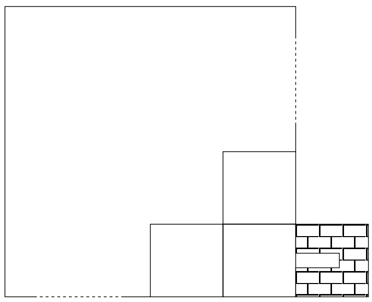


Figure 3: The board for the new game

game is played. When asked about a particular square, Adversary puts there either the tile a (answer "no") or b/c rotated appropriately (answer "yes"). In the latter case, he uses these to construct the entire path from the bottommost right square. It is also clear how Prover can observe any inconsistency in the answers and thus win.

We can now explain Adversary's strategy. To do this, we need to introduce some more concepts. We call squares that Prover has asked about *marked.* We should note that there might be squares neither marked nor empty, namely these used in constructing the entire road after every positive answer. Another concept is the *current end of the road.* Initially, it coincides with the beginning of the road, i.e. it is the bottommost right square. After every answer "yes" Adversary has to enlarge the road from its current end to the square having just been marked. After that, the road is directed to some neighboring square of the last one. If this square is empty, it becomes the new current end of the road. Otherwise, a contradiction is found as the road cannot be enlarged anymore, and therefore Prover wins.

At any stage in the game, we consider the *connected components* of empty squares, initially the entire board being such a component. We call the unique component, which the current end of the road belongs to, *bad* and the others - *good*. We should note that Prover does not need to ask in the good components as Adversary can safely answer “no” to all such questions. Therefore we can assume that all Prover’s questions are inside the current bad component.

The Adversary’s strategy is now both natural and simple: As far as Prover’s questions do not disconnect the bad component the answers are “no”. Otherwise, Adversary looks at the sizes of the new components obtained by disconnecting the bad one. The largest one will be the new bad component. Therefore, if the current end of the road happens to be already there, Adversary answers “no”. If not, he answers “yes” using the appropriate tile to direct the road to the new bad component.

We can now state and prove the first main lemma.

Lemma 1 *Prover has to ask at least $\frac{1}{2\sqrt{2}}m$ questions in order to win the Road Game.*

Proof First of all, we need to observe that the “border” of every component, either bad or good, consists of marked squares and (parts of) the sides of the chessboard itself. We can prove a simple isoperimetric inequality.

Proposition 1 *Let A be a connected component consisting of s empty squares. Let us also suppose that A touches at most two neighboring sides of the chessboard. The number of marked squares needed to isolate A from the rest of the board is at least $\sqrt{s}$.*

Proof (of the proposition) W.l.o.g. let us suppose that A together with its “border” of marked squares is contained in an $a \times b$ rectangle where $a \geq b$. Obviously, the number of marked squares has to be at least a - at least one in every row of the rectangle. Then

$$a^2 \geq ab \geq s.$$

□

Note that this proposition does not hold if A touches three of the sides of the chessboard. As an example, we can take small number of squares connecting two neighboring sides of the board, near to one of the corners. They divide it into two connected areas, one of them being much smaller than the other. It is now clear that the proposition does not hold for the bigger component.

We can now prove the lemma. Let us consider the first square such that after having marked it the size of the bad component gets less or equal to $\frac{m^2}{2}$. Two cases are possible

- The square has not disconnected the bad component. Then its size is exactly $\frac{m^2}{2}$.

- The square has disconnected the bad component. At most four new connected components could have appeared and the new bad is the largest among them. Therefore its size is at least $\frac{m^2}{8}$.

