Robust and Efficient Covering of Unknown Continuous Domains with Simple, Ant-Like A(ge)nts

Eliyahu Osherovich, Vladimir Yanovsky, Israel A. Wagner and Alfred M. Bruckstein

Abstract—We propose a new “Mark-Ant-Walk” algorithm for robust and efficient covering of continuous domains by ant-like robots with very limited capabilities. The robots can mark places visited with pheromone marks and sense the level of the pheromone in their local neighborhood. In case of multiple robots these pheromone marks can be sensed by all robots and provide the only way of (indirect) communication between the robots. The robots are assumed to be memoryless, and to have no information besides that mentioned above. Despite the robots’ simplicity, we show that they are able, by running a very simple rule of behavior, to ensure efficient covering of arbitrary connected domains, including non-planar and multidimensional ones. The novelty of our algorithm lies in the fact that, unlike previously proposed methods, our algorithm works on continuous domains without relying on some “induced” underlying graph, that effectively reduces the problem to a discrete case of graph covering. In addition we demonstrate various benefits of our successive visits of the robot at the same location, that makes it suitable for patrolling.

Finally we provide a new theoretical bound on covering time of a wide family of such mark and walk algorithms.

I. INTRODUCTION

Suppose we want to cover (or clean or search or paint) a connected domain in $\mathbb{R}^2$ simple robots having effectors (or arms) that can sweep a well-defined neighborhood of the robots when they are stationary. We shall say that a domain was covered by the (team of) robots if each and every point of the domain was swept by a robot effector. In fact, every time we want to build an automatic machine suitable for applications such as floor cleaning, snow removal, lawn mowing, painting, mine-field de-mining, unknown terrain exploration and so forth, we face the problem of complete covering of domains by such devices.

A. Problem Constraints

The approach to solving covering problems depends, of course, on the capabilities we assume for our robots and on various environmental constraints. Hence many algorithms can be, and actually have been, developed to accommodate constraints and assumptions on the robots used for the covering problem. The various considerations are:

This research was partly supported by the Israeli Ministry of Science infrastructural grant No. 3-942 and Israeli Ministry of Defense (MAFAT) grant.

E. Osherovich, V. Yanovsky, and A. M. Bruckstein are with the Department of Computer Science, Technion, Haifa
I. A. Wagner is with IBM Haifa Labs

1) The domain type (e.g., discrete versus continuous, a simply connected or multiply connected region, a general graph or a grid, etc.)
2) The capabilities of robots (their communication capabilities, the amount of on-board memory, the size of their footprint and the areas swept by the effectors and range of robots’ sensors)
3) The type of information the robots are assumed to be able to get or gather via their sensors either global or local, off-line or on-line.
4) The local behavior and interaction model in case of multiple robots (such as synchronous or asynchronous operation, centralized or distributed control, etc.)

In this paper we adopt the model proposed in [1], which assumes that the robots are anonymous, (i.e., all robots are identical), memoryless, (i.e., have no ability to “remember” anything from the past), and have no means of direct communication which means there is no direct exchange of information between the robots. The only means of (indirect) communication is via some marks they leave in their environment. This model was originally inspired by ants and other insects using chemicals called pheromones that are left on the ground as a mean of achieving indirect communication and coordination. Ant colonies, despite the simplicity of single ants, demonstrate surprisingly good results in global problem solving and pattern formation [3]–[7]. Consequently, ideas borrowed from insects (and especially ants) behavior research are becoming increasingly popular in robotics and distributed systems [6]–[11]. Simple robots were found capable of performing quite complex distributed tasks while enjoying benefits of being small, cheap, easy to produce and easy to maintain.

This paper is organized as follows. Our formal robot model is presented in Section II-A. In Section II-B we define the Mark-And-Walk (MAW) covering algorithm, followed by a short survey of previously proposed covering algorithms in Section III. As mentioned earlier the number of such algorithms is fairly large, therefore, we limited our survey to those that share some common principles with the algorithm suggested in this paper. Formal proofs of complete coverage and efficiency analysis are given in Sections IV-A and IV-B respectively. In Section V we provide various extensions of

1It is fair to say that recent publications show that ants are not memoryless (see, e.g., [2]). However it is certainly true that for certain ant species these individual capabilities play a limited role in navigation and trail laying or trail following mechanisms.
the basic MAW algorithm including its multi-agent or swarm system performance and its behavior when the environment contains noise (false pheromone marks). Results of simulations are presented in Section VI. Section VII provides a summary of our results and a discussion of possible extensions and implementation details.

II. The Basic Mark-Ant-Walk (MAW) Covering Algorithm

A. Robot Model

Below we define the mathematical problem of robot covering along with the model for the robots that we use throughout this paper.

The domain to be covered will be denoted by $\Omega$. At the moment we consider only two-dimensional domains (however, extensions will be given in Section V-D). Given any two points $a, b \in \Omega$ denote the geodesic distance between $a$ and $b$ as $\|a - b\|$, i.e., the length of the shortest path restricted to lie entirely in the domain $\Omega$ that connects $a$ and $b$. For the sake of brevity we shall omit the word “geodesic” and use simply “distance”. At the moment, we assume that this length is measured as a common Euclidean length in two-dimensional space; extensions to other measures are discussed in Section V-D. We shall then assume that the robot is able to sense the pheromone level at its current position $p$ and in a closed “geodesic” ring $R(r, 2r, p)$ lying between the internal radius $r$ and the external radius $2r$ around $p$. $R$ is formally defined as follows:

$$R(r_1, r_2, c) = \{ a \in \Omega \mid r_1 \leq \|a - c\| \leq r_2 \},$$  \hspace{0.5cm} (1)

and $r$ is considered to be an intrinsic parameter of the robot.

Additionally, our robot is able to set a constant arbitrary pheromone level in the area swept by its effector, which is, we assume, an open disk $D(r, p)$ of radius $r$ around its current location $p$. The formal definition of $D$ is as follows:

$$D(r, c) = \{ a \in \Omega \mid \|a - c\| < r \}. $$ \hspace{0.5cm} (2)

Note that, as we mentioned before, all distances are measured as geodesic ones, hence, for example, only area $A$ in Figure 2 will be available to the robot. Area $B$, on the other hand, will not be “visible” to the robot since the distance from the robot location $p$ to any point in $B$ is greater than $2r$.

