

# How Many Robots? Group Size and Efficiency in Collective Search Tasks

Adam T. Hayes

Collective Robotics Group  
136-93 California Institute of Technology, Pasadena CA 91125  
athayes@caltech.edu  
www.coro.caltech.edu/People/athayes/athayes.html

**Keywords:** Collective Autonomous Robotics, Distributed Exploration

**Abstract.** This paper presents a quantitative analysis of the tradeoffs between group size and efficiency in collective search tasks that considers both the time-sensitive nature of search completion and the system operating cost. First, the search task is defined and a performance metric is presented that can account for all of the costs associated with the task. Next, for both random and coordinated search strategies, analytical expressions are derived that can be used to predict optimal system performance bounds given a particular task description, and the performance benefit of using coordinated search is shown to be dependent on the relative values of the different cost components. Finally, an embodied computer simulation is used to support the analytical results, suggesting that the assumptions involved in their derivation are sound.

## 1 Introduction

Search tasks, because they submit well to parallelization, are an ideal application for multi-agent systems. Search is a well studied problem (for a review, see [1]), and there has been a significant amount of investigation into the efficiency tradeoffs between random and coordinated search strategies [2]. However, how to assess the performance of multi-agent search systems is still an open problem. Some researchers take into account only energy used [2], while others consider only the time required until completion [3] when analyzing the performance of multi-agent systems on similar search tasks. Clearly, the performance metric used must be appropriate for the task being studied, but there is reason to believe that a more complete cost metric might offer further insight into the design tradeoffs present and aid in the comparison of results across research groups.

## 2 Search Task Description

The search task examined in this paper can be described as follows: a group of  $N$  agents each having a sensor radius  $r$  must locate a single target contained

within an enclosed 2-D arena. For simplicity consider this arena to be a square of length  $L$ , with  $L \gg r$  so that the agents are likely to disperse throughout the arena before the target is found. To ensure that the agents do not begin with full coverage of the arena (thus driving the search time to 0), initial agent deployment must be within a single deployment area of radius  $R$ . It is assumed that  $L \gg R$ , although the deployment area may be located anywhere within the arena. Figure 1 shows a schematic of an example task layout.

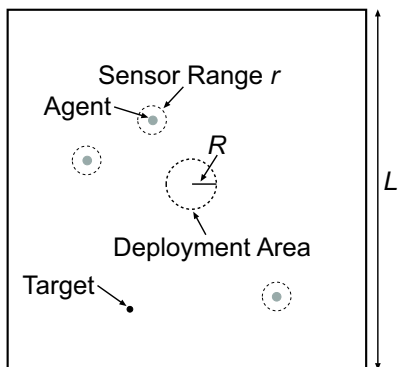


Fig. 1. Example task layout in which  $N = 3$

## 2.1 Performance Metric

Performance on this search task can be measured in terms of  $T$ , the time elapsed before an agent detects the target, and  $D$ , the sum of the distances traveled by each of the agents.  $D$  then correlates to the amount of energy needed for system operation. There are also setup costs that need to be considered in a complete system evaluation. Since these measures are physically independent, a composite metric incorporating a task-specific weighting of these basic factors can be considered. For  $N$  agents:

$$C = \alpha T + \beta D + \gamma N \quad (1)$$

There are three basic cost components.  $\alpha$  is taken to be the cost per unit time of not completing the task,  $\beta$  is the cost per unit distance of running the system, and  $\gamma$  is the initialization cost per agent.  $C$  represents the total cost incurred before the task is completed. By choosing specific values for  $\alpha$ ,  $\beta$ , and  $\gamma$  the appropriate relationship between time required, energy used, and initial cost can be generated for evaluating any particular application.

To simplify the analysis, if the control algorithm used maintains an average speed  $v$  across time, the total distance traveled can be approximated by the time required to complete the task:

$$D = TNv \quad (2)$$

Substituting into equation 1 above,

$$C = \alpha T + \beta TNv + \gamma N \quad (3)$$

Thus, for any given group size, the system cost can be obtained directly from the time required. Although  $C$  is the metric used in the analysis section of this paper, in order to facilitate comparison across environments, it can be normalized by the minimum completion cost to generate a unitless performance metric. The minimum cost is based on the optimum values for the given task ( $T_{MIN}, D_{MIN}$ ) for a single agent with prior knowledge of the source location, as determined from the average distance between starting location and target location as well as maximum agent speed.

$$P = \frac{\alpha T_{MIN} + \beta D_{MIN} + \gamma}{C} \quad (4)$$

This form of  $P$  ensures that for any cost  $\alpha$ ,  $\beta$ , or  $\gamma$  greater than 0, the optimal system will achieve a performance of 1, and any that requires more time, distance, or agents will have a performance less than 1.

