Optimal Pheromone Utilization

Yehuda Afek, Roman Kecher and Moshe Sulamy
{afek,roman.kecher,moshe.sulamy}@cs.tau.ac.il

Abstract

The minimal amount of pheromones necessary and sufficient to deterministically find a treasure (food) by a colony of ants, each modeled by either a Finite State Machine or a Turing Machine, is considered. The k mobile agents (ants), initially located at the origin (nest) of an infinite grid, communicate only through pheromones to perform a collaborative search for an adversarially hidden treasure at an unknown distance D. We begin by proving a tight lower bound of $\Omega(D)$ on the number of pheromones required by a FSM to even complete the search, and continue to reduce the lower bound to $\Omega(k)$ for the stronger ants modeled as TM. Matching optimal algorithms, in terms of run time as well as pheromone usage, are provided both for synchronous and asynchronous models in each case.

1 Introduction

Performing mostly simple and local computations, ants solve complicated problems with great collaboration. Such collaborative work, with limited communication and computational power, is at the heart of distributed computing. By studying and understanding the behavior of ants from a distributed computing perspective, progress and insights can be made for both the distributed computing and biology fields.

In this paper, we analyze the minimum number of pheromones required by ants to find a treasure—a food item. This is the basis of the *Ants Nearby Treasure Search (ANTS)* problem, first presented by Feinerman et al. [7]. In the original ANTS problem, k ants start at the origin (i.e., the *nest*) of an infinite grid, and need to find a treasure located at an unknown distance D. The ants are randomized mobile Turing machines, and cannot communicate after leaving the nest. Once any ant steps onto the grid point containing the treasure, the treasure is found and the algorithm terminates. A trivial lower bound on the time required to find the treasure is $\Omega(D + D^2/k)$.

Our model is based on the model presented by Lenzen et al. [9], where ants mark *each* visited grid point by a pheromone. Pheromones are a biological resource which might be limited in quantity or costly to be used lightly. In our model, we allow ants to *choose* whether or not to leave a pheromone upon visiting a grid point, thus analyzing the minimal amount of pheromones required to find the treasure for various computational models is a natural research direction.

1.1 Contributions

We present lower bounds of $\Omega(D)$ and $\Omega(k)$ on the minimal amount of pheromones required by k ants in order to find a treasure at an unknown distance D, for ants that are modeled either as a *Finite State Machine* or a *Turing Machine*. We then present matching upper bounds for both computational models, in both the synchronous and asynchronous models.

Remark 1. The number of pheromones used during an asynchronous search is potentially unlimited, e.g., the scheduler might stop scheduling any ant right before it reaches the treasure and keep scheduling the rest. Therefore, we consider the maximal number of pheromones which is *required* in order to ensure that the treasure is found, given that every ant is scheduled infinitely often.

Remark 2. In the synchronous case we assume the existence of an *emission scheme*, an idea introduced in [4,5], which ensures that no two ants leave the nest at the same round. Notice that, without an emission scheme, there is no deterministic way of breaking the inherent symmetry of the synchronous model without resorting to randomization. Let $t_e(k)$ denote the time to emit k ants.

*This research was supported by the Israel Science Foundation (grant 6693/11).

Remark 3. To break the symmetry in the asynchronous case ants are capable of sensing and emitting a pheromone in one atomic operation, like a test-and-set operation (see [2] for details).

To the best of our knowledge, no such lower bounds have been previously analyzed; the only relevant algorithm, presented in [9], uses $O(D^2)$ pheromones to find the treasure with ants modeled as FSM with extra capabilities.

For the FSM model, we prove that $\Omega(D)$ pheromones are required to find the treasure. We provide optimal algorithms that, for k ants, find the treasure in $O(t_e(k) + D + D^2/k)$ and $O(D + D^2/k)$ rounds in the synchronous and asynchronous models, respectively, while using only $O(D)$ pheromones.

For the TM model, no pheromones are required to find the treasure; however, we prove that $\Omega(k)$ pheromones are required in order to find the treasure in optimal time, in $O(D+D^2/k)$ rounds. We provide an optimal algorithm that, for k ants, finds the treasure in $O(D+D^2/k)$ rounds in the asynchronous model, while using $O(k)$ pheromones. We also show that the same algorithm is suitable for the synchronous case, bearing an extra cost of $t_e(k)$ rounds.

