SIAM J. COMPUT.
Vol. 36, No. 1, pp. 279–280
© 2006 Society for Industrial and Applied Mathematics

ERRATUM: DISTRIBUTED ANONYMOUS MOBILE ROBOTS: FORMATION OF GEOMETRIC PATTERNS*

ICHIRO SUZUKI† AND MASAFUMI YAMASHITA‡

Abstract. In this note we make a minor correction to a scheme for robots to broadcast their private information. All major results of the paper [I. Suzuki and M. Yamashita, *SIAM J. Comput.*, 28 (1999), pp. 1347–1363] hold with this correction.

Key words. anonymous robots, broadcast

AMS subject classification. 68Q99

DOI. 10.1137/050631562

1. Correction. Algorithms $\psi_{f-point(2)}$ in section 3 and $\psi_{getview}$ in section 4 of [1] use the following technique for all robots r_i to simultaneously "broadcast" to the other robots a privately chosen directed line ℓ_i (e.g., the positive x-axis of its local coordinate system Z_i). As outlined in the paragraph that follows the proof of Theorem 3.4, the basic idea is that each r_i moves repeatedly along ℓ_i in a fixed direction each time it becomes active until, for each $j \neq i$, it has seen r_j at two or more distinct locations. Robot r_i can then figure out ℓ_j based on two distinct locations that r_j has occupied. In an effort to ensure at the same time that r_j has also seen r_i at two or more distinct locations (so that it can figure out ℓ_i), we made an incorrect claim that if r_i has observed r_j at three or more distinct locations, then r_j has observed r_i at two or more distinct locations. This claim must be replaced by the following.

PROPOSITION 1. *For any integer $m \geq 1$, if r_i has seen r_j at $2m$ distinct locations in the time interval $[0, t]$, then r_j has seen r_i at m or more distinct locations in $[0, t)$.*

Proof. Suppose r_i becomes active at times t_1 and t_2, $t_1 < t_2$, and observes r_j at two distinct locations. Since r_j occupies distinct locations at t_1 and t_2, r_j must become active in $[t_1, t_2)$ and observe r_i at a location on the line segment $\overline{p_1 p_2}$, where p_1 and p_2 are the locations of r_i on line ℓ_i at t_1 and t_2, respectively. This means that r_j observes r_i at a distinct location each time r_i observes r_j at two distinct locations, since r_i moves along ℓ_i in a fixed direction each time it becomes active. □

Therefore, to ensure that r_j has seen r_i at two distinct locations, r_i must continue to move along ℓ_i until it has seen r_j at four or more distinct locations. However, if we allow r_i to simply stop moving as soon as it has observed r_j at four or more distinct locations, then r_j may not be able to observe r_i at four or more distinct locations. This means that r_j may never finish the broadcast.

Fortunately, broadcasting a line is usually a preliminary step that precedes a main task. In the case of $\psi_{f-point(2)}$, the ultimate goal of r_i and r_j is to move to the midpoint p of their initial positions. (Because of the way ℓ_i and ℓ_j are chosen in $\psi_{f-point(2)}$, both r_i and r_j can compute p from ℓ_i and ℓ_j.) Note that when r_i has seen r_j at four or more distinct locations, r_i knows not only ℓ_j, but also that r_j knows ℓ_i (because by Proposition 1 r_j has seen r_i at two or more distinct locations). Thus r_i

*Received by the editors May 13, 2005; accepted for publication (in revised form) February 22, 2006; published electronically June 19, 2006.
http://www.siam.org/journals/sicomp/36-1/63156.html
†EECS, University of Wisconsin-Milwaukee, Milwaukee, WI 53201 (suzuki@cs.uwm.edu).
‡Department of Computer Science and Communications Engineering, Kyushu University, Fukuoka 812-8581, Japan (mak@csce.kyushu-u.ac.jp).

can now safely quit broadcasting ℓ_i and move to (or toward) p, regardless of whether r_j has seen r_i at four or more distinct locations. By moving to a point not on ℓ_i, r_i effectively "announces" to r_j that it now knows ℓ_j. Robot r_j eventually observes r_i at a location not on ℓ_i and learns that it can quit broadcasting ℓ_j as well and proceed to p.

Based on the above discussion, r_i can use the following scheme to broadcast ℓ_i to all other robots:

1. r_i moves along ℓ_i in a fixed direction each time it becomes active until, for each $j \neq i$, either
   (a) it has seen r_j at four or more distinct locations, or
   (b) it observes that r_j is at a location not on ℓ_j. (Note that r_i knows ℓ_j by the time this case occurs.)
2. Then it moves to a point not on ℓ_i.

Algorithm $\psi_{f-point(2)}$, the proof of Theorem 3.3, and the paragraph that follows the proof of Theorem 3.4 should be revised accordingly.

Algorithm $\psi_{getview}$ must be modified using a similar idea. In $\psi_{getview}$, each robot r_i, initially located at the origin o_i of its local coordinate system Z_i, broadcasts three lines: its x-axis, y-axis, and line L_i through o_i in direction $f(d_i)$, where d_i is the minimum distance between any two initial positions of the robots, and for $x > 0$, $f(x) = (1 - 1/2^x) \times 90°$ is a monotonically increasing function with range $(0°, 90°)$. (This function replaces $f(x) = (1 - 1/2^x) \times 360°$ used in the paper. Both $f(d_i)$ and d_i are measured in terms of Z_i.) Note that the orientations of the three lines are all distinct.

Each robot r_i first broadcasts its x-axis by moving along it in the positive direction. When r_i knows that all other robots know r_i's x-axis and its orientation, i.e., for each $j \neq i$, either

(a) r_i has seen r_j at four or more distinct locations, or
(b) r_i observes that r_j has changed the direction of its motion (r_i knows the x-axis of r_j by the time this occurs),

it returns straight to o_i and starts broadcasting its y-axis by moving along it in the positive direction (thereby changing its direction of motion and announcing to others that it now knows the x-axes of all other robots). Eventually all robots finish broadcasting their x-axes and start broadcasting their y-axes. r_i ends the broadcast of its y-axis when it knows that all other robots know its y-axis and its orientation, returns straight to o_i, and starts broadcasting line L_i by moving in direction $f(d_i)$ (thereby changing its direction of motion again, announcing to others that it knows the y-axes of all other robots). Eventually all robots r_j start broadcasting their lines L_j. Once again, r_i terminates this broadcast when it knows that all other robots know L_i, and returns to its initial position o_i. By returning to o_i, r_i announces to others that it has finished the broadcast of L_i.

REFERENCE

[1] I. Suzuki and M. Yamashita, *Distributed anonymous mobile robots: Formation of geometric patterns*, SIAM J. Comput., 28 (1999), pp. 1347–1363.