In both cases, we have that the size of the bad component is between $\frac{m^2}{8}$ and $\frac{m^2}{2}$. Now, let us consider all possible shapes of its border:

1. It consists of marked squares and at most two neighboring sides of the chessboard. The proposition applies, so that the number of marked squares is at least $\frac{m}{2\sqrt{2}}$.
2. It consists of some marked squares and either two opposite sides of the board or three sides of the board. In these cases, there must be a connected "path" of marked squares connecting two opposite sides of the chessboard. Every such a path contains at least m squares.
3. It consists of some marked squares and all four sides of the board. In this case, we consider the connected components of all good parts of the board (i.e. we join all such parts that have common borders). The lemma applies to them, so that if their sizes are $s_1, s_2, \dots s_k$ we have to have at least $\sum_{j=1}^{k} \sqrt{s_j}$ marked squares. Then

$$\left(\sum_{j=1}^{k} \sqrt{s_j} \right)^2 \geq \sum_{j=1}^{k} s_j \geq \frac{m^2}{2},$$

so that we have at least $\frac{m}{\sqrt{2}}$ marked squares.

In all the cases, the number of marked squares is at least $\frac{m}{2\sqrt{2}}$.$\square$

3.3 Reduction

In this subsection, we will show how to reduce the original problem to the Road Game. The general idea is as follows. We divide the mutilated chessboard into non-overlapping *constant-size* squares that we will further call *zones*. Every big enough constant is proper for our proof. However, we use 26×26 squares, although much smaller constant is most likely enough. We also "move" one of the missing squares near to the other as shown on fig. 4a for a $(26n+2) \times (26n+2)$ chessboard. At the end, we have only one *bad* zone, namely the bottommost right one which contains two missing white squares. All other zones are *good*, initially complete 26×26 squares.

We can now introduce the most important concept in our proof - *critical question*. Informally speaking, the (unique) critical question for a particular zone is the first question inside the zone and such that both "yes" and "no" answers do not contradict to any previous answers. After having answered a such a question, Adversary decides a complete tiling for the corresponding zone, so that all future question about it have forced answers. When Prover asks a

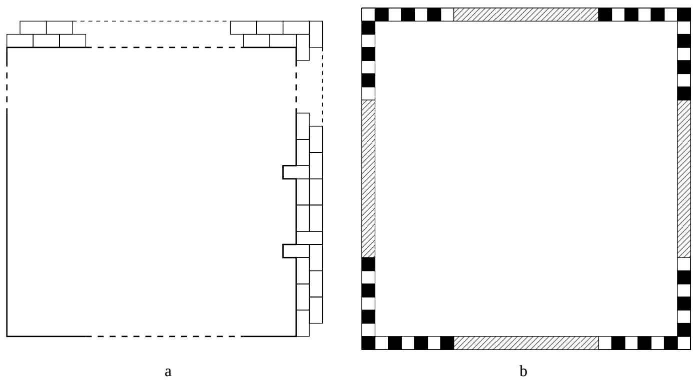


Figure 4:

critical question for a zone, she receives not only its answer, but also a complete tiling of the zone for free. Obviously, in a particular Prover-Adversary game, the sequence of the critical questions depends completely on Prover's strategy, that is the decision tree she holds.

We can define critical question formally as follows. Let us fist remind that any question is a pair of a square and a tile. The critical question for a particular zone is the first one such that

1. The square is inside the zone.
2. Either the tile is completely inside or the square is one of the dashed squares on fig. 4b.
3. Both answers "yes" and "no" do not contradict to the current partial tiling.

Conditions 1 and 3 correspond to the intuitive explanation we have already given. The second says that a critical question affecting the a neighbouring empty zone should not be too near to the corners. This prevents the situation when two critical questions "cut" the corner of an empty zone, so that an immediate contradiction is found.

Now, we can describe the reduction in details. We first explain the shape of zones we can have during the game. There is only one bad zone that corresponds to the current end of the road in the Road Game. All other zones are either tiled or good. The good zones correspond to the empty squares in the Road Game. We show how any zone can look like on fig.5 . The possible patterns shown on fig. 5b. The squares bordered with dashed lines are missing, i.e. they are covered by the tiling of the corresponding neighbouring zone. These patterns can appear on dashed area shown on fig. 5a. A good zone can have the pattern 1 only (two missing neighbouring squares) there. The bad zone should have either 2 or 3 on one of its sides. On the others, it can have the pattern 1. Patterns 4 and 5 do not appear anywhere during the game. They are shown because we use them in our proof.

