Elements, Modules and Algorithms
For
Planning Optimal Searches

A High Level Overview

30 September 2002

Prepared for
U. S. Coast Guard Office of Search and Rescue

Prepared by
J. R. Frost
Potomac Management Group, Inc.
510 King Street, Suite 200
Alexandria, Virginia 22314

Table of Contents

Abstract

The evolution of search planning methodology for search and rescue (SAR) from its origin in WWII is reviewed. Advances in search theory and algorithm development up to the present are also reviewed. Computer-based implementations of both analytical and stochastic simulation search planning models that have been developed by or for the Coast Guard are examined in some detail. For purposes of comparison, a simple non- SAR model developed in a computer programming language specifically designed for computer-based simulations is shown. Certain important differences between the basic characteristics of simulations of that type and the SAR search planning simulation embodied in the Coast Guard's Computer Assisted Search Planning (CASP) system are studied. Particular emphasis is given to features that should be common to all simulations but which are either handled differently or ignored by CASP. Of these, the most crucial is handling the passage of simulated *time* via a simulation clock. Attention is called to the problems caused by these differences. A new paradigm and suggested elements, modules and algorithms for the next search planning tool are given. These include the basic features common to all simulations as well as the advances in search theory, algorithms, environmental data, computer technology, and support software such as geographic information systems. A critical success factor for developing any new search planning tool is a multi-disciplinary development team that includes experts in all the germane fields: advanced search theory and algorithms, oceanography, meteorology, development of computer-based simulations for operational use, environmental models and data products, especially those for the near-shore region, SAR operations, sensor and search platform capabilities, search object pre- and post-distress motion and detection characteristics, etc.

Executive Summary

Search theory and practical implementations of it originated in WWII. At first these practical implementations were gross simplifications of the theory since computers were not then widely available. All solutions for search planning problems were done manually with pencil, paper, maneuvering boards and paper charts. Perusing the various editions of the National Search and Rescue Manual between 1959 and 1991 reveals the evolution of manual methods.

Circa 1970 the USCG fielded its first computer-based search planning tool—the Search and Rescue Planning (SARP) system. This tool was basically a software implementation of the manual search planning method with some additional capabilities made possible with the limited computing power then available. The primary improvement over manual methods was the ability to compute hourly drift updates from either gridded environmental data products or user-supplied environmental data.

In 1974 the USCG fielded its first version of the Computer Assisted Search Planning (CASP) system. CASP was the first, and to date the only, tool that applies computer-based simulation technology to SAR search planning. Whereas SARP computed one or two drift trajectories, CASP computed thousands by sampling from statistical distributions about each of the oceanic drift parameters. Unfortunately, the sophistication of the simulation was severely limited by the lack of computing resources at that early date. (A modern PC has literally millions of times more computing power and data storage capacity than the Coast Guard's central computer did in 1974.) In the early 1980s, SARP was discarded and CASP was completely re-written in conjunction with a change in the Coast Guard's central computing resources. However, the CASP re-write largely confined itself to software engineering and user interface issues. Specifically, it did not address the design limitations imposed by the limited computing capacity originally available even though the new computing resources enlarged the available capacity by many times.

Computer-based simulation was still a new field in 1974. That, combined with the computing limitations, caused the original CASP developers to make a number of simplifying assumptions and take a number of computational shortcuts that have hampered CASP's further development ever since. The 1980's re-write did not address these issues. In addition, the subsequent availability of the Coast Guard's first geographic information system (GIS), the Geographic Display Operations Computer (GDOC), led to two developments that further restricted opportunities to improve CASP. First, an interface between GDOC and CASP was developed that presented a graphical user interface (GUI) to the user but behind the scenes emulated the old text-based menu-driven dialogue. This was a complex and fragile kludge that prevented many improvements due to the need for complex and closely coordinated changes to two software suites at two different Coast Guard commands with different missions and priorities. Second, an automated manual method (AMM) was developed and integrated with GDOC. Although the algorithms and capabilities were considerably more primitive than those of SARP, this method became the mainstay of the National SAR School's maritime search planning curriculum. GDOC has been replaced by the Command and Control Personal Computer (C2PC) GIS, AMM has been replaced by the Automated Manual Solution (AMS). While there have been substantial improvements in the user interface, the underlying solutions are no better than those obtained with pencil and paper prior to SARP. The CASP interface has also remained basically unchanged and unchangeable.