### 3 Deriving Performance

The stochastic nature of real systems (e.g. from sensor noise, agent movement, or deployment and target location variation), means that for each trial the cost to complete a search task is drawn from some distribution. For some applications the designer is interested in minimizing the average cost of system operation, and for other tasks the value of interest is a composite of the average cost and its variation. This work focuses on bounding the cost of a given percentage of trials, that is, determining the cost  $C$  which exceeds the cost of some fraction  $S$  of all trials in that particular environment.

Expressions for the optimal cost of random and coordinated search strategies are derived in the following sections. For clarity, a summary of the variables used is provided in Table 1.

#### 3.1 Random Search

In a system performing random search the agents move randomly while searching for the target without any explicit attempt to partition the space amongst agents or avoid searching the same area multiple times. Given that a system has some probability  $g$  of finding the target during a time interval  $t$ , the probability  $S$  that the target is found before some time  $T$  can be expressed as a geometric series:

$$S = \sum_{t=1}^T g(1-g)^{t-1} \quad (5)$$

**Table 1.** Summary of Parameters and Variables

|          |                                |         |                                      |
|----------|--------------------------------|---------|--------------------------------------|
| N        | Number of agents               | r       | Sensor radius                        |
| L        | Arena length                   | R       | Deployment area radius               |
| T        | Time to complete task          | D       | Total distance to complete task      |
| $\alpha$ | Cost of not finishing          | $\beta$ | Cost of operation                    |
| $\gamma$ | Initialization cost per agent  | C       | Total system cost                    |
| v        | Average agent velocity         | P       | System performance measure           |
| S        | Desired performance bound      | g       | Probability of system finding target |
| t        | Time interval                  | k       | Minimum dispersion time              |
| $\eta$   | Sensor detect probability      | p       | Probability of agent finding target  |
| $x^*$    | Optimal value for variable $x$ | U       | Single agent trial search time       |

To solve for  $T$ , the series can be simplified as follows:

$$S - (1 - g)S = g - g(1 - g)^T \quad (6)$$

$$T = \frac{\log(1 - S)}{\log(1 - g)} \quad (7)$$

The above equation describes the time to complete the task based on search success probability and desired performance bounds. To be more accurate, however, a term needs to be added to account for the fact that the agents cannot begin the task with full coverage of the entire search area (because all agents start within the deployment area).

$$T = \frac{\log(1 - S)}{\log(1 - g)} + k \quad (8)$$

The factor  $k$  represents the time required to cover the distance between the deployment area and target point, and serves as a lower bound of the time needed to perform the task (i.e.  $k = T_{MIN}$ ).

$g$  can be decomposed in terms of the number of individual agents  $N$  performing the task, and the probability  $p$  of a single agent scanning the target per time period  $t$ .  $p$  can be approximated using the ratio of the area of scanned per  $t$  to the total area of the arena  $L^2$ . A sensor detect probability  $\eta$ , modeled here as the probability of target detection given that the target enters the sensor range, factors in as well.

$$p = \frac{rv\eta}{L^2} \quad (9)$$

Assuming that the probability of each agent succeeding is fully independent, given  $p$  and a group size of  $N$  agents, the probability  $g$  of the system locating the target during a time period  $t$  can be calculated to be:

$$g = 1 - (1 - p)^N \quad (10)$$

Plugging this value into equation 8:

$$T = \frac{\log(1 - S)}{N \log(1 - p)} + k \quad (11)$$

Now the optimum number of robots, the optimum time, and the optimal cost for a given task can be derived. Beginning with a substitution, let:

$$U = \frac{\log(1 - S)}{\log(1 - p)} \quad (12)$$

$U$  represents the length of time necessary for  $S$  percent of trials using a single agent to locate the target (after the initial dispersion period  $k$ ). Substituting into equation 11:

$$T = \frac{U}{N} + k \quad (13)$$

And substituting this value into equation 3, another form of the total system cost is derived:

$$C = \frac{\alpha U}{N} + \alpha k + \beta v U + \beta N v k + \gamma N \quad (14)$$

