Proceedings of the Autonomous Vehicles in Mine Countermeasures Symposium, Monterey CA, 4-7 April 1995

# Many-Robot MCM Search Systems

Douglas W. Gage

RDT&E Division, Naval Command Control and Ocean Surveillance Center
NCCOSC RDTE DIV 531, San Diego, CA 92152-7383

## ABSTRACT

A system comprising a large number of identical and very inexpensive robotic search vehicles may serve as an effective tool in a variety of MCM operations, providing improved mine detection and clearance capabilities while reducing cost and the physical risk to MCM personnel. Moreover, the many-robot approach to addressing MCM applications becomes increasingly viable as continuing technological developments provide needed mine detection and other subsystem capabilities at ever decreasing cost. Real challenges remain, however, at the system design level, and the most effective and cost-effective systems will result from careful attention to actual system requirements.

Following a discussion of detection sensor models, a simple analytical framework is employed to demonstrate that the design of a cost-effective many-robot search system can depend sensitively on the interplay of sensor cost and performance levels with mission-specific functional and performance requirements. The issue of how to achieve effectively randomized search strategies that provide uniform search coverage over a specified area is then treated. Finally, the importance of using detailed detection statistics to estimate the independence of successive sensor sweeps and to generate an adequate model of sensor performance is discussed.

## 1. INTRODUCTION

The detection and clearance of mines both ashore and in the surf zone constitute problems which have not yet been satisfactorily addressed, from minefield reconnaissance prior to an amphibious assault, through breaching operations, to humanitarian demining after hostilities have ceased. The obvious dangers to personnel, the problems associated with detecting non-metallic mines, and the difficulties of navigation and communication in the surf zone present difficult challenges to the technologist who would build a Mine Countermeasures (MCM) system. However, recent (and continuing) advances in a number of technology areas offer hope that tools can now be developed to make MCM operations safer and more effective. For many applications, a system comprising a large number of identical and very inexpensive robotic search vehicles may provide an appropriate solution.

In many respects, in fact, MCM operations is an application area which appears to be perfectly matched to the many-robot systems concept:

• The MCM environment is dangerous to humans; a robotic solution allows MCM operators to be physically removed from the hazardous area.

• The MCM environment is also dangerous to machines; the use of multiple inexpensive robotic search elements minimizes the cost of lost system assets, and allows the mission to be performed by the remaining elements.

• One important MCM task is the destruction of mines; using very cheap, deliberately expendible elements allows a one-element-per-mine approach.

• Many mines must be dealt with; the use of many robots allows these targets to be prosecuted in parallel, rather than one at a time.

Prominent among the rapidly advancing technologies which promise to make many-robot systems increasingly feasible and cost effective is Micro ElectroMechanical Systems (MEMS), which is a technology for making objects which are not necessarily small themselves, but posess small features. MEMS devices can be coupled to many physical phenomena to produce a wide range of sensors which are small, inexpensive, robust, and sensitive. The continuing rapid evolution of VLSI and the resulting geometric improvement in computer price/performance provide the processing tools needed to make quasi-intelligent behaviors affordable. The key is mass production driven by the consumer marketplace that promises to make sophisticated technologies available to the implementer of robotic systems at orders of magnitude lower costs than have been traditionally experienced. Many-robot MCM systems will exploit emerging technologies being or soon to be developed for products such as unmanned lawnmowers and vacuum cleaners, and toy "pets" capable of rich interactions with their owners.

This paper briefly explores several issues involved in the design and implementation of man-robot MCM search systems that can be both effective and cost-effective: reducing the very broad design decision space, implementing cost effective search algorithms that provide uniform coverage of a search area, and validating the actual effectiveness of a many-robot MCM system.

## 2. DETERMINING COST-OPTIMAL SEARCH SYSTEM DESIGN

The designer of a many-robot system faces the opportunities and challenges of working within a design space providing many degrees of freedom. Required system level functionality and performance may be achievable with many different system configurations, and the designer must make appropriate choices concerning the system as a whole (e.g., the number of different castes of elements, the populations of each, the desired ensemble behaviors, and the command and control organization of the ensemble), and the capabilities of each element type (e.g., effectiveness and range of mission sensors and effectors, vehicle platform limitations such as speed, endurance, weight, power source, and communications capabilities and programmed behaviors). In some applications, a large number of the simplest possible elements may be the right answer; in other applications the best solution may be a much smaller number of elements incorporating higher capability sensors, effectors, processing and/or communications resources. The elements may exhibit simple independent behaviors, or complex coordinated ones. And superficially similar behaviors (both individual and ensemble) can often be implemented in a number of very different ways. In this section we explore some system design dimensions and the resulting system tradeoffs in the context of the MCM area search application.