While CASP was and still remains the most sophisticated search planning tool in existence, it has many shortcomings and flaws. Some of these are programming errors or oversights but many are a result of the necessarily limited design parameters of the 1970s. Chief among these limitations is CASP's inability to aid the search planner with the scenario analysis needed to initialize the search problem. Such analysis is crucial whenever the location of the distress is not known within close limits. This shortcoming can be overcome by using well-known simulation techniques. Applying such techniques will also solve many other previously insoluble CASP problems. Methods for correcting this and the many other deficiencies in the Coast Guard's search planning software are known, available, and can be implemented at low risk provided a multi-disciplinary development team that collectively possesses the required expertise in all the germane fields can be assembled. In addition to correcting deficiencies, a multi-disciplinary team will make substantial enhancements and improvements to Coast Guard search planning possible. These include better methods, also described in the body of the report, for optimally allocating the available search effort that account for search object motions and states, accounting for uncertain sweep width, improved drift models and improved search evaluation techniques that properly account for the relative motion between search facilities and search objects. Many of these methods are described in the body of the report and an extensive bibliography is provided.

Due to the difficulties mentioned above, CASP has yet to take full advantage of the large body of SAR-related research done by the USCG Research and Development Center, the large body of research in search theory, or the advances in environmental data quality and availability that have come about since CASP's original implementation. Such improvements will reduce the mean time required to locate and assist survivors, save more lives, reduce risk to searchers by reducing the amount of time they are exposed to the hazards of searching, make more efficient use of the available assets, make more resource hours available for other SAR and non-SAR missions and ultimately reduce the average cost per search.

Elements, Modules and Algorithms For Planning Optimal Searches

A High Level Overview

1 Introduction

The U. S. Coast Guard is authorized by Congress to perform search and rescue (SAR) operations to protect the safety of life and property at sea. The Coast Guard has a strong tradition of rendering assistance to distressed mariners dating back to its inception as the Revenue Cutter Service in 1790. However, prior to the Second World War, it seems that the main emphasis was on rescue of survivors from reported or observed distress incidents. Searching for survivors seems to have been generally limited to searching the immediate vicinity of the distress incident for a relatively brief period.

1.1 Organizational Evolution of SAR

Things changed dramatically for SAR (or "air sea rescue" as it was called) during World War II, as related by those who were there at the time. The following summary is excerpted from an article in the *Air Sea Rescue Bulletin* issue of July 1946 [ASR, 1946a] entitled "Evolution of SAR – an editorial":

> "Before World War II there was little need for an extensive air sea rescue organization. Few planes attempted the long over water flight across the Pacific, and flying under adverse weather conditions was negligible. Most forced landings of aircraft were probably due to mechanical failure, and international merchant shipping services were considered sufficient to care for them, and for the comparatively few cases of a marine disaster.
>
> "With the advent of war this picture changed rapidly. All types of aircraft were required to fly over water in all kinds of weather. Due to the necessity of speeded-up training, inexperience, according to peacetime standards, was the rule rather than the exception among pilots and crew. Added to these were the normal hazards of war. As a result, forced landings and ditchings increased rapidly, increasing numbers of vessels were sunk or disabled, and with this situation came the need for an increasing rescue coordination.
>
> "Official recognition came in May 1940, when the original British Air Sea Rescue Unit was formed in the critical Dover area. The outstanding success of this small unit, not only in saving lives but in increasing the morale of airmen, gave impetus to the service. …
>
> "Prior to the United States' entry into the war, there was no comprehensive American air sea rescue plan or program. The Coast Guard had developed some of its aspects but, on the whole, purely local facilities were used, and the practice of diverting commercial surface craft in cases of forced landing at sea had grown up parallel with prewar aviation advancement.
>
> "This faint pioneering, combined with the experiences of the British, served as a nucleus for the present American unit. … In February 1944, the joint chiefs of staff recog-