Furthermore, we shall assume that our time steps are discrete. We denote by $\sigma(a, t)$ the pheromone level of point $a \in \Omega$ at time instance $t$, where $t = 0, 1, 2, 3, \ldots$

**B. The Mark-Ant-Walk (MAW) Algorithm**

Initially, we consider the case where no point is marked with the pheromone, thus all $\sigma$ values are assumed to be equal to zero:

$$\sigma(a, 0) = 0; \forall a \in \Omega.$$ \hspace{0.5cm} (3)

Suppose further that a starting point is (randomly) chosen for a (single) robot and then a MAW step rule (as described in Fig 3) is applied repeatedly. Note that there is no explicit stopping condition for this algorithm; nevertheless, one can use an upper bound (that will be provided later in this paper) on the cover time in order to stop robots after a sufficient time period has elapsed that effectively guarantees complete covering. It is easy to see that the robot markings create some kind of potential field, where high values of the pheromone level roughly indicate areas that have been visited many times up to the moment and lower values of the pheromone level correspond to a smaller number of robot’s visits. According to the MAW rule, the robot tries to avoid areas with high pheromone values by moving toward lower levels, striving to reach areas yet uncovered.

III. Related Work

A first step toward a “pheromone-marking”-oriented model was taken by Blum and co-workers [12], [13] where pebbles

<table>
<thead>
<tr>
<th>Mark-Ant-Walk step rule (current time is $t$ and robot location is $p$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Find $x$ := a point in $R(r, 2r, p)$ with minimal value of $\sigma(x, t)$ (In case of a tie, i.e., when the minimal value achieved at several places - make an arbitrary decision) /* note that $|p - x| \geq r$ */</td>
</tr>
<tr>
<td>2. If $\sigma(p) \leq \sigma(x)$ then $\forall u \in D(r, p)$ set $\sigma(u) = \sigma(x) + 1$ /* we mark open disk of radius $r$ around current location */</td>
</tr>
<tr>
<td>3. $t := t + 1$</td>
</tr>
<tr>
<td>4. move to $x$</td>
</tr>
</tbody>
</table>

Fig. 2. Geodesic distance example

Fig. 3. MAW step rule
were used to assist the search. Pebbles are tokens that can be placed on the ground and later be removed. The idea of using pebbles for unknown graph exploration and mapping was further developed in [14].

A. Discrete Domain Covering Problems

Covering of discrete domains (graphs) is an important problem in theoretical computer science and thus a number of solutions have been proposed and comprehensively analyzed by researchers. Well known examples are the Breadth-First Search (BFS) and the Depth-First Search (DFS) algorithms for graph traversal. Both algorithms provide excellent results in terms of time complexity. Formal proofs of complete coverage and efficiency analysis can be found in books on discrete mathematics and algorithms (see, e.g., [15]). Note that the DFS algorithm can be readily adapted to fit our robot model as opposed to the BFS algorithm which requires additional on-board memory in order to maintain a queue of already discovered, but yet unvisited vertices. Moreover, the amount of on-board memory required depends on the graph to be explored.

Two further algorithms that fit our paradigm entirely, i.e., they rely on a group of identical autonomous robots that mark the ground with pheromones, were suggested for efficient and robust graph covering. One, called the Edge-Ant-Walk, marks the graph edges [16]. Another one, called the Vertex-Ant-Walk, leaves marks on graph vertices instead [1], [17]. Both algorithms provided significant improvement over DFS in terms of robustness along with quite efficient cover times. Both are capable of completing the traversal of the graph with multi-robot teams even in the case when almost all robots die and/or the environment graph changes (edges/vertices are added or deleted) during the execution.

B. Continuous Domain Covering Problems

Covering continuous domains is a relatively new problem. Several algorithms addressing this problem are summarized in a good survey paper by Choset [18]. We shall provide below a short review of algorithms developed for robot models that are close to the one used in this paper.

1) Random Walk and Probabilistic Covering: Random walks were defined for both discrete and continuous domains and enjoy unrivaled robustness and scalability; however, they cannot guarantee complete coverage and can be analyzed in terms of expected coverage time only. Though some researchers have demonstrated a way to estimate this expected coverage time, i.e., in terms of electrical resistance of the domain [19], [20], in this work we would like to have solutions that guarantee complete coverage after a deterministic and bounded time period.

2) Motion Planning Guided by a Potential Field: A popular approach is to introduce an artificial potential field concept in order to accomplish the robot motion planning task (e.g., [21], [22]). In [22], for example, the authors used the distance transform as the potential field. Such an approach could also be adopted by our robots, the potential being represented by the odor level. However, this type of work assumes that the potential field can be constructed prior to the start of robot’s motion and thus requires a global knowledge of the domain boundaries and obstacles. Such knowledge is not available in our model.

3) Trail Lying Algorithms for Continuous Domains: Several authors proposed to use trails that mark the path traveled by the robot so far, and proposed local behavior for the robots that resulted in some kind of peeling/milling of the domain. Here two major approaches for organizing the motion exist: contour-parallel and direction-parallel. In the former approach, the robot moves along the boundaries of the domain; in the latter one, the robot moves back and forth in some predefined direction. These approaches often fail for non-convex domains and thus the whole domain must be approximated as a union of convex non-overlapping cells, which, in turn, can be covered with the assumed type of motion [23]–[25]. Another representative of trail-based algorithms is the Mark-And-Cover (MAC) algorithm presented in [19], which is actually an adaptation of the DFS algorithm to continuous domains. This algorithm provides efficient and effective coverage with a provable bound on cover time. Additionally, the robot model used in the paper fits our paradigm very well. Nevertheless, the drawback of the MAC algorithm, and, in fact, of all trail-based algorithms, is their sensitivity to noise and robot failure. Moreover, in the multi-robot case trails of the robots interfere with the motion of the others and may hamper their efficiency. Another shortcoming of these algorithms is seen in the situation when domain needs to be covered repeatedly, e.g., in surveillance tasks or in the scenario described in [26] where autonomous robots are used to de-mine minefields using imperfect sensors (in the sense that the probability of a mine detection is less than 1). Our algorithm guarantees that the whole domain is covered repeatedly, time after time. Furthermore, the time between successive visits at any point can be bounded in terms of the (unknown) size of the problem (see Section V-A).