Assuming that all the parameters in the system are fixed except  $N$ , determining the critical points leads to an expression for the optimal number of robots  $N^*$ . Taking the derivative of  $C$ , setting it equal to 0, and then solving for  $N^*$ :

$$\frac{\delta C}{\delta N} = -\frac{\alpha U}{N^2} + \beta v k + \gamma = 0 \quad (15)$$

$$N^* = \sqrt{\frac{\alpha U}{\beta v k + \gamma}} \quad (16)$$

The positive root is taken because the number of agents must be positive, and the second derivative  $\frac{\delta^2 C}{\delta N^2}$  is positive so  $N^*$  occurs at a minimum value of  $C$ . Plugging this value into equation 13 produces the optimal search time  $T^*$ .

$$T^* = \sqrt{\frac{U(\beta v k + \gamma)}{\alpha}} + k \quad (17)$$

Equations 14 and 16 can be combined to arrive at the optimal cost  $C^*$  for searching a particular environment using random search:

$$C^* = \alpha k + 2\sqrt{(\beta v k + \gamma)\alpha U} + \beta v U \quad (18)$$

$C^*$  breaks down into essentially three terms. The first,  $\alpha k$ , represents the minimum cost of having to disperse throughout the arena before finding the

target. Generally, however, because the sensor radius is assumed to be small compared to the arena size,  $U \gg k$  so this term will not have a substantial influence on the overall cost. The second term represents the cost of not finishing the task accrued while performing the task (e.g., the damage done by the target before it can be located and neutralized). It will dominate when  $\alpha$  is the dominant cost component.  $\beta$  and  $\gamma$  play a role in this term as well because they influence the optimal number of agents and thus the speed at which the task can be accomplished. The third term represents the cost of searching the required area to complete the task. It will dominate when  $\beta$  is the dominant cost component. It has a relatively simple form because the number of agents in the system does not influence the size of the area that must be searched. Substituting back in for  $U$ , the optimal random search cost can be specified in terms of the component costs and basic task parameters:

$$C^* = \alpha k + 2\sqrt{(\beta vk + \gamma)\alpha \frac{\log(1-S)}{\log(1-\frac{rv\eta}{L^2})}} + \beta v \frac{\log(1-S)}{\log(1-\frac{rv\eta}{L^2})} \quad (19)$$

### 3.2 Coordinated Search

The performance of coordinated search algorithms has been well-studied [1]. In terms of the variables described in this paper, the results are as follows. Coordinated search for  $N$  agents requires breaking the search space into  $N$  equal partitions, and assigning one agent to sequentially search each. The total amount of time  $T_{Pass}$  required for each agent to make a single pass over its entire partition can be stated in terms of the arena size  $L$ , agent speed  $v$ , and sensor range  $r$ :

$$T_{Pass} = \frac{L^2}{Nrv} \quad (20)$$

Given a sensor detect probability  $\eta$ , the total number of passes  $M$  each robot must make can be expressed similarly to equation 7 above:

$$M = \frac{\log(1-S)}{\log(1-\eta)} \quad (21)$$

Thus the total time required for the optimal system to search the arena is as follows:

$$T = T_{Pass}M = \frac{\log(1-S)L^2}{\log(1-\eta)Nrv} + k \quad (22)$$

Where  $k$  represents the time required for the robots to move from the deployment area to their respective partitions. If  $U_{cor}$  is defined as follows:

$$U_{cor} = \frac{\log(1-S)L^2}{\log(1-\eta)rv} \quad (23)$$

Equation 13 is again reached:

$$T = \frac{U_{cor}}{N} + k \quad (24)$$

All of the optimal value derivations in the previous section now apply.