nized the requirement and necessity for rapid joint service development of air-sea rescue equipment, procedures, and techniques, and requested the Secretary of the Navy to establish in the Coast Guard the Air Sea Rescue Agency to coordinate studies conducted in these fields by the various United States services, and to maintain liaison with services of allied governments. …

"The Air Sea Rescue Agency is not to be confused with the Coast Guard's Office of Air Sea Rescue. The difference between them lies in the fact that while the Agency is charged primarily with the collection and distribution of information concerning the development of equipment, procedures, and techniques, and coordinating studies in these fields for the benefit of United States and allied military services—the Coast Guard's air sea rescue organization is an integral part of its office of operations. …

"It is just a little more than a year since the nations of the world met at Chicago to discuss what cooperative means could be found for developing international civil aviation for the benefit of mankind . . . and just a few months ago the instrument created by those nations, the Provisional International Civil Aviation Organization (PICAO), began to work for the amicable, equitable and orderly development of international air traffic in all its phases. …

"The Search and Rescue subcommittee of PICAO is developing a suggested program for the nations of the world in developing the rescue services of the world into a globe-encompassing network that would save many victims of air crashes in remote places along the world's air lanes. It submitted a report of standards it believed necessary for international search and rescue operations. It recognized that while the wartime experience is by no means a fair criterion to apply to commercial aviation, the record of the wartime years in saving life indicates the importance of having an organized plan for preparedness in advance of the time when the emergency arises. …

"The mission of search and rescue is a joint responsibility of all nations and peoples. Further, it is a team effort . . . an operation which, at one time or another, will call upon every type of vessel and plane, plus the ingenuity and initiative of groups and individuals, to assure success.

"The scope of peacetime search and rescue widens as world airlines increase the frequency of their transoceanic flights. The resumption of peacetime maritime pursuits, and the development of private flying also step up the ratio of possible emergency incidents.

"In spite of man-made safety measures, mechanical failures, floods, and storms will continue to be hazards for the traveler on land or sea. Wartime experiences in search and rescue, however, point the way to greater safety and will stimulate confidence on a vastly larger pattern than ever before.

"It will be noted that the term 'air sea rescue' and 'search and rescue' are both used in this article. The former—air sea rescue—is a term conceived in war and applied to the rescue of survivors from incidents peculiar to military operations—especially combat operations. Further, while the orders and operations plans of military commanders stipulated that aid was also to be extended to the survivors of surface-ship disasters, the term was too frequently interpreted to apply to airmen only.

"Thus PICAO, in quest of a more definitive term, adopted 'search and rescue.' At its North Atlantic Route Service conference in Dublin, it was recommended that the term be defined as . . . 'The act of finding and returning to safety the survivors from an emergency incident.' This definition is sufficiently broad to embrace the rendering of aid to survivors of civil and military aircraft and ships . . . and where aircraft are involved, it will apply on land or at sea."