4) Tessellating Algorithms for Continuous Domains: Another possible approach is to split the domain into tiles such that each tile is easily covered by the robots (e.g., convex tiles are suitable for “onion peeling” or “back and forth” algorithms). After a particular tile is covered, the robot goes to a new one that is a neighbor of the current tile [11], [23], [27]. This approach, in fact, takes us back to a graph covering of an underlying graph whose vertices are associated with the tiles, the edges between vertices being defined according to the inter-tile connectivity.

IV. THE MAW ALGORITHM: FORMAL PROOF AND EFFICIENCY ANALYSIS

A. MAW - Proof of Complete Coverage

Let us first show that a single robot governed by the MAW rule covers any connected bounded domain in a finite number of steps. The outline of the proof is as follows.

First, we prove that at any time instance, any two points that are close enough, i.e., their distance from each other is less than or equal to r, must have pheromone levels that differ by one at most. This property closely resembles Lipschitz
continuity for functions $f(\cdot)$:
\[ |f(p_1) - f(p_2)| \leq \kappa \|p_1 - p_2\|, \tag{4} \]
for some constant $\kappa$. However, our $\sigma$ function measuring the level of the pheromone at each location is, by definition, not continuous and thus cannot be Lipschitz continuous. Instead, $\sigma$ obeys the following inequality:
\[ |\sigma(p_1) - \sigma(p_2)| \leq \kappa \|p_1 - p_2\| + 1. \tag{5} \]

We call this the **proximity principle**, and it has also been used in several previous papers [1], [16], [17].

Second, we look at the diameter $d$ of the domain, defined as the length of the longest geodesic path that can be embedded in the domain: consider a pair of points $a, b \in \Omega$. Since $\Omega$ is connected, there is at least one path that connects $a$ and $b$ and lies entirely in the domain $\Omega$. Among all such paths (including $a$ and $b$, that are restricted to lie entirely in the domain $\Omega$), there is at least one that is the shortest. We call the length of this shortest path the **distance** between $a$ and $b$ and denote it by $\|a - b\|$. Among all possible pairs of points $a, b \in \Omega$, there exists a pair $(a_0, b_0)$ that has the greatest possible distance between them, i.e.,
\[ \|a_0 - b_0\| \geq \|a - b\| \ \forall a, b \in \Omega. \tag{6} \]

We call the length of this longest geodesic path the **diameter** of the domain and denote it by $d$. In other words:
\[ d \triangleq \sup_{(a, b) \in \Omega} \|a - b\|. \tag{7} \]

Assuming that $d$ is finite, we easily conclude with the aid of the proximity principle that, at any time $t$ for any two points $a, b \in \Omega$, the difference between the pheromone levels of these two points is upper bounded by $[d/r]$. This, in turn, means that once a value of $[d/r] + 1$ is reached by $\sigma(t)$ at any point, no unmarked point may exist and thus the whole domain has been covered by the robot. Finally, we show that the maximal pheromone marker value goes to infinity as time goes to infinity hence we shall surely, at some time reach the value of $\lceil d/r \rceil + 1$. Formal proofs are given below.

**Lemma 1**

The difference between marker values of near by points is bounded.
\[ \forall t; \forall a, b \in \Omega : \text{if } \|a - b\| \leq r \text{ then } |\sigma(a,t) - \sigma(b,t)| \leq 1 \]

**Proof:** Note that the distance $\|a - b\|$ is the length of the shortest path between these two points, which is restricted to lie entirely in the domain $\Omega$. We shall prove the lemma by mathematical induction on the step number. The lemma is clearly true at $t = 0$ when all the marks are assumed to be zero. Assuming it is also true at time $t$, we shall show it remains true at time $t + 1$. Let us look at two arbitrary points $a, b \in \Omega$, such that $\|a - b\| \leq r$. In the trivial case neither $a$ nor $b$ change their marker values at the $(n+1)$th step; therefore, the lemma continues to hold. If both $a$ and $b$ change their values, then $\sigma(a, t + 1) = \sigma(b, t + 1)$ since the algorithm assigns the same values to all the points it changes. Hence the only interesting case is when only one point (say $a$) changes its marker value, while the other one remains unchanged. Assuming the current robot’s location is $p_t$ we conclude that $a$ belongs to $D(r, p_t)$, otherwise it could not change its marker value. Therefore, $\|a - p_t\| < r$. However, $b$ does not change its marker value and thus $\|b - p_t\| \geq r$. Combining these constraints we get:
\[ \begin{cases} \|a - b\| \leq r \\ \|a - p_t\| \leq r \Rightarrow r \leq \|b - p_t\| \leq 2r, \end{cases} \tag{8} \]
or, equivalently, $b \in R(r, 2r, p_t)$. Now let us recall how the new marker value of $a$ is determined. First, we look for the minimal marker value among all points in $R(r, 2r, p_t)$. Assume that this value is attained at some point $x \in R(r, 2r, p_t)$. The new marker value of $a$ is then set if and only if the pheromone level at the current robot’s location is smaller than or equal to that at $x$: $\sigma(p_t, t) \leq \sigma(x, t)$. In this case we have:
\[ \sigma(a, t + 1) = \sigma(x, t) + 1. \tag{9} \]

Since both points $x$ and $b$ belong to $R(r, 2r, p_t)$, we have
\[ \sigma(b, t) \geq \sigma(x, t), \tag{10} \]
because of the way the point $x$ was chosen. Now, on the one hand, we have:
\[ \begin{cases} |\sigma(a,t) - \sigma(b,t)| \leq 1 \\ \sigma(b,t) \geq \sigma(x,t) \Rightarrow \sigma(a,t) \geq \sigma(x,t) - 1; \end{cases} \tag{11} \]
and on the other hand:
\[ \begin{cases} |\sigma(a,t) - \sigma(p_t,t)| \leq 1 \\ \sigma(p_t,t) \leq \sigma(x,t) \Rightarrow \sigma(a,t) \leq \sigma(x,t) + 1. \end{cases} \tag{12} \]
Combining inequalities (11) and (12), we get
\[ |\sigma(a,t) - \sigma(x,t)| \leq 1. \tag{13} \]

Using the system of inequalities (11), we conclude that
\[ 0 \leq \sigma(b,t) - \sigma(x,t) \leq 2. \tag{14} \]
Combining the above inequality with the fact that $\sigma(a, t+1) = \sigma(x, t) + 1$ and $\sigma(b, t+1) = \sigma(b, t)$, we get the desired result:
\[ |\sigma(a, t + 1) - \sigma(b, t + 1)| \leq 1. \tag{15} \]

Thus the lemma is proved.

