Information and Computation 244 (2015) 122–133

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

Position discovery for a system of bouncing robots ☆

Jurek Czyzowicz [a], Leszek Gąsieniec [b], Adrian Kosowski [c], Evangelos Kranakis [d,*], Oscar Morales-Ponce [e], Eduardo Pacheco [d]

[a] Université du Québec en Outaouais, Gatineau, Québec J8X 3X7, Canada
[b] University of Liverpool, Liverpool L69 3BX, UK
[c] INRIA Bordeaux Sud-Ouest, LaBRI, 33400 Talence, France
[d] Carleton University, Ottawa, Ontario K1S 5B6, Canada
[e] Chalmers University of Technology, Maskingränd 2, 412 58 Göteborg, Sweden

ARTICLE INFO

Article history:
Received 3 October 2014
Received in revised form 19 March 2015
Available online 26 July 2015

Keywords:
Position
Detection
Mobile
Robots
Distributed
Computing
Algorithms
Complexity

ABSTRACT

A collection of n anonymous mobile robots is deployed on a unit-perimeter ring or a unit-length line segment. Every robot starts moving at constant speed, and bounces each time it meets any other robot or segment endpoint, changing its walk direction. We study the problem of *position discovery*, in which the task of each robot is to detect the presence and the initial positions of all other robots. The robots cannot communicate or perceive information about the environment in any way other than by bouncing nor they have control over their walks which are determined by their initial positions and their starting directions. Each robot has a clock allowing it to observe the times of its bounces. We give complete characterizations of all initial configurations for both the ring and the segment in which no position detection algorithm exists and we design optimal position detection algorithms for all feasible configurations.

1. Introduction

A mobile robot is an autonomous entity with the capabilities of *sensing*, i.e. ability to perceive some parameters of the environment, *communication* – ability to receive/transmit information to other robots, *mobility* – ability to move within the environment, and *computation* – ability to process the obtained data. Mobile robots usually act in a distributed way, i.e. a collection of mobile robots is deployed across the territory and they collaborate in order to achieve a common goal by moving, collecting and exchanging the data of the environment. The typical applications are mobile software agents (e.g. moving around and updating information about a dynamically changing network) or physical mobile robots (devices, robots or nano-robots, humans).

In many distributed applications, mobile robots operate in large collections of massively produced, cheap, tiny, primitive entities with very restricted communication, sensing and computational capabilities, mainly due to the limited production cost, size and battery power. Such groups of mobile robots, called *swarms*, often perform exploration or monitoring tasks in hazardous or hard to access environments. The usual swarm robot attributes assumed for distributed models include

☆ This is an extended and revised version of a paper that appeared in the proceedings of the 26th International Symposium on Distributed Computing, DISC 2012, Salvador, Brazil, October 16–18, 2012, in Lecture Notes in Computer Science, pp. 341–355.

* Corresponding author.
E-mail addresses: jurek.czyzowicz@uqo.ca (J. Czyzowicz), L.A.Gasienec@liverpool.ac.uk (L. Gąsieniec), kosowski@labri.fr (A. Kosowski), kranakis@scs.carleton.ca (E. Kranakis), mooscar@chalmers.se (O. Morales-Ponce), epacheco@cmail.carleton.ca (E. Pacheco).

http://dx.doi.org/10.1016/j.ic.2015.07.005
0890-5401/

anonymity, negligible dimensions, no explicit communication, no common coordinate system (cf. [1]). Moreover, some of these models may assume obliviousness, limited visibility of the surrounding environment and asynchronous operation. In most situations involving such weak robots the fundamental research question concerns the feasibility of solving the given task (cf. [2,3,1]). When the question of efficiency is addressed, the cost of the algorithm is most often measured in terms of length of the robot's walk or the time needed to complete the task. This is also the case of the present paper, despite the fact that the robot does not have any control over its walk. In our case, the goal is to stop the robot's walk, imposed by the adversary, at the earliest opportunity – when the collected information (or its absence) is sufficient to produce the required solution.

There have been other papers studying theoretical aspects involving extremely limited robots in the context of their positioning in simple, uni-dimensional environments (e.g. [4–6]). One of the central problems in robotics concerns the related problem of multi-robot localization or positioning (cf. [7–13]), where the robots use sensing devices (e.g. [11,12]) or they leave and observe landmarks (e.g. [7,9]) in order to exchange knowledge with other robots. The position discovery problem may be viewed as a version of mapping (cf. [14]), where the target is to discover and map of the static environment, rather than recognize the other robots.