1.2 Related work

The *Ants Nearby Treasure Search (ANTS)* problem was introduced by Feinerman, Korman, Lotker, and Sereni [7], showing that matching the lower bound requires knowledge or a constant approximation of k.

In [6], lower bounds on the memory required by the ants to find the treasure are given. In [4, 5], ants modeled by a *Finite State Machine* are discussed, in both synchronous [4] and asynchronous [5] models. Emek et al. take a computability point of view [3], by analyzing the minimal number of ants required to find the treasure within finite time. In [8], the *Selection Complexity* is studied, a measure of how likely a given algorithmic strategy is to arise in nature.

Pheromones were introduced to ANTS by Lenzen et al. [9], who assumed that ants mark *each* visited grid cell. Our model differs in that the ants are allowed to *choose* whether or not to emit pheromones.

2 Ants as Deterministic Finite State Machines

In this section, all ants are implemented by a deterministic Finite State Machines (FSM).

2.1 Lower bound

Theorem 2.1. *In the Finite State Machine computational model, an ant has to emit $\Omega(D)$ pheromones to search every grid cell at distance $d \leq D$.*

Proof. Assume by contradiction that there exists FSM (ant) A capable of finding a treasure at any distance $D \in \mathbb{N}$ using $o(D)$ pheromones. Let S be the number of states in A.

Define *layer* n as the set of grid cells (x, y) such that $|x| + |y| = n$. There exists a distance D such that there are $S+1$ consecutive layers with no pheromones within the first D layers, otherwise there are always $\Omega(D)$ pheromones overall. From these layers denote the layer furthest away from the origin as layer l.

Consider the first time the ant arrives at layer l. Denote its state at that point as s'. Looking at its path to this point up to S steps back, the ant must have passed state s' once more. Therefore we have a path that starts at some layer $l - i$ $(1 \leq i \leq S)$ with state s' and ends at layer l in the same state, during which the ant has not placed any pheromones nor encountered any. This path therefore must be repeated infinitely from layer l onwards. Since layer l contains more grid cells than all previous layers, the ant does not cover all grid cells with distance $d \leq D$. $\square$

Corollary 2.2. *A lower bound on the number of pheromones emitted during an exhaustive search for a treasure which is located at an unknown distance D from the origin in the Finite State Machine computational model is $\Omega(D)$, regardless of the number of ants taking part in the search.*

2.2 Synchronous Finite State Machines

We use north, east, south and west rays of pheromones, forming a plus sign with the nest in the center (using pheromones as guides, an idea inspired by [4, 5]).

The ants start their exploration from the northern ray, and perform pairs of east-south "zigzag" moves (a move east followed by south) until encountering a pheromone at the eastern ray. They then emit two pheromones to extend the east ray and switch to west-south "zigzag" moves, to reach and extend the south ray. Continuing this process, the ants reach the western ray and then the northern ray, at which point an ant moves north to find an odd-indexed pheromone-free grid cell, indicating its next layer to explore.

An ant that was just emitted from the nest and another ant might reach a cell on the north ray simultaneously, both starting a layer at the same time. We tackle this issue by partitioning the ants into two logical groups. Newbie ants, which have yet to explore their first layer, and non-newbie ants. An ant is considered a newbie when it is emitted, and becomes non-newbie after exploring its first layer.

A non-newbie ant travels north until it encounters the first odd-indexed pheromone-free grid cell. It marks it with a pheromone and also emits a pheromone on the first grid cell it visits during its exploration phase (directly to the east). This pheromone is later checked by newbie ants; a newbie ant looks for the first odd-indexed pheromone-free grid cell on the north ray, exactly as non-newbies, but after marking such a cell it then makes one step to the east and waits a single round to make sure that it has not collided with a non-newbie ant. If it discovers a pheromone, the newbie ant just proceeds to search for the next odd-indexed free layer. Notice that in order to keep the gaps introduced by the emission scheme, newbie ants perform these steps (of traveling to the eastern grid cell and back) for every odd-indexed cell on the north ray which *does* contain a pheromone, as well.