**Lemma 2**

The difference between marker values of any two points is bounded at any time instance.
\[ \forall t; \forall a, b \in \Omega : |\sigma(a,t) - \sigma(b,t)| \leq \left[ \frac{d}{r} \right], \]
where $d$ is the diameter of $\Omega$.

**Proof:** Let us consider a path connecting the points $a$ and $b$. We can always split the path into sub-paths of length $r$, as depicted in Figure 4. According to Lemma 1 difference between the marker values at the endpoints of every such sub-path is limited by 1, hence, we can conclude that
\[ |\sigma(a,t) - \sigma(b,t)| \leq \left[ \frac{l}{r} \right], \tag{16} \]
where $l$ represents the length of the path. Obviously, among all paths connecting $a$ and $b$ the shortest path will provide best
Imagine that the domain \( \Omega \) is tessellated into \( n \) cells so that the diameter of every such cell is less than \( r \) (for convex cells we can, alternatively, require that the diameter of the circumscribing circle must be less than \( r \)). Let us examine the following expression:

\[
S_t = \sum_{i=1}^{n} m^i_t - \sigma(p_t, t),
\]

where \( m^i_t \) is the minimal marker value over the \( i \)-th cell at time \( t \) and \( \sigma(p_t, t) \) is the marker value at the robot’s location \( p_t \) at time instance \( t \). It was shown in Lemma 3 that marker values of any point inside \( \Omega \) form a non-decreasing sequence. Hence we claim that

\[
S_{t+1} > S_t.
\]

Indeed, for the non-marking step, the sum of the minima does not change: \( \sum_{i=1}^{n} m^i_t = \sum_{i=1}^{n} m^i_{t+1} \); however \( \sigma(p_t, t) > \sigma(p_{t+1}, t+1) \), and therefore, \( S_{t+1} > S_t \). For the marking step, assuming that the robot goes from cell \( k \) into cell \( l \), we have \( \sigma(p_t, t) \leq \sigma(p_{t+1}, t+1) \), and therefore, \( m^k_t \leq \sigma(p_{t+1}, t+1) \). Additionally, the whole cell \( k \) was marked during this step and thus \( m^k_{t+1} = \sigma(p_{t+1}, t+1) + 1 \). Hence we have

\[
\begin{align*}
& m^k_{t+1} - \sigma(p_{t+1}, t+1) = 0 \\
& m^k_{t+1} - \sigma(p_{t+1}, t+1) = 1.
\end{align*}
\]

Since the sum of the other minima cannot decrease as was shown in Lemma 3, we conclude again that \( S_{t+1} > S_t \). Given that \( S_0 = 0 \), we readily conclude that

\[
S_t \geq t \Rightarrow \sum_{i=1}^{n} m^i_t \geq t \quad \forall t,
\]

which leads us to the conclusion that there exists \( k \in \{1, 2, \ldots, n\} \) such that

\[
m^k_t \geq \frac{t}{n}.
\]

Hence the lemma is proved.

At this point we are ready to prove the main result of this work.

**Theorem 1**

The domain \( \Omega \) will be covered within a finite number of steps.

**Proof:** According to Lemma 4 after \( n \left\lceil \frac{d}{r} \right\rceil + 1 \) steps, at least one of the \( m^i \) values will be greater than \( \left\lceil \frac{d}{r} \right\rceil \) and thus the whole domain will be covered.

**B. MAW - Efficiency Analysis**

As we proved earlier the domain \( \Omega \) will be covered by a single robot after \( n \left\lceil \frac{d}{r} \right\rceil + 1 \) steps where \( d \) is the diameter of the domain, \( r \) - the covering radius of the robot’s effector and \( n \) - the number of cells in some tessellation of \( \Omega \) (see Lemma 4 for the definition of \( n \)). We shall now analyze how good this upper bound is. In order to make such comparison we shall find an approximation to \( n \) and to find out what is the best upper bound possible.
Let us denote by $N_r$ the minimal number of steps required by the robot to cover the whole domain $\Omega$ (here we only assume that the robot can cover an open disk of radius $r$ at every step, however, no assumption is made regarding the algorithm governing the robot’s behavior). Clearly $N_r \geq \frac{A_\Omega}{\pi r^2}$, where $A_\Omega$ and $a_r$ are the areas of the domain $\Omega$ and robot’s effectors respectively. Obviously, this is an ultimate lower bound and no algorithm can beat it. However this bound fails to be tight enough for domains whose shape factor (ratio of the squared domain perimeter to its area multiplied by $4\pi$) is far from 1 (not round). For example, domains, comprised of a finite number of curves and line segments would have zero area providing lower bound to be zero, which is, of course, meaningless. As a possible solution, in [19] the authors used the area of the “augmented” domain $\bar{\Omega}$, which results from an inflation or expansion $\Omega$ in all directions by $r$, i.e., $\bar{\Omega}$ has undergone a morphological dilation with a disk of radius $r$. In this case we have [19]:

$$A_\bar{\Omega} \leq A_\Omega + rP_\Omega + \pi r^2,$$

(26)

where $A_\Omega$ and $P_\Omega$ are the area and the perimeter of $\Omega$, respectively.

We suggest another way of performance assessment which is correct by construction and does not depend on geometric properties like area or perimeter. Consider best possible algorithm that covers in each step an open disk of radius $r/2$, besides this requirement we do not limit the algorithm in any other way. Assume that the domain $\Omega$ can be covered by such an algorithm, i.e., there exists a finite sequence of points $P_1, P_2, \ldots, P_N$ of robot’s locations that results in complete coverage, that is:

$$\Omega \subset \bigcup_{i=1}^{N} D\left(\frac{r}{2}, P_i\right).$$

(27)

Alternatively, we can say that for every point $a$ in $\Omega$ there exist a number $k$ ($1 \leq k \leq N$) such that $a \in D(r/2, P_k)$. Now, if we consider the best possible algorithm, as before we denote its coverage time by $N_{r/2}$, we easily conclude that the upper bound on the number of steps required by MAW algorithm is:

$$N_{r/2} \left\lceil \frac{d}{r} \right\rceil + 1.$$  

(28)

Indeed, if we consider a particular coverage path or the robot described by the sequence of its successive positions: $P_1, P_2, \ldots, P_{N_{r/2}}$, we always can perform Voronoi tessellation around these points. Each cell in this tessellation will have diameter smaller than $r$ and thus this tessellation will be like the one we used in Lemma 4.