Minesweeping is the archetypal area search application: a two dimensional area of interest is suspected to contain some number of objects of interest (targets), and the search task is to detect these targets. For the purposes of this analysis, the targets are assumed to be stationary. While the basic problem addressed -- in a number of interesting variations -- by the field of optimal search theory [1, 2] is to devise strategies to optimally allocate a given set of search resources, our purpose in the following simple analysis is to develop a framework for choosing an optimal set of robotic resources from within a very rich design space, given specified operational goals and cost constraints.

We consider the following simple search problem: a number of search elements, each carrying a detection sensor of some sort, move through a predefined search area attempting to detect an unknown number of stationary target objects. The targets are a priori equally likely to be at any point within the search area. Whether any given searcher actually detects any given target at any point in time depends on the relative geometry of searcher and target at that instant, as well as the physical characteristics of the detection sensor(s), the environment, and the target itself.

### 2.1 Models of Detection

It is standard practice to approximate a detection sensor's performance in order to facilitate analytical modeling of the search process. The simplest approximation is the *definite range law*: targets which come within a certain distance of the searcher are always detected, and targets which do not come that close are never detected (hence the searcher cuts a clean swath like a "cookie cutter"). A simplifying virtue of this model is that the sensor is characterized by a single parameter: its maximum detection range. If this critical range value is chosen appropriately, this model yields the correct number of targets detected by a single searcher making a single pass through an area.

A somewhat more sophisticated approximation to a sensor's detection performance results if we are willing to add a second parameter and specify some mean probability of detection <u>less than one</u> over a specified range. As was the case with the "cookie cutter", this *imperfect sensor* model can be parameterized to give the correct number of targets detected by a single searcher making a single pass through an area. Moreover, in the case of multiple coordinated sweeps through an area, the imperfect sensor provides better results than the cookie cutter, which gives an unreasonably optimistic prediction. Since many-robot systems are likely to use low quality sensors, the *imperfect sensor* model is used in our analysis below.

### 2.2 The Problem, More Precisely

Let us now be more specific in describing our search system and application. Our system consists of some number N of identical robotic elements, each of which can move about in the search area while carrying an imperfect sensor (in the sense of section 2.1) of nominal range r and detection probability p: any target which lies within a distance r of an element's track is detected with probability p as the element passes by. Targets farther than r are never detected, false alarm detections do not occur, and what action the element might take with detected targets is of no concern. Each element is capable of traveling a total distance d during the mission; this limitation may be due to limited energy storage, or to operational constraints on the duration of the mission, coupled with the maximum speed which can be achieved while operating the sensor payload. Let S be the average number of times that each point in the search area (whose area is A) is sensed; we call S the "sweep fraction" of the search, and calculate it as

$$S = 2r\, d\, N / A \tag{1}$$

### 2.3 Coordinated and Random Search Processes

If we let D represent the probability that any given target is detected, or, equivalently, the expected fraction of targets detected, then our first step is to calculate D as a function of S for two archetypal search strategies: in the first case, the elements execute a perfectly coordinated search pattern -- we may have only one element following a "lawn-mower" pattern, perhaps by employing a very accurate navigational system, or we may have a number of elements moving in a tightly coordinated formation; in the second case, we employ (presumably less expensive) elements capable only of staying within the designated search area, but otherwise wandering completely randomly.

We define these two archetypal search strategies in abstract and discretized form to provide the basis for further probabilistic analysis:

We are given an urn containing a very large number of otherwise identical marbles, a small percentage of which bear an invisible mark. We have a machine that can tell us if a marble presented to it is one that bears a mark. The machine is not perfect: it detects a marked marble only with probability p, but it never indicates a false positive. The task is to separate out the marked marbles by running the marbles through the testing machine. The two candidate strategies are:

Coordinated search: we pull marbles at random from the urn and test them on the machine. After testing, we place the unmarked marbles into a second urn. When all the marbles have been tested, we pour these marbles back into the first urn and repeat the process.