There exists extensive literature in the robotics community on the related aspects of the problem, e.g. on robot-coordination, domain coverage and clean-up, target tracking, etc. [15] provide heuristics and a protocol to enable agents to negotiate and form coalitions (e.g., agents required to join together) when there is uncertain and heterogeneous information. Ref. [16] proves complexity results concerning the efficient use of "ant robots" for covering a connected region on the Z^2 grid, whose area n is unknown in advance, but which is expanding at a given rate. For example, the minimum number of such robots is shown to be in $\Omega(\sqrt{n})$, while $\Theta(\sqrt{n})$ is sufficient when the region is expanding at a sufficiently slow rate. Ref. [17] studies the cooperative cleaners problem which requires that several agents clean a connected region of "dirty" pixels in Z^2. Ref. [18] studies team coordination for the collision free target tracking problem of multi-agent robot system. Ref. [19] presents an approximation algorithm utilizing finite-horizon planning and implicit coordination (to achieve linear scalability in the number of searchers) for the problem of locating a mobile, non-adversarial target in an indoor environment using multiple robotic searchers. Ref. [20] is a feasibility study on building terrain-covering ant robots which leave trails in the terrain so as to cover a closed terrain repeatedly. Ref. [21] studies both theoretically and by simulation the behavior of ant robots for one-time as well as repeated coverage of a terrain (e.g., for lawn mowing, mine sweeping, patrolling, etc).

Although the most frequently studied question for mobile robots is environment exploration, numerous papers related to such weak robots often study more basic tasks, such as pattern formation [3,22,1,23]. Gathering or point convergence [24, 25] and spreading (e.g. see [26]) also fall into this category. Ref. [1] introduced anonymous, oblivious, asynchronous, mobile robots which act in a so-called *look-compute-move* cycle. An important robot sensing capacity associated with this model permits to perceive the entire [3,22,1] or partial [27,25] environment.

Contrary to the above model, in our paper, a robot has absolutely no control over its movement, which is determined by the bumps against its neighbors or the boundary points of the environment. In [28,29] the authors introduced *population protocols*, modeling wireless sensor networks by extremely limited finite-state computational devices. The agents of population protocols also move according to some mobility pattern totally out of their control and they interact randomly in pairs. This is called *passive mobility*, intended to model, e.g., some unstable environment, like a flow of water, chemical solution, human blood, wind or unpredictable mobility of agents' carriers (e.g. vehicles or flocks of birds). In the recent work [30], a coordination mechanism based on meetings with neighboring robots on the ring was considered, also aiming at location discovery. The approach of [30] is randomized and the robots operate in the discrete environment in synchronous rounds.

Pattern formation is sometimes considered as one of the steps of more complex distributed task. Our involvement in the problem of this paper was motivated by the patrolling problem [31], where spreading the robots evenly around the environment may result in minimizing the *idleness* of patrolling, i.e., the time interval during which environment points remain unvisited by any robot. Clearly, position discovery discussed in the present paper is helpful in uniform spreading of the collection. A related problem was studied in [26], where the convergence rate of uniform spreading in a one-dimensional environment in synchronous and semi-synchronous settings was discussed. Previously, [32] studied the problem of n robots $\{0, 1 \dots, n-1\}$, initially placed in arbitrary order on the ring. It was shown that the rule of each robot i moving to the middle point between $i-1$ and $i+1$ may fail to converge to equal spreading (it was also shown in [32] that the system would converge if a fair scheduler activates units sequentially).

The model adopted in our paper assumes robot anonymity, passive mobility (similarly to that adopted in [28,29]), restricted local sensing through bounce perception with a neighbor robot only, no communication between the robots, and continuous time. The only ability of the robot is the tacit observation of the timing of bounces and the computation and reporting of robots' locations. The clock of each robot turns out to be a very powerful resource permitting to solve the problem efficiently in most cases.

2. The model and our results

We consider a continuous, connected, one-dimensional universe in which the robots operate, which is represented either by a unit-perimeter ring or by a unit-length line segment. The ring is modeled by a real interval $[0, 1]$ with 0 and 1 corresponding to the same point. A set of n robots $r_0, r_1, \dots, r_{n-1}$ is deployed in the environment and each of them starts moving at time $t = 0$ (where the indexing of the robots is used for purposes of analysis, only). The robots are not aware