Theoretical Computer Science 605 (2015) 80–94

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

A local strategy for cleaning expanding cellular domains by simple robots

Rolf Klein, David Kriesel, Elmar Langetepe *

Department of Computer Science I, University of Bonn, Germany

ARTICLE INFO

Article history:
Received 4 August 2014
Received in revised form 5 May 2015
Accepted 27 August 2015
Available online 2 September 2015
Communicated by N. Ollinger

Keywords:
Motion planning
Finite automata
Expanding contamination
Cleaning strategy

ABSTRACT

We present a strategy *SEP* for finite state machines tasked with cleaning a cellular environment in which a contamination spreads. Initially, the contaminated area is of height h and width w. It may be bounded by four monotonic chains, and contain rectangular holes. The robot does not know the initial contamination, sensing only the eight cells in its neighborhood. It moves from cell to cell, d times faster than the contamination spreads, and is able to clean its current cell. A speed of $d < \sqrt{2}(h+w)$ is in general not sufficient to contain the contamination. Our strategy *SEP* succeeds if $d \geq 3(h+w)$ holds. It ensures that the contaminated cells stay connected. Greedy strategies violating this principle need speed at least $d \geq 4(h+w)$; all bounds are up to small additive constants.

1. Introduction

During the last years, researchers in technical fields have become increasingly fascinated by the potential of biological, decentrally organized systems. From myriads of fireflies powdering entire meadows with shallow light, even flashing in a synchronized way, to ant colonies with sometimes millions of individual beings building most sophisticated structures that allow for air conditioning, storage and even growth of food, such systems exhibit fault-resistance and cost-efficiency while being flexible and able to solve most complex tasks (an extensive survey can be found at, for instance, [15]).

Understanding such phenomena represents a serious challenge to theoretical computer science. Although there is a rich body of work on autonomous agents, comparably few papers offer theoretical results on agents who have limited perception, limited computing and translocating capabilities, and yet successfully deal with dynamically changing environments.

In this paper we are studying cellular environments in the plane. Two cells are adjacent if they share an edge. At each time, finitely many cells may be contaminated, all others are clean.

Definition 1. The set of all contaminated cells at a time is called *contamination*.

Definition 2. We assume that an initial contamination C has the following geometric properties. It is connected, and its outer boundary consists of four monotonic chains; they connect the extreme edges supporting the bounding box of C. Inside, C may contain rectangular holes consisting of clean cells; see Fig. 1. Let $\mathbb{C}$ be the set of all such contaminations.

* Corresponding author.
E-mail address: elmar.langetepe@informatik.uni-bonn.de (E. Langetepe).

http://dx.doi.org/10.1016/j.tcs.2015.08.036

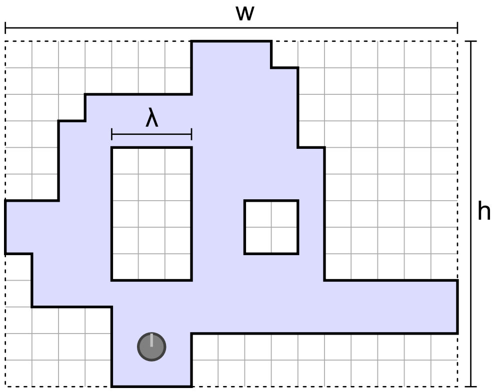

Fig. 1. An initial contamination in $\mathbb{C}$. Also depicted are the contamination's axis aligned bounding box (the rectangle outlined in a dashed way), its width (w) and its height (h). λ is the length of the longest short side among any holes.

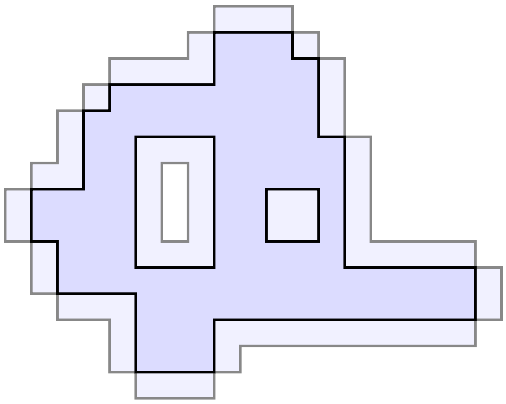

Fig. 2. The contamination from Fig. 1. The cells that will become contaminated during a spread are colored lighter.

Definition 3. Inspired by forest fires or oil spills, every d time units a contamination spreads from each contaminated cell to its four neighbors, as shown in Fig. 2.

We want to enable a robot to clean the contamination. Initially, the robot is located in one of the contaminated cells. It can sense the status of the eight cells in its neighborhood; see Fig. 1. In each time unit, the robot can turn, move to one of the four adjacent cells, and decide to clean it. Thus, d measures the robot's speed against the contamination's. The robot is a finite automaton. It has no previous knowledge about C, and because its memory is of constant size it cannot store a lot of information as it moves around. There is no global control or any other information the robot could make use of.

Whether or not environment C can be cleaned depends on its initial extension and its spreading time relative to the robot's speed, d. Let h and w denote height and width of the bounding box of C, respectively. In our model the perimeter of C, i.e., the number of edges on its outer boundary, equals $2(h + w)$. Thus, $h + w$ is a reasonable measure for the size of C; see Fig. 1.

In Theorem 3 we will establish a geometric lower bound in terms of h and w. No robot can clean all environments of height h and width w if its speed d is less than $\sqrt{2}(h + w) - 4$ (not even if the robot knows C and has Turing machine power).

Our main contribution is a strategy *Smart Edge Peeling (SEP)* for which we can prove the following performance guarantee. Let λ denote the maximum length of all shorter edges of the rectilinear holes inside C; see Fig. 1.

Theorem 1. *Given speed $d \geq 3(h + w) + 6$, and starting from a contaminated cell, strategy* SEP *cleans each contamination in $\mathbb{C}$ of height h and width w in at most $(\frac{\lambda}{2} + h + w + 5)d$ many steps.*

Starting from a contaminated cell, strategy *SEP* heads for the outer boundary of C, without attempting to enlarge any holes. Then it carefully peels the perimeter of C, layer by layer, making sure that the set of contaminated cells always stays connected. In order to maintain this invariant the strategy will *not* clean *critical* contaminated cells which would destroy connectivity locally. The strategy is precisely defined in Section 4. We have also implemented the strategy. A supplementary video of an execution of the strategy can be found at http://www.geometrylab.de/Video/smartedgepeeling.mp4.