Random search: we pull marbles at random from the urn and test them on the machine. After testing, we return each unmarked marble to the urn and mix it thoroughly.

What can we say about these two abstract strategies? First, the coordinated search is clearly superior, since we never test a marble for the nth time until all marbles have been tested n - 1 times (and the more times a marble has been tested, the less likely that it is an undetected marked marble). If p = 1, we will find all the marked marbles by making just one pass (testing each marble just once). On the other hand, with a random search we can never guarantee that we will find all the marked marbles, no matter how many passes we make. The second thing to note is that in order to perform the coordinated search we have to have a second urn; in real life as well, implementing a coordinated search will generally entail additional costs.

In fact, a coordinated search is the best strategy you can employ, and most real-world search tactics are designed to achieve a coordinated search. Unfortunately, real-world constraints (such as navigation inaccuracies) mean that the results actually achieved usually more closely reflect those predicted for a random search. On the other hand, a random search is not the *worst* strategy that can be employed, and, as we will see below, it is not readily apparent how to realize a *deliberately* random search that will provide uniform coverage of a two dimensional or three dimensional space.

In the marble world abstraction, S measures the average number of times each marble has been tested: the total number of tests we have made divided by the number of marbles in the urn. For the coordinated case, we calculate

$$D_c = 1 - (1 - p)^{S_c} \tag{2}$$

In fact, this equation is true only for integer S, with straight line interpolation between integers, but we won't worry about this approximation for now -- it provides an upper bound on the performance of the coordinated strategy, and the approximation gets very good for large S. On the other hand, we find for the random case (in the limit of infinitely many marbles)

$$D_r = 1 - e^{-p\, S_r} \tag{3}$$

Unlike equation (2), equation (3) is accurate for non-integer S. Figure 1 shows the behaviors of equations (2) and (3) when p = 0.8.

## 2.4 Measures of Search Strategy Effectiveness

We now consider the question of how much "better" is the completely coordinated search (first case) than the completely random search (second case).

Different search applications require different measures of systems effectiveness. The task may involve many targets, as in the case with minesweeping, or it may involve only a single target (a sunken submarine, for example). And, in the case of many targets, the desire to minimize the number of targets missed in a swept area (as in minesweeping) is not the same thing as the desire to maximize target detections per amount of search effort (as in prospecting for manganese nodules).

For the minesweeping application our goal is to minimize the cost of detecting a specified fraction D of the targets, which allows us to make the notion of "better" a precise one by recasting the question as: for a given sensor effectiveness p, how much larger a sweep fraction $S_r$ is required so that the detect fraction D of a completely random search (second case) is equal to that of a completely coordinated search (first case) with sweep fraction $S_c$? By equating $D_c$ in equation (2) with $D_r$ in equation (3), we calculate the "search gain" G, or factor reduction in required search effort afforded by a coordinated search as compared to a random search:

$$G_{c,\text{ many target}} = S_r / S_c = -\ln (1 - p) / p \qquad (4)$$

This result says that, for any sensor detection probability $p < 1$, we can achieve any desired detect fraction D equally well by (a) performing a completely coordinated search with sweep fraction $S_c$ we calculate from equation (1), or by (b) performing a random search with sweep fraction $S_r$ calculated from equation (2), which is larger than $S_c$ by a factor (equation (4)) which depends only on p, and is independent of the desired detect fraction D and the S required to achieve it. Figure 2a shows how this ratio varies with p; it is fairly small for poor sensors, and, as is obvious from inspection of equation (4), the numerator of the expression dominates the behavior as p approaches 1.

The fact that $S_r/S_c$ is independent of the desired D suggests that we might be able to make use of the corresponding ratio to quantitatively describe the relative effectiveness of other search strategies. Accordingly, we define the "search gain" G of any given search strategy s as:

$$G_s = S_r / S_s$$

In other words, the search gain $G_s$ of a search strategy s is the factor by which that strategy reduces the search fraction required to achieve any desired detect fraction, compared to a random search. So we can then write the detect fraction $D_s$ for any search strategy s as:

$$D_s = 1 - e^{-p\ Gs\ Ss} = 1 - e^{-p\ Gs\ N\ 2r\ d\ /\ A} \qquad (5)$$

G will be most meaningful if it depends only on p, as in the case of the fully coordinated search of case 1, but it will still be useful if it varies slowly and predictably with S. Note that while the G calculated for the completely coordinated search -- equation (4) -- serves as the maximum value achievable by any search strategy, it is entirely possible for a strategy's G to be less than 1.0 if element paths are positively correlated, as in the case of ants following each others' pheromone trails.