### 3.3 Performance Comparison

Comparing the optimal costs of different search algorithms can provide insight into the conditions under which each type might be more suitable. This can be done by looking at the ratio of the optimal cost of random search  $C_{rnd}^*$  to the optimal cost of coordinated search  $C_{cor}^*$ . The choice of algorithm influences only the value  $U$ , and  $U_{rnd}$  (equation 12) and  $U_{cor}$  (equation 23) are defined above. As shown in [4], the ratio  $U_{rnd}$  to  $U_{cor}$  simplifies as follows:

$$\frac{U_{rnd}}{U_{cor}} = \frac{\frac{\log(1-S)}{\log(1-\frac{rv\eta}{L^2})}}{\frac{\log(1-S)L^2}{\log(1-\eta)rv}} \approx \frac{-\log(1-\eta)}{\eta} \quad (25)$$

The approximation holds when  $\frac{rv\eta}{L^2}$  is close to 0, as is typical when the search arena is large. This equation indicates that as the sensor reliability decreases, the performance gap between random and optimal search strategies closes. However, the cost components play a role as well.

As stated in section 3.1, when  $\alpha$  is the dominant cost component, the second term in the cost function (equation 18) will dominate, so assuming all cost components remain constant across the different algorithms,

$$\frac{C_{rnd}^*}{C_{cor}^*} = \sqrt{-\frac{\log(1-\eta)}{\eta}} \quad (26)$$

Likewise, when  $\beta$  dominates, the third term in the cost function is the most important, so

$$\frac{C_{rnd}^*}{C_{cor}^*} = \frac{-\log(1-\eta)}{\eta} \quad (27)$$

Therefore, aside from sensor detect probability, tasks for which there is considerable time pressure will be more suited to random search strategies than tasks that emphasize economy of effort. This is not an unexpected finding, but this analysis formalizes the tradeoffs involved.

Because the cost  $\gamma$  of building and maintaining different types of robots suitable for each algorithm is difficult to deal with abstractly, it is not considered here. However, it is worthwhile to note that robots capable of the coordinated action will likely cost more than robots suitable for random search.

## 4 Supporting Simulations

Formulation of the optimal search cost is straightforward, but the analysis of the random search algorithm required assumptions about the independence of the success probability over sequential time periods for a single agent as well as across agents. To verify that these assumptions are valid for this type of task, the search task was implemented in simulation and the time and distance required for groups of various sizes to succeed was recorded. To implement the random search behavior, the agents moved forward at a constant speed, making random turns (between  $\frac{\pi}{4}$  and  $\frac{3\pi}{4}$  rad) away from obstacles (walls and other agents) when necessary.

### 4.1 Webots

To maintain a close correspondence with the structure and function of real robots, Webots [5], a 3D sensor-based, kinematic simulator, originally developed for Khepera robots [6] was used to simulate the search task. This embodied simulator has previously been shown to generate data that closely matches real robot experiments [7], [8], so there is reason to believe that the results are representative of a real system.

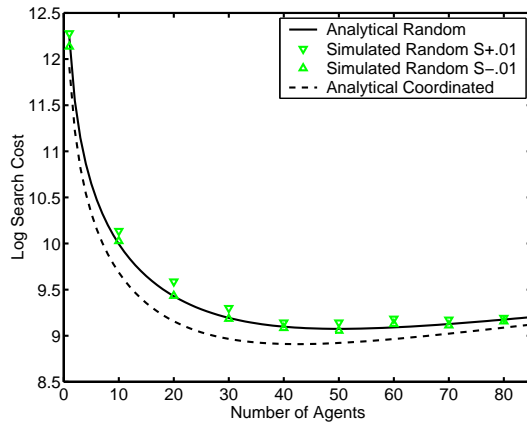
### 4.2 Results

1000 simulations of the random algorithm with group sizes from 1 to 80 agents were run and the time and group distance required to complete the task were measured. The deployment area was always placed in the arena center, and the target was placed randomly throughout the arena for each trial. The dispersal time  $k$  was calculated from the arena length and the agent speed. The task and cost parameter values selected are shown in Table 2. Note here  $\eta$  is significantly less than one and  $\alpha \gg \beta$ , so the random algorithm is expected to perform similarly to the coordinated search.