We have only to estimate the $d/r$ fraction. Obviously, $\frac{d}{r} \leq N_{r/2}$, on one hand, on the other hand we can estimate the lower bound: $\frac{d}{r} \geq \frac{\sqrt{N_{r/2}}}{\pi}$ (for domains that have a shape close to a circle). Hence we have shown that the upper bound time is polynomial with respect to best possible time of algorithm whose covering radius is $r/2$.

$$t_{\text{coverage}} \leq N_{r/2}^r,$$

(29)

where $0.5 \leq x \leq 2$.

The main question here is whether we can conclude that

$$t_{\text{coverage}} \leq N_r^x,$$  

(30)

i.e., whether our coverage time is bounded by a polynomial function of $N_r$, which is best possible coverage time among all algorithms whose covering radius is $r$. The general answer is “no”. In fact, a general theorem regarding limitations of such a type of algorithms is the following:

**Theorem 2**

*Given an algorithm whose marking area is an open disk of radius $r$ and step size is grater than or equal to $r$. The time for complete coverage $t_{\text{coverage}}$ of a continuous domain is (tightly) limited from below by $N_{r/2}$. Moreover $t_{\text{coverage}}$ cannot be expressed as a bounded function of $N_r$ for any $x > r/2$.**

**Proof:** The proof is by example of such a domain: we consider a domain comprised of $n$ line segments emanating common origin $O$, as shown if Figure 5. Assuming that the length of each line segment is $r/2$ we easily verify that $N_{r/2} = n$ and that $t_{\text{coverage}} = n$ (assuming that the robot’s initial position was an end-point of any line segment. Hence $t_{\text{coverage}}$ is (tightly) bounded from below by $N_{r/2}$. Now if we assume that $n$ is infinite we easily conclude that such domain cannot be covered in a finite number of steps by our algorithm (again, provided that the initial robot location was at an end point of any line segment), while this domain can be covered in two steps by an optimal algorithm whose covering radius is greater than $r/2$ (we simply go to the origin $O$ in the very first step and the whole domain will be covered in the next step). Hence, the theorem is proved.

![Fig. 5. A “pathological” star shaped domain.](image)
that do not include the boundary of the domain are convex, those that do include domain’s boundaries may not be convex. Each convex cell in this tessellation can be covered by a finite (and well-defined) number of disks of radius \( r/2 \), hence for these cells we can conclude that \( N_r \propto N_{r/2} \). For non-convex cells this claim may not be correct, thus, for domains whose tessellation consists mainly of convex cells we have approximately \( N_r \propto N_{r/2} \). Note that a similar analysis was carried out on the MAC algorithm [19] where the authors claim that MAC algorithm is asymptotically linear for domains that obey \( \Lambda >> Pr \). This is, fact, equivalent to say that the number of cells in the Voronoi tessellation described above that do not contain domain’s boundary is large compared to the number of the cells that do.

V. Extensions

A. Repetitive Coverage

In some scenarios we might be interested in repetitive coverage of the domain. For example, repetitive coverage is necessary in the aforementioned scenario when robots perform minefield de-mining and their mine detection is not perfect, i.e., the probability of detecting a mine when the robot’s sensors sweep above it is less than one. In this case repetitive coverage is required to minimize the probability of leaving any mines undetected. In this case, we have to give an upper bound on time between two successive visits of the robot in order to guarantee an improvement in detection probability. This requirement also arises naturally in tasks such as surveillance or patrolling where robots are required to visit every point over and over and the time between two successive visits must be limited by a constant. We shall now show that our algorithm has the property of patrolling. Before we start proving this result we shall need the following lemma.

Lemma 5

For any two time instances \( t_1 \) and \( t_2 \), if \( t_2 > t_1 \) then the following inequality must hold:

\[
S_{t_2} - S_{t_1} \geq t_2 - t_1.
\]

Proof: Let us write \( t_2 = t_1 + n \) for some natural \( n \) and prove the lemma by mathematical induction on \( n \). For \( n = 1 \) the lemma holds due to Equation (22). Assuming that the lemma holds for some \( n \), we can easily conclude that the lemma holds for \( n + 1 \) as well.

Theorem 3

For any point \( a \in \Omega \), the time period between two successive visits of the robot is bounded from above by

\[
2n \left( \left\lceil \frac{d}{r} \right\rceil + 1 \right).
\]

Proof: If we show that after a sufficient time period the pheromone level changes at all locations in the domain \( \Omega \), we can obviously be sure that all points were re-visited by the robot during this time period. Let us look at time instance \( t_s \) when the robot covers our point of interest \( a \). We denote by \( \sigma_{\text{max}}(t_s) \) the maximal pheromone level over \( \Omega \) at that time. If we show that at some time instance \( t_e \) the minimal pheromone level denoted by \( \sigma_{\text{min}}(t_e) \) becomes greater than the maximal value that was at time \( t_s \):

\[
\sigma_{\text{min}}(t_e) > \sigma_{\text{max}}(t_s),
\]

then we can easily conclude that during the time period \( t_e - t_s \) the pheromone level changed at all points and thus all points (including \( a \)) were re-covered by the robot. Let us examine \( S_{t_e} \) and \( S_{t_s} \) as defined in the Equation (21). On the one hand:

\[
S_{t_s} = \sum_{i} n m_{t_s}^i - \sigma(p_{t_s}, t_s) \geq \sum_{i} n m_{t_s}^i \geq \sum_{i} \sigma_{\text{min}}(t_s) = n \sigma_{\text{min}}(t_s)
\]

According to Lemma 2

\[
\sigma_{\text{min}}(t_s) \geq \sigma_{\text{max}}(t_s) + \left\lceil \frac{d}{r} \right\rceil.
\]