## 2.5 The System Design Space

The reason for choosing to write equation (5) in this format is that the variables in the expression essentially provide a coordinate system for the system design space, and they break naturally into three groups:

- p and r are characteristics of the primary mission sensor: its single-pass probability of target detection and its range. Note that in some cases the choice of a sensor will have immediate implications for other system characteristics; for example, p and/or r may depend critically on element speed, as with sonar self noise.

- N and d, which together with r determine the sweep factor S, are parameters pertaining to the element platforms: the number of elements to be employed, and the effective search range of each. Mission time and stealth requirements, maximum platform speed, and energy storage limitations may be important in determining d.

- G is the gain in search effectiveness due to the coordination (*negative* correlation) of element search paths to provide balanced coverage; this is where vehicle search strategy behaviors are accounted for.

Figure 3 presents a (fictitious, or "toy") simple application example in which the design space consists of 12 possible systems, allowing any of three possible sensors, a completely random or a completely coordinated search strategy, and an optional battery upgrade to double vehicle range, with each choice having an associated specified cost. As the quality of the sensor (its raw target detection probability p) improves, the most cost effective design shifts from employing a large number of the least expensive and least capable elements to a much smaller number of more expensive and more capable elements. In the real world, with a much more complex design space and unavailable, unreliable, and expensive cost estimates, the design process would probably begin with the determination of p for the mission sensor package, since determining p determines the maximum G that can be achieved. The lower the value of p, the more likely it is that it will be more cost effective to utilize a random search strategy (G = 1) and increase S, rather than implement a coordinated search strategy, with its added cost and complexity. Once the search strategy is selected, determining G, then the tradeoff between sensor range, number of elements, and search range per element can be made to realize the required S in the most cost effective fashion.

| sensor | | | | G*S | coord | coord? | double | cost / | num | |
|---|---|---|---|---|---|---|---|---|---|---|
| | cost of | | | | | | | | | |
| cost | range | | detect | p | reqd | gain G | | range? | elem | |
| elems | | system | | | | | | | | |
| 1 | 1 | 0.5 | 9.21 | 1.39 | | | 11 | 460.52 | 5066 | <-- |
| - | | | | | | | | | | |
| 1 | 1 | 0.5 | 9.21 | 1.39 | | yes | 26 | 230.26 | 5987 | |
| 1 | 1 | 0.5 | 9.21 | 1.39 | yes | | 41 | 332.19 | 13620 | |
| 1 | 1 | 0.5 | 9.21 | 1.39 | yes | yes | 56 | 166.10 | 9301 | |
| 10 | 1 | 0.7 | 6.58 | 1.72 | | | 20 | 328.94 | 6579 | |
| 10 | 1 | 0.7 | 6.58 | 1.72 | | yes | 35 | 164.47 | 5756 | <-- |
| - | | | | | | | | | | |
| 10 | 1 | 0.7 | 6.58 | 1.72 | yes | | 50 | 191.25 | 9562 | |
| 10 | 1 | 0.7 | 6.58 | 1.72 | yes | yes | 65 | 95.62 | 6216 | |
| 15 | 1 | 0.9 | 5.12 | 2.56 | | | 25 | 255.84 | 6396 | |
| 15 | 1 | 0.9 | 5.12 | 2.56 | | yes | 40 | 127.92 | 5117 | |
| 15 | 1 | 0.9 | 5.12 | 2.56 | yes | | 55 | 100.00 | 5500 | |
| 15 | 1 | 0.9 | 5.12 | 2.56 | yes | yes | 70 | 50.00 | 3500 | <-- |
| - | | | | | | | | | | |

Figure 3. Design spreadsheet for a fictitiously simple application design space consisting of only 12 possible systems. The mission is to search an area A = 10000 and achieve a detect fraction D = 0.99. A base vehicle costs 10, and has a range of 100. The additional cost of better batteries which double the range to 200 is 15. The cost of sensors and processing to implement a completely coordinated search strategy is 30. Three possible mission sensors are available. Equation (4) has been used to calculate the number of elements required, the cost per element, and the total system cost for each of the twelve possible combinations of mission sensor, random or coordinated behavior, and baseline or improved batteries. It is seen that (a) the most cost effective system using the least expensive sensor uses the simplest possible elements, (b) the intermediate sensor justifies the incorporation of the battery upgrade, but not the coordinated search behavior, and (c) using the most expensive sensor justifies both the battery upgrade and the coordinated search behavior.