1.2 Theoretical Developments

While the above excerpt provides some valuable insights into the early history of SAR as a national and international, rather than purely local, endeavor, there were other major developments afoot with respect to sensors, survival equipment, rescue equipment and, of particular interest to the present project, the development of search theory and its practical application to operational search planning. The following excerpt is taken from the preface to *Search and Screening* [Koopman, 1980]:

> "The theory of search, as a scientific discipline uniting physical and operational facts through mathematical concepts and theorems, may fairly be said to date from World War II, and to have received its major impulse from various phases of the war at sea, in particular, those involving the submarine. At the center of the development of this new branch of operations research was the group of scientists assigned during 1942-1945 to the Commander in Chief, U.S. Fleet, designated first by the name Anti-Submarine Warfare Operations Group (ASWORG); later, Operations Research Group (ORG). After 1945, it became the Operations Evaluation Group (OEG), and is now a subgroup of the Center for Naval Analyses.
>
> "At the close of the war, it was decided that the OEG should publish those parts of its work of general scientific or historical interest in three volumes: *Methods of Operations Research* (P. M. Morse and G. E. Kimball) [6], *Summary of ASW Operations in World War II* (Allan Thorndyke) [9], and *Search and Screening* (B. O. Koopman) [4]. In each case, the authors named had many collaborators, too numerous to list. These books were published by the Summary Reports Group of the Columbia University Division of War Research under contract OEMsr-1131 with the Office of Scientific Research and Development, and printed and bound by the Columbia University Press. *Search and Screening* was designated as Volume 2B of Division 6 in the series of Summary Technical Reports of the National Defense Research Committee. Originally entitled OEG Report 56 and Confidential, it was later declassified.
>
> "*Search and Screening* contained no references to the scientific literature on search: none existed. During the ensuing third of a century, research and publications on the theory of search and its various military and civilian applications have grown enormously and would fill many library bookshelves. Moreover, the advances in research and development of the systems of detection, their platforms and targets, have been so great that an entirely new picture is presented, both to the military and the civilian applications of the scientific theory of search. And finally, progress in the mathematical theory bearing on search, and making itself all the more effective by using modern computers, has given its own impetus to the subject.
>
> "For these reasons it has seemed to the author and to many of his colleagues that a new and updated edition of *Search and Screening* is desirable; a very practical reason being that the original volume was issued in limited numbers and has long been out of print—in spite of the fact that extracts and translations of parts of it are being used both in the United States and in many other nations giving scientific naval instruction.

“Having decided to publish such a new edition, many difficult questions are raised: To what extent should the contemporary scene influence the treatment? What balance should be struck between the three cardinal factors: (1) physics and engineering developments; (2) mathematical theory; (3) practical recommendations and illustrations (‘how to …’)? …

“Regarding the proper balance between its parts, there is, first, the natural tendency of engineers and experimental physicists to put the instrumentalities of search (the sensors) and the physical properties of the target and its environment (the medium) at the center of the stage, and to reduce to a secondary role *the operation of search as an organic whole having a structure of its own—more than the sum of its parts*. …

“The second danger is overemphasis on immediate practical answers to all kinds of questions of how to plan searches—rules without scientific explanation. There are manuals for this purpose, but they must be based on knowledge and reasoning (often mathematical): to supply such a basis has in fact been the intention both of the first and the present edition of this textbook. On the other hand, every textbook must include a wealth of concrete illustrations, motivated by practical problems, and applying the theoretical results. These are of a very general and simple form in the first seven chapters. In the last three, the practical implementations of the theory developed earlier have been illustrated in greater detail by examples drawn from World War II, principally from the war at sea, although a mathematical framework is given for more modern applications. The three reasons for this choice are that they are *available* (neither classified nor proprietary), they are *detailed* and were put to the actual tests of practice (few similarly detailed examples are available), and they have the *historical* importance of examples of theoretical scientific reasoning that achieved success.

“The third danger facing a textbook on search is the inappropriate handling of the mathematics. At one extreme, there is often an impulse to leave it out as such. But since in most cases the essential reasoning concerns the structural and quantitative aspect of the question—viz., concerns essentially mathematical factors—to leave out the mathematics is to leave out the essential reasoning. … At the opposite extreme, there is the danger of elaborating the mathematical treatment to a degree of generality and abstraction that is not needed, either practically or theoretically, by the applications. …”

“In order to emphasize the link between the world of abstractions and that of physical reality, we have been uncompromising in requiring that all of the terms and quantities be given an *operational definition* … While this notion will be elaborated in the definition of ‘event’ in Appendix A, of ‘compatibility of observations’ in Chapter 1, and in ‘density of searching effort’ in Chapters 6 and 7, and in many other places, we may recall that it is the requirement that the physical preconditions and method of observation or measurement used in defining any quantity entering the mathematical formulas must be stated explicitly—unless these are already known, or derivable by a validated process from known ones.