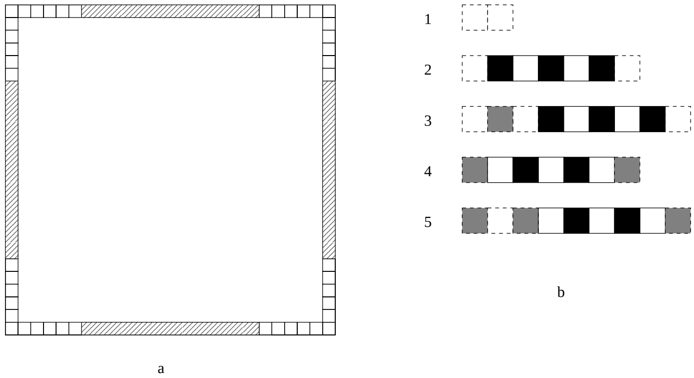


Figure 5:

We can now prove that the described above invariants can be satisfied while playing the Road Game. We will also show that when asked a critical question, Adversary can give both answers without affecting its strategy. To do this, we consider all the possible situations when Prover asks a critical question

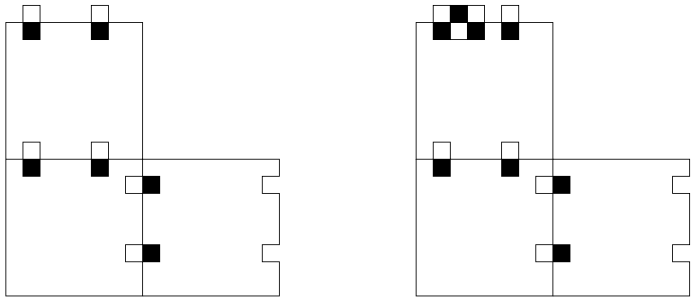
1. The answer is “no”, i.e. this zone is not on the road, in the Road Game. There are two possibilities for the question:

   (a) The tile is completely inside the zone. Both “yes” and “no” answers can be realized with the tile lying inside the zone (“no” answer is realized by putting the tile in any position covering the square, but different than the one in the question). What remains to prove is that a good zone with a domino put in any position inside it can be tiled without affecting the neighbouring zones. We give the proof in the appendix.

   (b) The tile goes outside of the zone. In this case, the “yes” answer affects a neighbouring zone, since it cuts a square from it. Therefore we need to cut another square, of different colour. Thus the pattern 1 from fig. 5b appears on the side of the neighbouring zone. The answer “no” can be realized by a tile which is completely inside the current zone.

2. The zone is on the road in the Road Game. We need to “move” the two missing white squares from the current bad zone to the new one. We move them along the road connecting these zones as shown on fig. 6. There are several possibilities depending on the tile of the critical question.

   (a) It is completely inside the current zone. In both cases, corresponding to “yes” and “no” answers, we are free to choose the two pairs of dominoes that affects the current zone (see fig. 6a). It remains to prove that a zone, having patterns 2 and 4 on two sides and possibly



a b

Figure 6:

1 on the other two and a domino put in any position inside it, can be tiled completely by dominoes. Again, we refer to the appendix for the proof

(b) The critical question is one of the two upper tiles on fig. 6a, i.e. it cuts one white square from the new bad zone. The answer "yes" forces us to use another such a tile to cut another white square. If the answer is no, we should choose another pair of tiles, as this one is forbidden by this answer. We can do that because we have chosen big enough zones, so that we have many enough such pairs (we need at least two). An analogous case is when the critical question cuts a black square from the previous zone on the road.

(c) The critical question cuts one black square from the new bad zone. The answer "yes" forces us to cut three white square from it as shown on fig. 6b. Thus the pattern **3** appears on the border of the bad zone. An analogous case is when the critical question cuts a white square from the previous zone on the road.