## 3. ACHIEVING UNIFORM COVERAGE WITH "RANDOM" SEARCH STRATEGIES

In the previous section we considered random search from the perspective of the performance predicted by a particular analytical model; in this section we consider the problem of how to actually generate a "random" search path which (a) provides uniform coverage over a designated search area and (b) can be implemented with a minimum investment in navigational sensors, processors, external navigational aids, communications resources to support coordination between search elements, etc.

"Randomized" search trajectories are also of special interest in searching for a potentially mobile target or targets [4,5]. By "randomized" we mean a path which is not predictable, so that a mobile target which somehow gains partial or even complete knowledge of a searcher's prior path and current position can not predict its future path and thereby evade detection. The strategy must clearly not attempt to be more efficient than the random search model, since in this case the target could evade capture by favoring locations that the searcher had already visited. In general, the searcher's path must not follow any predictably discernible pattern that can permit the target to evade capture. More critically, the search path must not be determined by fixed reaction to environmental features, since this may permit the target to spoof the searcher into changing his path (see [6] for a discussion of security issues relevant to robotic systems). In addition, navigational inaccuracies due to sensor bias may lead to systematic gaps in coverage even if the target is not sophisticated.

A search strategy of following straight line paths between points chosen at random within the search area was proposed by Henze [7], but simulations by McNish [8] showed that this does not lead to a uniform distribution of search effort, instead providing excessive coverage of the central area at the expense of the periphery. In order to increase the coverage near the boundaries, McNish considered a number of other algorithms that generate paths consisting of sequences of chords within a circular search area.

The use of a search path consisting of chords is appealing for several reasons. First, the searcher travels as far as possible between changes of direction and is guaranteed not to visit any point twice during the transit of a chord. Second, a chord-based strategy is completely specified by the algorithm that determines the direction of the next chord each time the searcher arrives at the boundary of the search area. The specific reflection algorithms simulated by McNish over a circular disk search area were *specular* reflection (angle of reflection equal to angle of incidence, like light from a mirror), *uniform* reflection (angle of reflection distributed uniformly), and *diffuse* reflection (like light reflecting from a matte surface). McNish found that only diffuse reflection reliably provided uniform search coverage.

The diffuse reflection algorithm specifies a random distribution of reflection angle $\theta$ with probability defined as:

$$\text{Prob}(\theta) = 1/2 \sin(\theta)\, d\theta \qquad (6)$$

where $\theta$ measures the angle of the chord from the tangent to the boundary at the reflection point. Again for the case where the search area is a circular disk, Lalley and Robbins [9] mathematically proved that, in the limit as the length of the search path goes to infinity (corresponding to the sensor range going to 0), the search coverage is uniformly distributed: "the long-run search effort devoted to any region of D is proportional to its area, and consequently, the 'infinitesimal search effort' is the same at every point of D. It is intuitively plausible that a good search plan should have this property, for if not, a target could gain an advantage by hiding at a point scheduled for a low level of search effort" [9]. For non-circular but still convex search areas, the "optimal" generalization of the diffusion algorithm has been shown to be:

$$\text{Prob}(\theta) = \text{constant}\ K(\theta) \qquad (7)$$

where $K(\theta)$ is the length of the chord from the reflection point in the direction $\theta$ [3].

It is clear, however, that there are an infinite number of search strategies (including diffuse reflection) that provide uniform coverage of, say, a rectangular area "in the limit as the length of the search path goes to infinity", but some do a lot better than others in the short term. Metrics must be developed to assess the uniformity (and hence effectiveness) of coverage produced by competing search algorithms, over finite search path lengths. The tradeoff between strategies must then be performed at the system level, taking into account the cost of implementing each strategy. For example, while the diffuse reflection algorithm requires only that each search element be capable of traveling in a straight line and detecting the presence and orientation of the boundary of the search region, the chord length algorithm requires in addition the ability to calculate the length of the chord across the search area in each direction from every possible point of reflection -- essentially to accurately "know" its own position and the complete geometry of the search area.