Theorem 2.3. *The algorithm finds the treasure in $O(t_e(k) + D + D^2/k)$ rounds and utilizes $O(D)$ pheromones, hence it is optimal given a matching emission scheme.*

2.3 Asynchronous Finite State Machines

This algorithm is a variation of the synchronous algorithm. No ant can make any assumptions on the progress of other ants, and hence cannot rely on pheromones emitted by other ants for navigation clues. We overcome this problem by visiting the rays in a different order, starting a layer exploration from the last ray, so that every ant only relies on pheromones that were already emitted when it starts the exploration.

The colony constructs four rays of pheromones which form a plus shape with the nest at its center. The ants start by extending the east, south and west rays, and then begin their exploration phase from the north ray. Upon finishing exploring a layer, every ant gets back to the nest to repeat the process. This procedure assures that every ant starts the search in a layer which already has pheromones in all the rays, ensuring it is able to finish the exploration successfully. Notice that the nest remains free of pheromones throughout the algorithm in order for ants to easily identify it.

Theorem 2.4. *The algorithm terminates successfully in $O(D + D^2/k)$ rounds and the total number of emitted pheromones is $O(D)$, which make it optimal.*

3 Ants as Turing Machines

A stronger computational model, of Turing Machines, is employed to further reduce the number of pheromones. Notice that the algorithms suggested here require only $O(log(D))$ bits of information.

3.1 Lower bound

A single TM ant can find the treasure with no pheromones in $O(D^2)$ time, by performing a spiral shaped search. However, for an optimal running time search, an $\Omega(k)$ lower bound on the number of pheromones required is given:

Theorem 3.1. *A distributed search performed by k ants for a treasure at an unknown distance $D \in \mathbb{N}$ in $O(D + D^2/k)$ time requires emitting $\Omega(k)$ pheromones, assuming ants do not know k in advance.*

Proof. Let us assume that k ants perform a search with $o(k)$ pheromones. Therefore there is at least one ant which does not place any pheromones. In the asynchronous model we can duplicate that ant and schedule the duplicates one right after the other, thus making them perform exactly the same work, contradicting the optimal runtime assumption. In the synchronous model, let us schedule the ants such that each ant is emitted right after the last pheromone is placed by all previously emitted ants. Let p be the first emitted ant that does not place any pheromones (the number of rounds until p is emitted is constant). Since all pheromones were already placed, any ant emitted after p has the same view as p, thus making them perform exactly the same work, again contradicting the optimal runtime assumption (see [2] for details). $\square$

3.2 Algorithm

We provide the same algorithm for both the synchronous and asynchronous scheduling models. The main idea is a static partition of the search space. We use pheromones to provide each ant: (i) a unique id (ii) an estimate on the total number of participating ants. Each ant keeps such an estimate, and updates it after each exploration of a layer. We define a layer differently in this section; *layer* l contains the set of grid cells (x, y) where $\max(|x|, |y|) = l$. An ant with id i and estimation k_i explores layers $l_i \equiv i \pmod{k_i}$.

If an ant learns that it has to update its current estimate of the total number of ants, its search space must be restarted. Otherwise, the exploring ants would miss a few layers (see [2] for details).

Theorem 3.2. *The algorithm completes in $O(t_e(k) + D + D^2/k)$ rounds for the synchronous case, and $O(D + D^2/k)$ rounds for the asynchronous case, emitting $O(k)$ pheromones in both cases.*

4 Conclusions and future work

We have presented different approaches to solve the ANTS problem with pheromones. The main focus has been on the optimal use of pheromones, as it is a biological resource which might only be scarcely available. The upper bound results are summarized in Table 1.

We have presented a lower bound on the number of pheromones required to solve ANTS with Finite State Machines, as well as Turing Machines, and provided matching as fast as possible algorithms under both the synchronous and asynchronous models.

	Synchronous Model		Asynchronous Model	
FSM	Runtime:	$O(t_e(k) + D + D^2/k)$	Runtime:	$O(D + D^2/k)$
	Pheromones:	$O(D)$	Pheromones:	$O(D)$
TM	Runtime:	$O(t_e(k) + D + D^2/k)$	Runtime:	$O(D + D^2/k)$
	Pheromones:	$O(k)$	Pheromones:	$O(k)$

Table 1: Summary of various solutions for the ANTS problem.

Open Questions In the synchronous case, we relied on an emission scheme: no two ants leave the nest at the same time. An open question is devising a randomized emission scheme that utilizes pheromones.