Summarizing all the above cases, what we need to prove, in order to compete the proof, are the following two propositions

Proposition 2 *Let G be a good zone, i.e. with the pattern 1 on all its sides, with two neighbouring squares missing. Then G can be tiled completely by dominoes .*

Proposition 3 *Let B be a zone, having either patterns 2 or 3 and either 4 or 5 on two of its sides and the pattern 1 on the other two sides, with two neighbouring squares missing. Then B can be tiled completely by dominoes .*

The proofs are given in the appendix.

References

[1] P. Beame and T. Pitassi. Propositional proof complexity: past, present and future. Technical Report 67, ECCC, 1998.

[2] E. Ben-Sasson and A. Wigderson. Short proofs are narrow - resolution made simple. Technical Report 22, ECCC, 1999.

[3] S.R. Buss and G. Turán. Resolution proofs of generalized pigeonhole principles. *Theoretical Computer Science*, 62:311–317, 1988.

[4] A. Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297–308, 1985.

[5] J. Krajíĉek. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*. Cambridge University Press, 1995.

[6] László Lovász, Moni Naor, Ilan Newman, and Avi Wigderson. Search problems in the decision tree model. *SIAM Journal on Discrete Mathematics*, 8(1):119–132, 1995.

[7] J. McCarthy. A tough nut for proof procdures. Stanford Artifical Intelligence Project Memo 16, July 1964.

[8] P. Pudlák. Proofs as games. *American Mathematical Monthly*, to appear.

[9] P. Pudlák and S.R. Buss. How to lie without being (easily) convicted and the lengths of proofs in propositional calculus. In Pacholski and Tiuryn, editors, *Computer Science Logic'94*, volume 993 of *LNCS*, pages 151–162. Springer-Verlag, 1995.

[10] Søren Riis. A complexity gap for tree-resolution. Technical Report RS-99-29, BRICS, September 1999.

[11] A. Urquhart. Resolution proofs of matching principles. 1998.

Appendix

We prove one of the two possible cases of the proposition 3. The other is analogous, and the proposition 2 is even easier.

More precisely, we prove that a zone, having pattern 2 on its bottom side, pattern 5 on its top side and possibly pattern 1 on the other two sides, with a domino put in any position inside it, can be tiled. This case corresponds to the tile b of the Road Game (see fig. 2). The proof is both simple and tedious. The zone is shown on fig. 7a where missing squares corresponding to the patterns 2 and 5 are presented by thicker borders.

We construct two cycles that contain the missing squares. On the picture, they are denoted by $ABCDEFG$ and $HIJKLMN$. The conditions that any of these cycles has to satisfy are the following:

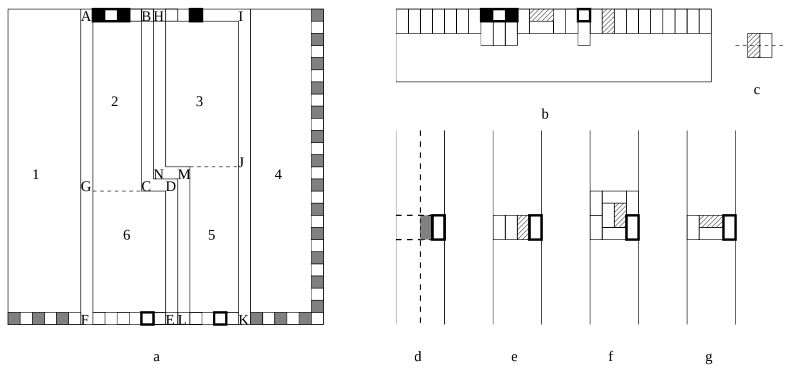


Figure 7:

- The distance between the longest vertical side (AGF) and the nearest side of the zone, parallel to it, is exactly 6.
- The distance between two shorter vertical sides (BC and DE) and the nearest missing squares, belonging to the cycle, is either 1 or 2.
- The middle horizontal side (CD), if exists (i.e. C and D do not coincide), should be not so near to any of the horizontal sides of the zone.

We will also use the following trivial assertion:

(*) A rectangle, having both sides greater or equal to two and at least one of them even, with a domino put at any position inside it can be tiled completely.