Use of a *diffusion* strategy has been analyzed by Eagle at the Naval Postgraduate School [10] and, more recently, a "quasi-diffusion" "move forward a randomized distance, turn a randomized angle, and repeat" algorithm has been simulated in the context of the "Lemmings" many-robot system for neutralizing shallow water mines [11,12]. These types of algorithms do not claim to provide uniform coverage, but they provide predictable coverage relative to their deployment point, and they can be implemented with only a compass and an odometer. Analysis that could guide the selection of "optimal" parameters for the distance and turn randomizations remains to be done; exploration of the approach has so far relied completely on simulation.

## 4. SWEEP (IN)DEPENDENCE, MODELING, AND PERFORMANCE VALIDATION

Both probabilistic analysis and Monte Carlo simulation of the search process require that some assumption be made regarding the statistical independence of the probability of detection of a given target with successive sweeps of the same or a different sensor. The easiest assumption, and the one underlying the entire analysis presented above, is that successive sweeps are indeed statistically independent. It is clear, however, that this is not in fact true: some mines will undoubtedly be buried deeper than others, and, in a mixed field, some mine types will be intrinsically easier to detect than others due to higher metal content.

This lack of independence can be accounted for by modeling the population of mines as a distribution of populations of varying probability of detection, and using actual field measurements to characterize the distribution. The procedure would be to make multiple sweeps of the search area and count the number times each mine is detected (or, if the search process affects the search environment, count the number of additional targets detected on each pass). The sweeps could be made using the same or different sensors, including an intensive manual sweep of the area which would, presumably, find 100% of the targets. The population of detected false alarm non-targets can be modeled in exactly the same way.

Even without knowing "ground truth", Bayesian techniques can be used to characterize the performance of each sensor against the various target classes in the field, and to estimate the remaining threat. A "back of the envelope" rough calculation goes something like this: if there are M mines actually out there, then the first sensor pass finds p M mines, leaving (1-p) M. The second pass finds p (1-p) M, or (1-p) times the number found on the first pass. The number of mines still undetected is $M (1 - (1-p)^2)$. Hence if we find 60 mines on the first pass, and 6 on the second pass, we estimate p = .9, and figure that about .66 mines remain. If, onthe other hand, we find 12 mines on the second pass, we estimate p = .8, and expect that about 3 mines remain undiscovered. Again, Bayesian techiques provide a much more rigorous method for doing this kind of analysis.

Using actual field measurements to arrive at a more accurate (and necessarily more complex) representation of the performance of a sensor against a target field can provide value in several contexts. During sensor development, it can characterize the variability of a sensor's performance, and help identify factors that affect it. During MCM system development, it can provide an understanding of how and whether to combine the outputs of different candidate detection sensor subsystems. At the beginning of a clearance operation, measurements on a small representative subarea can be used to develop expectations of the effectiveness of the search of the whole area. Finally, data gathered during the sweep operation and from additional samplings can be used to provide a more accurate estimate of the threat remaining after the operation is complete.

## 5. CONCLUSION

Technological developments promise to make the many-robot approach increasingly viable for addressing MCM applications, by providing at ever decreasing cost the vehicle, sensor, and processing subsystem capabilities needed both for mine detection and for navigation in the operational environment. Real challenges will remain, however, at the system design level. The most effective and cost-effective <u>systems</u> will result from careful attention to actual system requirements, and need not <u>necessarily</u> employ the latest and most capable subsystems.

MCM search systems consisting of many inexpensive robots may well tend to use randomized rather then coordinated search strategies for two reasons: (1) the increase in effectiveness provided by a coordinated (as opposed to random) search strategy decreases as the capability of the search sensor (probability of target detection) decreases; and (2) the cost of implementing the navigation capabilities necessary to support a coordinated search strategy may be prohibitive, relative to the cost of a less capable search element. However, careful analysis is required to select and implement an appropriate strategy that efficiently provides the required area coverage. Developers of robotic MCM search systems must do more than simulate one or two candidate search strategies; fifty years of developments in the field of search theory can provide them with analytical tools to help <u>understand</u> tradeoffs relative to overall system search performance. Meanwhile, the advent of search systems consisting of inexpensive robots carrying minimum capability sensors provides search theorists with the opportunities of a previously unexplored problem domain.