Combining Equations (33) and (34) we get

\[
S_{t_s} \geq n \left( \sigma_{\text{max}}(t_s) - \left\lceil \frac{d}{r} \right\rceil \right).
\]

On the other hand we want to know the time instance \( t_e \) that guarantees that \( \sigma_{\text{min}}(t_e) \geq \sigma_{\text{max}}(t_s) + 1 \). Instead of estimating \( t_e \) directly from \( \sigma_{\text{min}}(t_e) \), we shall look for \( t_e \) that guarantees the existence of \( \sigma \) value greater than or equal to \( \sigma_{\text{max}}(t_s) + 1 + \left\lceil \frac{d}{r} \right\rceil \), which guarantees by Lemma 2 that \( \sigma_{\text{min}}(t_e) \geq \sigma_{\text{max}}(t_s) + 1 \). Now, in the same way as the proof of Theorem 1, we can say that once \( S_{t_e} \geq n(\sigma_{\text{max}}(t_s) + 1 + \left\lceil \frac{d}{r} \right\rceil) \), we have \( \sigma_{\text{min}}(t_e) \geq \sigma_{\text{max}}(t_s) + 1 \). Thus we have

\[
S_{t_e} - S_{t_s} \leq n \left( \sigma_{\text{max}}(t_s) + 1 + \left\lceil \frac{d}{r} \right\rceil + 1 \right) - \left( \sigma_{\text{max}}(t_s) - \left\lceil \frac{d}{r} \right\rceil \right) = 2n \left( \left\lceil \frac{d}{r} \right\rceil + 1 \right).
\]

According to Lemma 5 we have

\[
t_e - t_s \leq S_{t_e} - S_{t_s} \leq 2n \left( \left\lceil \frac{d}{r} \right\rceil + 1 \right),
\]

which completes the proof.

We now make two important observations. First, the upper time bound between two successive visits by the robot does not depend on the current pheromone level distribution. This is determined completely by the geometric parameters of the problem: \( r, n, \) and \( d \). Second, we observe that this time limit is twice as long as the time period needed for complete coverage. This situation is quite intuitive. Indeed, observe the pheromone level along some path in \( \Omega \), such as the one shown in Figure 6a. In this case the robot may start by “filling” the hollow area on the right until it becomes a hill and only then covers the left-hand part, which used to be a summit point and has now became the lowest point in the pheromone level profile as shown in Figure 6b.
B. Noise Immunity

So far we always assumed that there is no noise in the input, i.e., the robot starts with a domain that does not contain any pheromone marks. Unfortunately, in real life such a clean environment is not always available. For example, spurious pheromone marks may arise as a result of previous attempts to explore the domain by similar robots, which might have used other algorithms and thus the initial pheromone level distribution does not necessarily obey the proximity principle. In general we assume that the initial pheromone level distribution is given by some function

\[ N : \Omega \rightarrow Z^* , \]  

(38)

where \( Z^* \) denotes the set of non-negative integers. As a short digression we have to note that such initial pheromone marks pose a severe problem to all trail-based algorithms. The reason is that such algorithms, for the sake of efficiency, do not get close to their own trails and thus any initial pheromone marks would be interpreted as trails, resulting in uncovered areas around such marks. The result may be even worse if such false trails split the domain into several disconnected parts, in this case the robot will not be able to exit the part where it was located initially. Our algorithm, on the contrary, can easily overcome this problem as we prove below. Actual covering times in presence of noise are demonstrated in Section VI-C.

Let us start with several lemmas.

**Lemma 6**

Immediately after point \( a \in \Omega \) has been marked by the robot for any point \( b \in \Omega \) such that \( ||a - b|| \leq r \) we have:

\[ \sigma(b, t + 1) \geq \sigma(a, t + 1) - 1 , \]  

(39)

where \( t \) denotes the time instance when the new pheromone level was assigned to \( a \).

**PROOF:** Since the pheromone value of \( a \) changes during the \( t \)th step we conclude that \( a \in D(r, p_t) \), where \( p_t \) denotes robot’s location at time \( t \). We know also that \( ||a - b|| \leq r \) and thus \( b \) either belongs to \( D(r, p_t) \) or to \( R(r, 2r, p_t) \). Hence there are two possible scenarios: either \( b \) belongs to \( D(r, p_t) \) or \( b \) belongs to \( R(r, 2r, p_t) \). In the former case \( \sigma(b, t + 1) = \sigma(a, t + 1) \) since the algorithm assigns the same value to all points in \( D(r, p_t) \) and the lemma clearly holds. In the latter case \( (b \in R(r, 2r, p_t)) \) we recall that the algorithm seeks for the minimal pheromone value inside \( R(r, 2r, p_t) \), say attained at some point \( x \) and set new pheromone level inside \( D(r, p_t) \) to be equal to \( \sigma(x, t + 1) \). Hence, we have:

\[ \begin{cases} 
\sigma(b, t + 1) = \sigma(b, t) \geq \sigma(x, t) \\
\sigma(a, t + 1) = \sigma(x, t) + 1 
\end{cases} \]  

(40)

This leads us again to the conclusion

\[ \sigma(b, t + 1) \geq \sigma(a, t + 1) - 1. \]  

(41)

Hence the lemma is proved.

Though this lemma resembles Lemma 1, it does not guarantee that the proximity principle is obeyed in noisy environment. We demonstrate a stronger result later in Lemma 8.

At this moment we shall prove that the pheromone level at marked points, i.e., pheromone left by the robot never decreases.

**Lemma 7**

Pheromone level values at any marked point form a non-decreasing sequence; that is

\[ \forall t \forall u \in \Omega : \sigma(u, t + 1) \geq \sigma(u, t) \]

given that \( u \) was marked by the robot at the time prior to \( t \).