“In conclusion to the questions of methodology, a word should be said concerning the impact of modern computers on problems of search. When the computers *add* their powers to those of the scientific mind, with its reasoning based on experience, special knowledge, and mathematical concepts and theorems, they are often capable of greatly extending the range of problems that can be solved. But computers are no substitute for the ra-

> tional process. It is indeed for the furtherance of the latter that this book has been written—its results to be implemented, as appropriate, by the former."

The two sources cited above have been quoted at length for both their historical perspective and importance, and because they contain valuable insights germane to the success of any effort to develop tools that support operational search planning.

Koopman was speaking of the "scientific mind" in the last paragraph quoted above, but his comments apply equally well to the "operational mind" of the search planner. Although the search planner need not necessarily be conversant with the concepts and theorems of higher mathematics, he must still use his mind, as a scientist must, in a disciplined, logical, rational manner for which computers and sophisticated software are no substitute. However, the search planner must also be careful not to attempt "manual" solutions to problems so complex they can be successfully resolved on a reliable basis only with the aid of sophisticated computer-based tools.

On the other hand, the development team for a sophisticated search planning software tool does need to be at least minimally conversant with the concepts and mathematics of search theory, as well as other scientific disciplines such as physical oceanography. The team also needs to have at least a minimal level of knowledge about the search planner's operating environment, and topics as varied as sensor and search platform capabilities, search object motion and detection characteristics, development of simulation software for applications of this nature, and other real-world operational and technical issues and constraints.

1.3 Initial Development of Practical Search Planning Tools

The first national-level doctrine on how to conduct search operations appears to have been published by the U. S. Navy Hydrographic Office circa 1944 as "Methods for Locating Survivors at Sea on Rubber Rafts," H. O. 235. This report was reprinted in the November 1944 issue of the *Air Sea Rescue Bulletin* [ASR, 1944]. The methodology presented was apparently based on, "…data collected and compiled for the Bureau of Ships, the Bureau of Aeronautics, and the Hydrographic Office by the Woods Hole Oceanographic Institution and the Scripps Institution of Oceanography of the University of California." It is not known whether Koopman's group was involved in developing the methodology published in H. O. 235, but there are some indications of either its involvement or the involvement of someone with knowledge of the group's work.

Another article on search planning, "Aerial Search—Fact or Fancy?" [ASR, 1946b] appeared in the August-September 1946 issue of the *Air Sea Rescue Bulletin*. This article specifically references the work of the group known first as ASWORG, then ORG, and finally (at that point) OEG. This article seems to be the first that spelled out the elements of the search planning problem in a reasonably complete form:

> "For the purpose of this discussion, an aerial search for a surface target at sea is defined as a visual search by one or more aircraft . . . and the actual target may be an aircraft, vessel or group of vessels, life raft, or a small island.

> "In planning and executing an aerial search—in addition to the desirable factors of simplicity and the utmost in efficient utilization of available facilities and personnel—there are several other basic considerations which will apply, such as:
>
> (a) The type of aircraft available for search, with particular emphasis on speed, range, and endurance.
> (b) Weather and cloud conditions.
> (c) Number and location of bases.
> (d) The type of target.
> (e) Available information concerning position, course and speed of target.
> (f) Urgency of the situation, as influenced by the type of target, number of persons involved, and adequacy of survival equipment at the scene of emergency."