## 6. ACKNOWLEDGEMENTS

This work is supported by the Intelligent Systems Program of the Office of Naval Research, Computer Science Division. The author wishes to acknowledge stimulating interactions with Alan Washburn, Jim Eagle, and Al Bottoms of the Naval Postgraduate School, Doug Hoskins of Boeing, and Teresa McMullen of ONR.

## 7. REFERENCES

1. Koopman, B.O. **Search and Screening**. Pergamon Press, New York, 1980.

2. Benkoski, S.J., Monticino, M.G., and Weisinger, J.R. "A Survey of the Search Theory Literature", **Naval Review Logistics**, Volume 38, August 1991, pp 469-494.

3. Gage, D.W. "Randomized Search Strategies with Imperfect Sensors", **Proceedings of SPIE Mobile Robots VIII**, Boston MA, 9-10 September 1993, Volume 2058.

4. Fitzgerald, C. "The Princess and Monster Differential Game", **SIAM Journal on Control and Optimization**, Volume 17, 1979, pp 700-712.

5. Lalley, S.P., and Robbins, H.E. "Stochastic Search in a convex region", **Probability Theory and related Fields**, vol 77, 1988, pp 99-116.

6. Gage, D.W. "Security Considerations for Autonomous Robots", **Proceedings of Symposium on Security and Privacy**, Oakland CA, 22-24 April 1985, pp 224-228. Reprinted in **Computer Security Journal**, vol 6, 1990, pp 95-99.

7. Henze, J. "Random Search Formulas and the Distribution of Detection Times", unpublished study for COMSUBPAC, August 1986.

8. McNish, M.J. "Effects of Uniform Target Density on Random Search", Master's Thesis, Naval Postgraduate School, September 1987.

8. Lalley, S.P., and Robbins, H.E. "Uniformly Ergodic Search in a Disk". **Lecture notes in pure and applied mathematics, v. 112, Search Theory**. Marcel Dekker, Inc., New York, 1989, pp 131-151.

10. Eagle, J.N. "Estimating the Probability of a Diffusing Target Encountering a Stationary Sensor", **Naval Research Logistics**, Volume 34, 1987, pp 43-51.

11. Mangolds, A. "Lemmings - a Swarming Approach to Shallow Water Mine Field Clearance", **AUVS-93, the Twentieth Annual AUVS Technical Symposium**, Washington DC, 29-30 June 1993,.

12. Guillebeau, C.A. "Comparison of Swarming Vehicle Motion Patterns for Very Shallow Water / Surf Zone Minefield Clearance", **Proceedings of Autonomous Vehicles in Mine Countermeasures Symposium**, Monterey CA, 4-7 April 1995.