**PROOF:** Let us assume the contrary, i.e., for some time instance \( t \) and for some point \( u \in \Omega \) we have:

\[ \sigma(u, t) < \sigma(u, t + 1) \]  

(42)

Let \( t \) be the first such time instance. As usual we denote by \( p_t \) the robot’s location at time \( t \). According to the MAW algorithm the robot seeks to the minimal pheromone level in \( R(r, 2r, p_t) \). Say this minimal level is attained at some point \( x \in R(r, 2r, p_t) \). Since \( u \) changes its value during the \( t \)th step we conclude that

\[ \sigma(p_t, t) \leq \sigma(x, t) \]  

(43)

Otherwise no change happens, according to the MAW algorithm. Since all points in \( D(r, p_t) \) get the same value in the marking step we conclude:

\[ \sigma(u, t + 1) = \sigma(p_t, t + 1) = \sigma(x, t) + 1 \geq \sigma(p_t, t) + 1 \]  

(44)

Moreover, according to our assumption:

\[ \sigma(u, t) > \sigma(u, t + 1) \Rightarrow \sigma(u, t) \geq \sigma(p_t) + 2 \]  

(45)

Thus, if we assume that the pheromone level at some point \( u \) decreases at time instance \( t \) Equation (45) must hold. Showing that this inequality is wrong we actually get a contradiction to the assumption and thus prove the lemma. Let us look at the time instance \( t_u \) when the current pheromone level of \( u \) \( (\sigma(u, t)) \) was set. According to Lemma 6:

\[ \sigma(p_t, t_u + 1) \geq \sigma(u, t_u + 1) - 1 \]  

(46)

Since \( t_u < t \) and \( t \) was chosen to be the first time when the pheromone level at any point in \( \Omega \) decreases we conclude that

\[ \sigma(p_t, t) \geq \sigma(p_t, t_u + 1) \]  

(47)
Substituting it into Equation (46) we get
\[
\sigma(p_t, t) \geq \sigma(u, t) - 1.
\] (48)

And thus, the inequality in Equation (45) does not hold. This contradiction completes the proof of Lemma 7.

**Lemma 8**

If, at some time instance \( t \), both \( a \) and \( b \) had been marked by the robot their pheromone levels obey the proximity principle, i.e.,

\[
\text{if } |a - b| \leq r, \text{ then } |\sigma(a, t) - \sigma(b, t)| \leq 1
\] (49)

**Proof:** Since both \( a \) and \( b \) had been marked prior to time instance \( t \) there are exist time instances \( t_a \) and \( t_b \) when \( a \) and \( b \) got their current pheromone levels accordingly. Applying Lemma 6 we get the following two equations:

\[
\begin{aligned}
\{ & \sigma(b, t_a + 1) \geq \sigma(a, t_a + 1) - 1 \\
& \sigma(a, t_b + 1) \geq \sigma(b, t_b + 1) - 1.
\end{aligned}
\] (50)

Or, substituting \( \sigma(b, t_b + 1) = \sigma(b, t) \) and \( \sigma(a, t_a + 1) = \sigma(a, t) \)

\[
\begin{aligned}
\{ & \sigma(b, t_a + 1) \geq \sigma(a, t) - 1 \\
& \sigma(a, t_b + 1) \geq \sigma(b, t) - 1.
\end{aligned}
\] (51)

Since \( t > t_a \) and \( t > t_b \) we can apply Lemma 7:

\[
\begin{aligned}
& \sigma(b, t) \geq \sigma(b, t_a + 1) \\
& \sigma(a, t) \geq \sigma(a, t_b + 1).
\end{aligned}
\] (52)

Substituting it into Equation 51 we get

\[
\begin{aligned}
& \sigma(b, t) \geq \sigma(a, t) - 1 \\
& \sigma(a, t) \geq \sigma(b, t) - 1.
\end{aligned}
\] (53)

Which means

\[
|\sigma(a, t) - \sigma(b, t)| \leq 1
\] (54)

Hence the lemma is proved.

**Lemma 9**

The maximal pheromone level tends to infinity as \( t \) goes to infinity.

**Proof:** The proof is identical to the one in Lemma 4. We again introduce a virtual tessellation of domain \( \Omega \) into \( n \) cells so that every such cell can be inscribed into a circle of diameter less than \( r \). And, as before, we look at the sum:

\[
S_t = \sum_{i=1}^{n} m_t^i - \sigma(p_t, t)
\] (55)

The only difference that this time \( m_t^j \) denotes the minimal marker value that was set by the robot and not as a result of the noise. As in Lemma 1 we get

\[ S_t \geq t \ \forall t. \] (56)

Thus the lemma is proved.

**Theorem 4**

For any initial noise profile \( N : \Omega \rightarrow \mathbb{Z}^+ \), the domain \( \Omega \) will be covered after \( n(M_N - m_N + \lceil \frac{n}{r} \rceil + 1) \) steps, where \( M_N \) and \( m_N \) denote the maximal and the minimal pheromone levels at time \( t = 0 \), respectively.

**Proof:** Let us denote by \( M_N \) and by \( m_n \), respectively, the maximal and the minimal values of the initial pheromone level given by \( N \). According to Lemma 12 the maximal pheromone level in \( \Omega \) grows and will eventually reach the value of \( M_N + \lceil \frac{n}{r} \rceil + 1 \). We claim that at this moment the whole domain is covered by the robot. Indeed, let us look at some point \( a \in \Omega \) that got at step \( t \) this pheromone level. According to Lemma 6 for any point \( b \in \Omega \) such that \( |a - b| \leq r \) we have

\[
\sigma(b, t + 1) \geq \sigma(a, t + 1) - 1 = M_N + \lceil \frac{n}{r} \rceil.
\] (57)

Since \( M_N + \lceil \frac{n}{r} \rceil \geq M_N \) we conclude that all such points are covered. In the same manner we get that all points whose distance from \( a \) is less than or equal to \( 2r \) are also covered. And so on, maximal distance between points in \( \Omega \) is bounded by \( d \), thus one any point reaches pheromone level of \( M_N + \lceil \frac{n}{r} \rceil + 1 \) we assure that the minimal possible pheromone level is \( M_N + 1 \) which means that the whole domain \( \Omega \) has been covered. The time needed to cover the domain is

\[
t_{\text{cover}} = n \left( M_N - m_N + \lceil \frac{n}{r} \rceil \right) + 1,
\] (58)

where \( m_N \) denotes minimal pheromone level at time \( t = 0 \).

Since the whole domain is covered we can guarantee, by Lemma 8 the proximity principle is obeyed by any two points in \( \Omega \) and further repetitive coverage is governed by Theorem 3.

**C. Multiple Robots**

As a natural extension we would like to analyze how the MAW algorithm can be applied to multi-robot environments. First of all, we must address problems such as collisions both between the robots themselves (if we deal with physical robots and not programs) and between different pheromone levels when two (or more) robots try to mark the same point in the domain.