A major part of real ant foraging process is what happens once the food (treasure) is actually found; commonly, the discovering ant has to find its way back to the nest and inform other ants of the discovery in order to get assistance in carrying the food back, for instance.

Other topics for future research are: coping with obstacles, ants failures (i.e., due to getting eaten), same cooperative search but from different initial nest locations, or what happens if the ants are capable of utilizing different flavors of pheromones (perhaps even with limited lifespan) or can employ pheromones together with randomization in their decision making process.

References

[1] Y. Afek, N. Alon, O. Barad, E. Hornstein, N. Barkai, and Z. Bar-joseph. A biological solution to a fundamental distributed computing problem. In *Science*, volume 331, pages 183–185, 2011.

[2] Y. Afek, R. Kecher, and M. Sulamy. Optimal pheromone utilization. Link: `http://www.cs.tau.ac.il/~afek/ants.pdf`.

[3] Y. Emek, T. Langner, D. Stolz, J. Uitto, and R. Wattenhofer. How many ants does it take to find the food? In *SIROCCO*, pages 263–278, 2014.

[4] Y. Emek, T. Langner, J. Uitto, and R. Wattenhofer. Ants: Mobile finite state machines. *CoRR*, abs/1311.3062, 2013.

[5] Y. Emek, T. Langner, J. Uitto, and R. Wattenhofer. Solving the ants problem with asynchronous finite state machines. In *ICALP (2)*, pages 471–482, 2014.

[6] O. Feinerman and A. Korman. Memory lower bounds for randomized collaborative search and implications for biology. In *DISC*, pages 61–75, 2012.

[7] O. Feinerman, A. Korman, Z. Lotker, and J.-S. Sereni. Collaborative search on the plane without communication. In *PODC*, pages 77–86, 2012.

[8] C. Lenzen, N. A. Lynch, C. C. Newport, and T. Radeva. Trade-offs between selection complexity and performance when searching the plane without communication. In *PODC*, pages 252–261, 2014.

[9] C. Lenzen and T. Radeva. The power of pheromones in ant foraging. *BDA*, 2013.

A Appendix

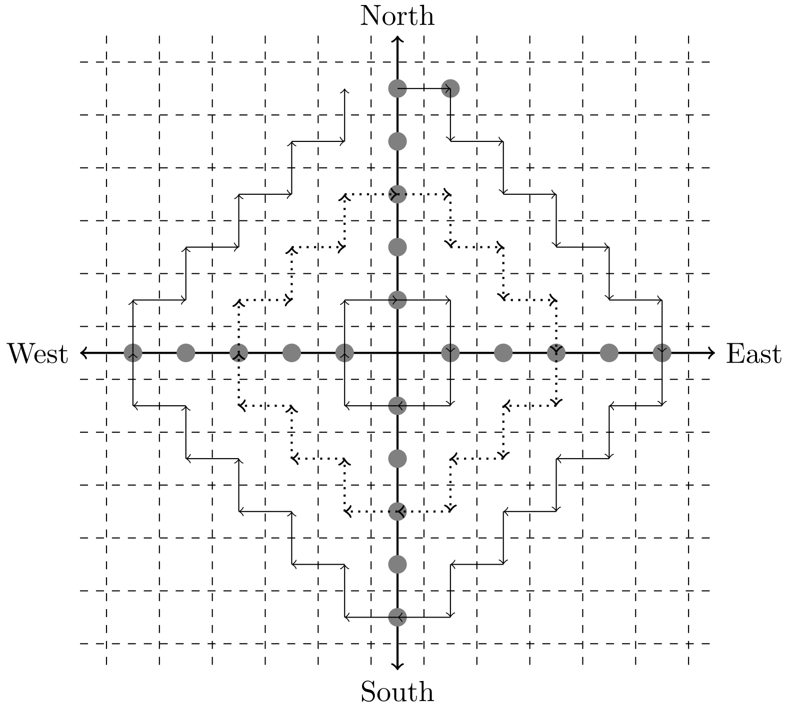

Figure 1: Two synchronous FSM ants searching for food. Circles indicate pheromones.