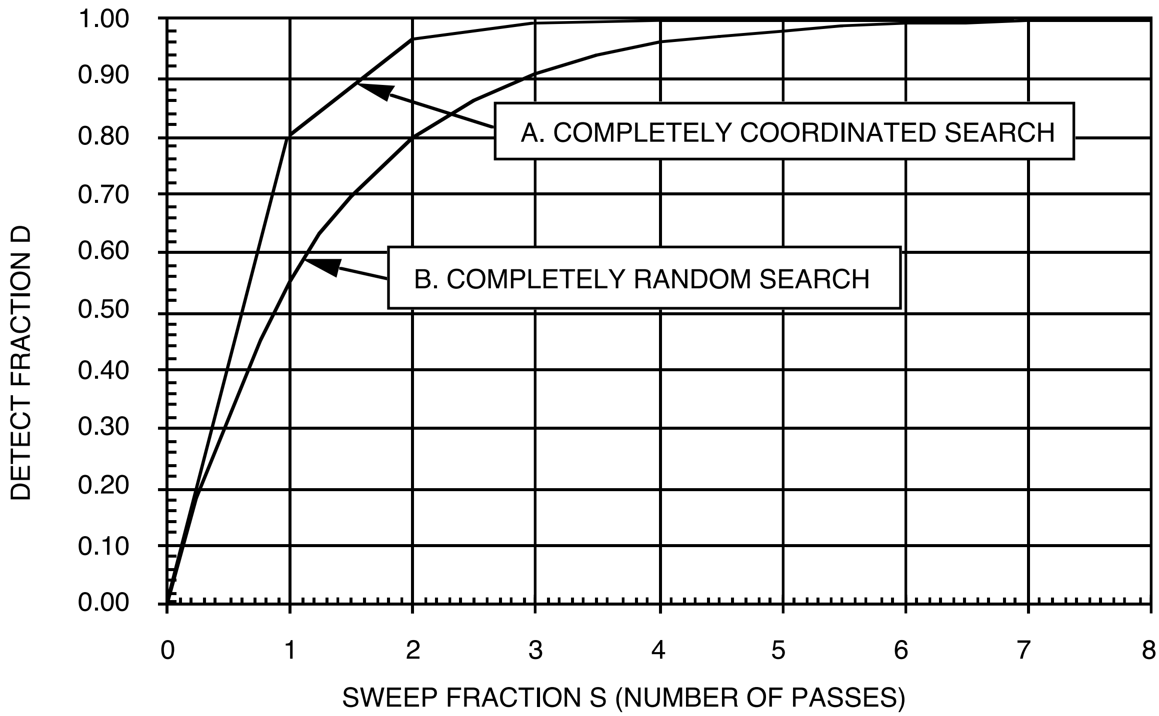


Figure 1. Detect Fraction (D) as a function of Sweep Fraction (S) for sensor probability of detection (p) equal to 0.8. (a) Completely coordinated search strategy, equation (2). (b) Completely random search strategy, equation (3).

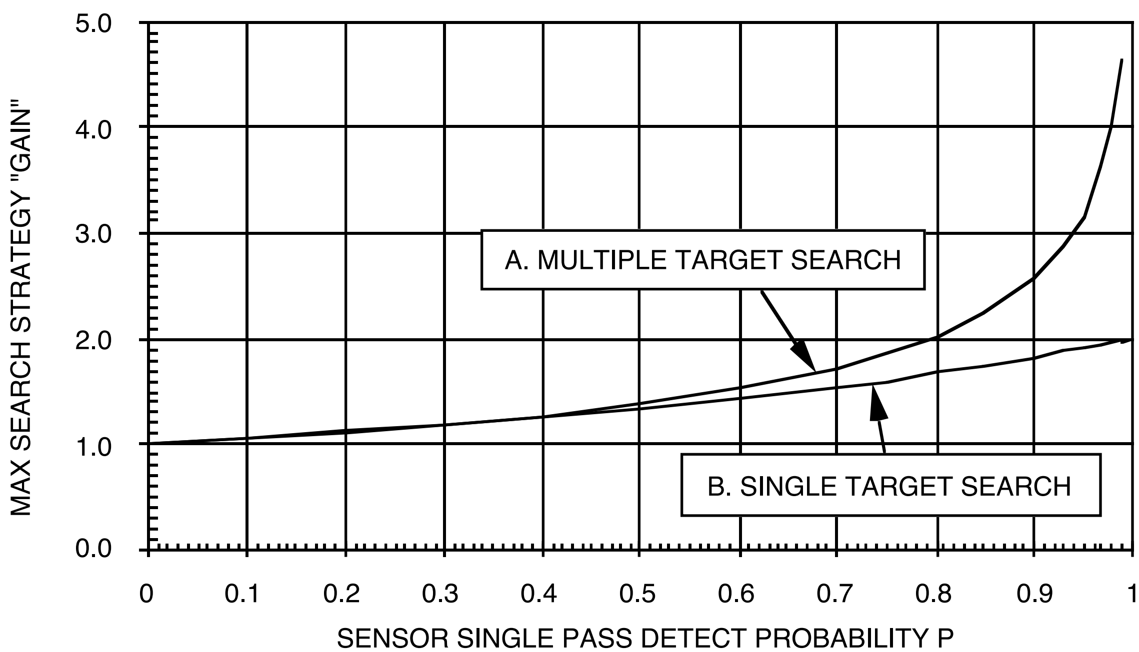


Figure 2. Search strategy gain ($G_c = S_r/S_c$) of a completely coordinated search strategy over a completely random strategy, as a function of the sensor target detection probability (p), for (a) finding a specified fraction of multiple targets, equation (4), and (b) finding a single target, see reference [3].