This list is followed by a table showing values for "effective visibility," a measure of "detectability" with a somewhat different definition, and hence different magnitudes for the same sensor (visual search), search object (target) and environmental conditions from those of "effective sweep width." The latter concept may have been still classified at the time as its first published usage was in Koopman's original Confidential 1946 report, *Search and Screening*.

In any case, many of the essential elements of the search planning problem may be found in the paragraphs and list quoted above. These include the amount of effort available ("…type of aircraft available … speed, range, and endurance), optimal allocation of resources ("…utmost in efficient utilization of available facilities …"), the sensor and its detection function ("… visual search…"), search object motion ("Available information concerning position, course, and speed of target"), environmental factors ("Weather and cloud conditions"), survivability considerations, ("Urgency of the situation, as influenced by the …adequacy of survival equipment…), etc. One key element not explicitly listed is an estimate of the probability density distribution on search object location and state. However, it is implicitly included under "optimal allocation of resources" since optimization cannot be done without an estimate of this probability density distribution.

The greatest difference between then and now is that today we have literally billions of times more computing power available at our fingertips as well as vast amounts of previously unavailable accurate environmental data to help us plan searches—assuming we can find way to use them productively. This should not be an insurmountable problem as ways to use computers to address search planning problems other than SAR have already been developed and proven for a variety of search situations ranging from the simple to the quite complex. Simulation technology that can use large and complex data sets to model tactical situations with computers has also vastly improved. Much of this progress has been made in just the last 20 years or so but has not yet been incorporated into SAR search planning tools.

Given that we can use the available computing power and environmental products properly, we no longer need to grossly over-simplify the problem or take questionable computational shortcuts to obtain even a poor solution (as opposed to no solution at all but just a guess instead). We can program the computer to model search situations much more realistically than manual methods ever could and, if done with sufficient skill and care, such computer-based methods should remain reasonably simple to use despite their internal sophistication and complexity. However, there is no getting around the fact that a complex case, such as an overdue vessel or aircraft, will require the search planner to consider and use more data and information, reason in more complex ways, and use the more complex features of the available software.

1.4 Evolution of Manual SAR Search Planning Methods

The U. S. Coast Guard published its first *Search and Rescue Manual* in 1957. The methodology it contained was clearly based on the earlier work of the U. S. Navy Hydrographic Office combined with a then recently published unclassified version of Koopman's work on search theory and its practical application [Koopman 1956a, 1956b, 1957]. In the paragraphs that follow, this is called the "classical" manual method. In 1959 the U. S. Coast Guard's *Search and Rescue Manual* was adopted under the *National Search and Rescue Plan* as the *National Search and Rescue Manual*. The "classical" manual method remained largely intact (except for some important technical corrections) until 1967 when the "minimax" technique was introduced. Although the "minimax" method has undergone numerous changes over the years, it has remained in use up to the present day.

1.4.1 "Classical" Manual Method

A search problem exists when there is uncertainty about the location of an object whose exact position needs to be ascertained. In SAR search problems, this generally means there is a presumption that survivors of a distress incident exist, their location is only approximately known, and there is a need to find and assist them in order to save their lives. If search resources are to be used in the most efficient manner, it is necessary to have some estimate of the probability density distribution of possible search object locations, the detection function relating the expected POD in a region to the amount of effort expended there, and the quantity of resources available for searching so those resources may be deployed in the most advantageous manner.