We can now prove the main proposition. Assume for a moment that we had not the extra tile inside the zone. In this case, we would tile the zone completely as follows: First we tile the two cycles. This is always possible because the missing squares disconnect any of them into two part of even length. After having done this, we observe that the empty part of the zone can be divided into six rectangles with at least one even side as shown on fig. 7a where they are numbered.

Now, let us put a domino at any position inside the zone. We consider all possible cases.

1. The tile does not intersect any of the two cycles and, moreover, it is not inside the rectangles 1 or 4. If it is completely inside any of the four rectangles we are done by (*). If the domino is on the border between $2-6$ or $3-5$, we first put one extra tile as shown on fig. 7c and then apply (*) for both affected rectangles.
2. The tile intersects either the segment AI or the segment EK. In this case, we put many extra dominoes, as shown on fig. 7b where the two possible

positions of the original domino are dashed. In this way, we reduce the problem from a 26×26 zone to a 24×26 zone in which no extra tile appears.

3. The tile intersects any other segment of the two cycles. Since we have chosen all the distances appropriately, we can move the affected segment(s) 2 squares left/right/up/down, still keeping the missing squares on the cycles and the six rectangles fulfilling the requirements of (*). After these moves, the tile does not intersect any of the cycles.

4. The last remaining case is when the tile is inside either 1 or 4. It is shown on fig. 7d-g where the tile from pattern 1 is shown with a thicker border, and the new tile is dashed. If this tile does not intersect the gray area on fig. 7d, we can use one of dashed lines to divide the rectangle into two new rectangles, satisfying the condition of (*) with one tile inside each of them. In the other case, we put some extra tiles as shown on fig. 7e-g, thus again dividing the rectangle into two new, satisfying (*).

Recent BRICS Report Series Publications

RS-00-10 **Stefan Dantchev and Søren Riis.** ***A Tough Nut for Tree Resolution*****. May 2000. 13 pp.**

RS-00-9 **Ulrich Kohlenbach.** ***Effective Uniform Bounds on the Krasnoselski-Mann Iteration*****. May 2000. 34 pp.**

RS-00-8 **Nabil H. Mustafa and Aleksandar Pekeč.** ***Democratic Consensus and the Local Majority Rule*****. May 2000. 38 pp.**

RS-00-7 **Lars Arge and Jakob Pagter.** ***I/O-Space Trade-Offs*****. April 2000. To appear in** ***7th Scandinavian Workshop on Algorithm Theory*****, SWAT '98 Proceedings, LNCS, 2000.**

RS-00-6 **Ivan B. Damgård and Jesper Buus Nielsen.** ***Improved Non-Committing Encryption Schemes based on a General Complexity Assumption*****. March 2000. 24 pp.**

RS-00-5 **Ivan B. Damgård and Mads J. Jurik.** ***Efficient Protocols based on Probabilistic Encryption using Composite Degree Residue Classes*****. March 2000. 19 pp.**

RS-00-4 **Rasmus Pagh.** ***A New Trade-off for Deterministic Dictionaries*****. February 2000.**

RS-00-3 **Fredrik Larsson, Paul Pettersson, and Wang Yi.** ***On Memory-Block Traversal Problems in Model Checking Timed Systems*****. January 2000. 15 pp. Appears in Graf and Schwartzbach, editors,** ***Tools and Algorithms for The Construction and Analysis of Systems: 6th International Conference*****, TACAS '00 Proceedings, LNCS 1785, 2000, pages 127–141.**

RS-00-2 **Igor Walukiewicz.** ***Local Logics for Traces*****. January 2000. 30 pp.**

RS-00-1 **Rune B. Lyngsø and Christian N. S. Pedersen.** ***Pseudoknots in RNA Secondary Structures*****. January 2000. 15 pp. To appear in** ***Fourth Annual International Conference on Computational Molecular Biology*****, RECOMB '00 Proceedings, 2000.**

RS-99-57 **Peter D. Mosses.** ***A Modular SOS for ML Concurrency Primitives*****. December 1999. 22 pp.**