**Table 2.** Task and Cost Parameter Values

|                           |          |               |
|---------------------------|----------|---------------|
| Agent radius              |          | .5 [m]        |
| Sensor radius             | r        | .5 [m]        |
| Arena length              | L        | 100 [m]       |
| Deployment area Radius    | R        | 10 [m]        |
| Average agent velocity    | v        | 2.9 [m/s]     |
| Minimum dispersion time   | k        | 17 [s]        |
| Desired performance bound | S        | .95           |
| Sensor detect probability | $\eta$   | .5            |
| Cost of not finishing     | $\alpha$ | 10 [\$/s]     |
| Cost of operation         | $\beta$  | .1 [\$/m]     |
| Initialization cost       | $\gamma$ | 82 [\$/agent] |





**Fig. 2.** Simulated and analytical results for this search task. For the simulated data the lower triangles are above  $S - .01$  of the cost data and the upper triangles exceed  $S + .01$  of the cost data. Good agreement between the simulated and analytical results indicates the random search model assumptions are sound

Figure 2 shows the results of calculating the costs in this system analytically compared to the costs derived experimentally. There is good quantitative agreement between the analytical and simulated results for the random algorithm, suggesting that for this task the assumptions of independence hold and the analytical results are valid. Also, it is worthwhile to note that the optimal group size for both algorithms is well above 1 (so the interest in multiple agents completing this task is warranted), and the optimal cost of the random algorithm is fairly close to that of the optimal system. This suggests that if the increased cost of adding coordination and fault tolerance into the optimal system is significant, the random system (which has fault tolerance built in because all of the agents perform the same actions) may be the most efficient.

## 5 Conclusion

This paper presented a quantitative analysis of the tradeoffs between group size and efficiency in collective search tasks that considers both the time-sensitive nature of search completion and the system operating cost. First, the search task was defined and a performance metric was presented that can account for all of the costs associated with the task. Note that computation of the cost parameters may not be simple, but estimates are feasible. Also, while the costs used in this paper were linear functions of the task metrics, any differentiable functional form can be used in this framework. Next, for both random and coordinated search strategies, analytical expressions were derived that can be used to predict optimal system performance bounds given

a particular task description. This analysis also allowed the prediction of the optimal number of agents required to complete a task most efficiently. In addition, the performance benefit of using coordinated search was shown to be dependent on the relative values of the different cost components, with coordinated search being less favored when the cost of not completing the task significantly outweighs the cost of operating the search system. Finally, an embodied computer simulation was used to support the analytical results, suggesting that the assumptions involved in their derivation are sound. These assumptions, which include minimal interference between agents and uniform coverage of the given arena, will not hold in all environments, but they will be approximately correct for many difficult applications where the area to be searched is much larger than the agent extent.

## 6 Acknowledgments

I would like to thank Sarah Farivar for her mathematical expertise and her help reviewing this manuscript. This work is supported in part by the Center for Neuromorphic Systems Engineering as part of the NSF ERC Program under grant EEC-9402726. Support has also been received from the ONR under grant N00014-98-1-0821 and from the ARO under MURI grant DAAG55-98-1-0266. Adam Hayes is supported by a NSF Graduate Research Fellowship.

## References

1. S. Benkowski, M. Monticino, and J. Wiesinger. A survey of the search theory literature. *Naval Research Logistics*, 38(4):469–494, 1991.
2. D. W. Gage. Randomized search strategies with imperfect sensors. In *Proceedings of SPIE Mobile Robots VIII*, volume 2058, pages 270–279, Boston, September 1993.
3. T. Balch and R. C. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Robotics and Automation*, 14(6):926–939, December 1998.
4. D. W. Gage. Many-robots mcm search systems. In A. Bottoms, J. Eagle, and H. Bayless, editors, *Proceedings of Autonomous Vehicles in Mine Countermeasures Symposium*, pages 9.55–9.63, Monterey, April 1995.
5. O. Michel. Webots: Symbiosis between virtual and real mobile robots. In *Proceedings of the First International Conference on Virtual Worlds, VW'98*, pages 254–263, Paris, France, July 1998. Springer Verlag.
6. F. Mondada, E. Franzi, and P. Ienne. Mobile robot miniaturization: A tool for investigation in control algorithms. In T. Yoshikawa and F. Miyazaki, editors, *Proc. of the Third International Symposium on Experimental Robotics ISER-93*, pages 501–513, Kyoto, Japan, 1993. Springer Verlag.
7. A. Martinoli, A. J. Ijspeert, and F. Mondada. Understanding collective aggregation mechanisms: From probabilistic modelling to experiments with real robots. *Robotics and Autonomous Systems*, 29:51–63, 1999.
8. A. T. Hayes, A. Martinoli, and R. M. Goodman. Swarm robotic odor localization. In *Proc. of the IEEE Conf. on Intelligent Robots and Systems*, pages 1073–1078, Wailea, HI, October 2001. IEEE Press.