At the moment we shall assume that the clock phases of all robots are slightly different so that no two robots are active at the same time. Thus each robot sees other robots as regular stationary obstacles and acts accordingly. With this approach we also have no problem of simultaneous attempts to set (probably different) pheromone levels at particular location by multiple robots, since only one robot is active at any given time.

Let us find the upper bound for complete coverage provided we have \( n \) robots. Using the same notation as in Equation (21) we have:

\[
S_t = \sum_{i=1}^{n} m_t^i - \sum_{j=1}^{n} \sigma(p_j^t, t),
\] (59)

where \( p_j^t \) denotes the location of the \( j \)-th robot at time \( t \). Using exactly the same reasoning as before, we again obtain:

\[
S_{t+1} > S_t,
\] (60)

and consequently

\[
S_t \geq t,
\] (61)

which leads us to the same upper bound we got for a single robot. Hence adding more robots does not necessarily guarantees better coverage time. However simulations (see Section VI) demonstrate that there is a substantial improvement when we use more robots.
D. Generalization for Other Metrics

Until now we always assumed that the domain $\Omega$ is flat two-dimensional domain, and the usual Euclidean notion of the distance was used. Nevertheless, it is easy to verify that all the proofs remain valid if we change the Euclidean, often referred to as $L_2$ distance to another valid metric. For example, we could use $L_1$ distance or, alternatively, the $L_\infty$ distance which is particularly suitable for computer simulations. Of course each choice of the metric changes the form of the robot’s effector. Three different forms, shown if Figures 7a, 7b, and 7c correspond to $L_1$, $L_2$, and $L_\infty$ metrics accordingly. Moreover we are not limited to 2-Dimensional spaces as the results remain valid for higher dimensions, e.g., we can use the same algorithm for covering 3D volumes assuming the robot’s effector is a ball of radius $r$ or, probably, a regular octahedron or a cube if we choose to work with $L_1$ or $L_\infty$ metrics respectively.

VI. SIMULATIONS AND EXPERIMENTS

A. General Notes

We used $L_\infty$ metric in our experiments because the square shape of the effector and the sensing area that correspond to this metric is particularly suitable for computer simulations. In experiments with noise the robots were forced to start at non-noisy location, i.e., at locations with minimal pheromone level at time $t = 0$. Additionally, in all experiments the robots were modeled as points and multiple robots were allowed to occupy the same location. We always measured the number of time steps until the robots covered the domain for the first time, averaged over 100 runs. Experiments were conducted on the domains shown in Figure 8. All domains are of size 100 × 100 pixels and marking radius in all experiments was set to 3, i.e., each step robot marks a square of 5 × 5 pixels.

B. Comparing MAW to other algorithms

In this experiment we studied performance of three different algorithm: MAW, MAC [19], and Random Walk. All algorithms used the same square effector of size $5 \times 5$ pixels; additionally, the steps of the Random Walk algorithm were restricted to be in interval $[r, 2r]$ just like the steps in the MAW algorithm.

As we can see the MAW algorithm is a clear winner when we use three or more robots. For fewer robots the MAC algorithm performs better on complex domains. Note that the MAW algorithm in general performs better than the theoretical upper bound we got in Section IV-B (recall that the bound is quadratic and we got results better than that of MAC, which is asymptotically linear). Cover time of the Random Walk was omitted from Figures 9b and 9c because the values were so big that the difference between the MAC and the MAW algorithms became invisible on this scale.
Note that our upper bound on coverage time is quadratic, while the above results suggest that the actual coverage time is linear. We can demonstrate the predicted quadratic coverage time by tailoring specific tie breaking rules for specific domains. For example we can use a domain comprised of $n$ loops as shown in Figure 10. For this domain and specific tie breaking rules we can obtain quadratic coverage time for MAW algorithm, while the MAC algorithm still demonstrates linear coverage time. These results are shown in Figure 11.

C. MAW in noisy environments

In this section we present the results of our simulation of MAW algorithm in presence of noise. In the first scenario we ran one robot on the Domain A, each time changing the amount of noisy pixels. Noise values are uniformly distributed in interval $[1, 10]$, i.e., given that 60 percent of the pixels are noisy there are about 6 percent that got value of 1, 6 percent that got value of 2 and so on. Example of such noise profile with 60 percent noisy pixels is shown in Figure 12a. In Figure 12 we demonstrate the cover time as a function of the amount of noisy pixels in this scenario.

In another scenario we chose to explore the influence of constant noise values on the performance of the algorithm. This time noise values in each experiment were constant and again randomly distributed in the space. Example of such noise profile for noise value of 10 and 30 percent noisy pixels is shown in Figure 13a.

We ran a series of experiments with the noise values of 10, 20, 30, 40, and 50. The results of the cover time versus the percentage of noisy pixels is shown in Figure 13b. As we can see the value of the noise does not play any role in this scenario, at least in the limits we used here, from 10 to 50. This is probably due to the nature of the algorithm that knows to discard high pheromone values in presence of lower values. To check this we conducted another experiment, that is similar to this one, however noise is this scenario occupies a compact space in the domain, i.e., given that there 40 percent of noisy pixels we form a plateau of noise as shown in Figure 14a. In this case the influence of the noise value is quite pronounced as one can expect. See Figure 14b for covering time versus noisy pixel percentage for noise values of 10, 20, 30, 40, and 50. As our experiments demonstrate the MAW algorithm has little sensitivity to “non-compact” distribution of the noise. Note that noise does not affect the Random Walk on the one hand and it destroys completely the MAC algorithm on the other hand, making it unable to cover the domain completely.

VII. CONCLUSIONS

This work has two major contributions. First, we presented a new ant-inspired algorithm for continuous domain covering. We provided a formal proof of complete coverage and upper time bounds for complete coverage and the time interval between two successive visits of the robot. We also proved that the algorithm is immune to pheromone noise in the environment. A formal proof was provided for multi-age)nt or multi-agent coverage under the assumptions that the robots have different clock phases. Second, a new way of performance analysis was suggested, that provided some bounds on possible coverage time of any such mark and walk algorithm.

REFERENCES


Fig. 13. MAW time in noisy environment 2

Fig. 14. MAW time in noisy environment 3