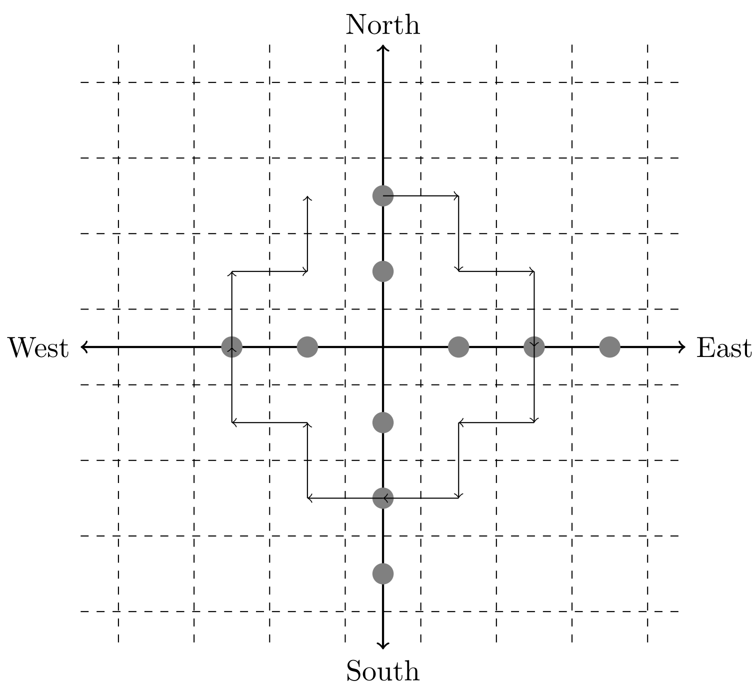

Figure 2: Asynchronous FSM ant searching for food. Circles indicate pheromones.

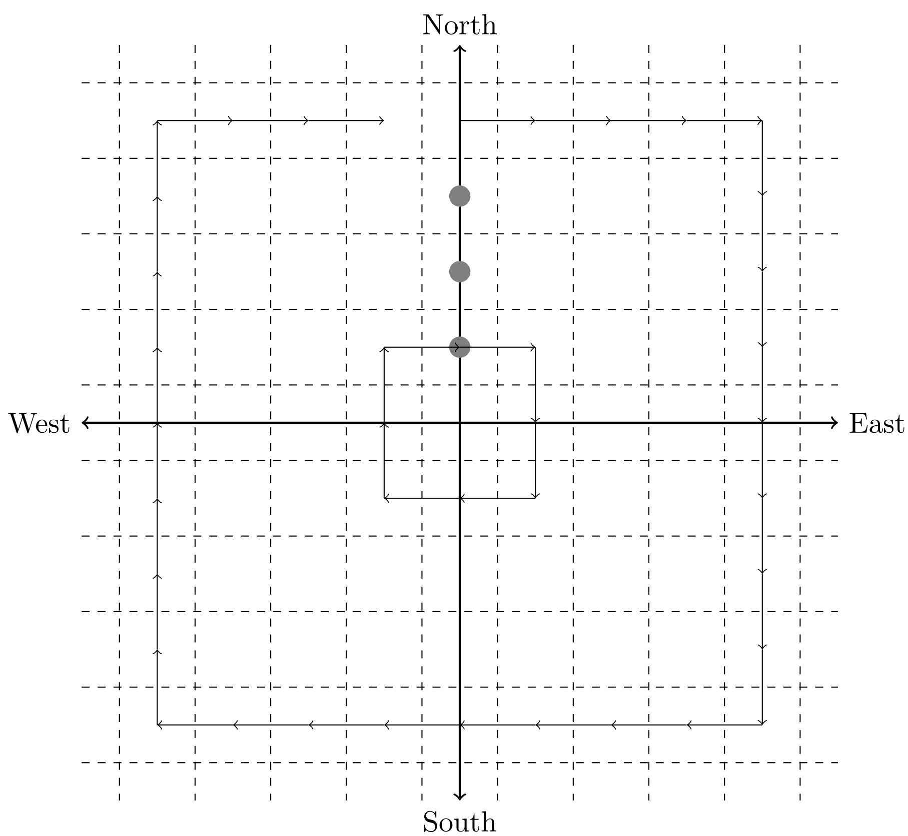

Figure 3: (A)Synchronous TM ant searching for food. First ant of a total of three.