The "classical" manual method addressed this problem by providing guidelines for estimating an "expected" or mean position for the planned time of a search and guidelines for estimating the amount of uncertainty about this mean reference, or *datum*, position. The uncertainty about the datum position relative to search facilities sent to the scene was characterized and quantified by the total probable error of position. This value in turn was based on estimates of the probable error in the distress incident position, the probable error in the drift estimate, and the probable error in the search facility's position when searching. The distributions of errors about the mean values in all cases were assumed to be of the circular bivariate normal type. The probable error was defined as the radius of the circle centered on the mean value that contained one-half (50%) of the distribution. Thus the probability density distribution on search object location was defined as a circular bivariate normal distribution having a probable error equal to the value computed by the prescribed procedure. Daylight visual search using a pattern of straight, parallel, equally spaced search legs (relative to the search object) was also assumed. The quantity of search facilities available was not explicitly addressed. For the first search it was then assumed that either,

a. The desired probability of success (POS) was 50%,
b. The desired (uniform) coverage over the recommended (optimal) area was 1.0, or
c. The amount of searching effort available was exactly equal to the minimum needed for a 50% POS on the first search or an optimal search for a uniform coverage of 1.0.

Each of these goals produces the same optimal result—a square search area centered on datum with an inscribed radius approximately ten percent larger than the total probable error of the datum position, the area to be searched with a coverage factor of 1.0. Subsequent searches were "optimized" in a similar manner, based on the assumption that all previous searching had been done exactly as the method recommended. A key shortcoming of this method was that it did not provide a general means for optimally allocating any amount of available effort other than levels that were consistent with all of the above conditions. It was apparently assumed that the level of effort needed for a coverage 1.0 search of the recommended area would always be available and used.

1.4.2 "Minimax" Manual Method

In 1967 an amendment to the *National SAR Manual* published a method for handling a type of uncertainty for which, apparently, it was assumed the "classical" method was inadequate. At first this added uncertainty revolved around whether a drogue, or sea anchor, was deployed from a life raft. A raft without a drogue deployed had a relatively high leeway rate while one with a drogue deployed had a slower leeway rate. The maximum difference according to the leeway graphs in use at the time was about 0.6 knots. If the state of drogue deployment was unknown (the usual case), two datum positions and their associated drift errors were estimated. Then a single large circle was drawn around both datums and their respective probable drift error circles and treated as the total probable drift error on an assumed circular bivariate normal distribution. However, it was quite clear that the distribution of possible life raft locations stemming from the minimax technique was not of the circular bivariate normal type nor did the radius of the large circle represent the probable error of the drift estimate. Nevertheless, the approximation was probably still close enough for practical use in most cases.

Although leeway for most objects was originally considered to be in the down wind direction, actual leeway experiments began to show that most objects tend to have leeway at some angle to the right or left of the down wind direction. This angle was called leeway divergence. At some point it became standard practice to treat leeway divergence, rather than leeway rate, as the minimax variable. Since some objects had maximum leeway divergence angles as great as 60 degrees, separation rates, and hence the separation distance, for "left" and "right" datums quickly became quite large, often more than double the drogue vs. no drogue situation. Drawing a single large circle as before produced huge search areas. Such circles clearly contained far more than 50% of the possible search object locations and the distribution of possible locations even more clearly was not of the circular normal type.

Another major problem was that the procedure for computing subsequent drift vectors for the second and subsequent searches was never clearly defined. This was probably because it could be demonstrated that any procedure that was chosen could produce illogical results in some (realistic) circumstances. This is not surprising since “minimax” was an attempt to integrate a method that worked with the “extremes” of one variable (but not those of other equally important variables) with a method that was based on mean values for all variables.

There were many other reasons why the minimax method never worked very well. The fundamental flaw was that it attempted to extend a manual method to problems whose complexity and proper solution lay well beyond the capability of a manual technique.

1.5 Evolution of Computer-based Methods

The Coast Guard’s first computerized search planning tool was implemented about 1970. In the years since, the Coast Guard has had two types of such tools: those based on the manual method current at the time of implementation and those based on a Monte Carlo technique.

1.5.1 Search and Rescue Planning (SARP)

About 1970 the Coast Guard introduced its first computer-based search planning tool—the Search and Rescue Planning (SARP) program. SARP was basically a computerized version of the manual method then in use with some additional capabilities made possible by a computer. For example, databases of environmental data (wind and current) were made automatically available in digital form. The user could also input wind and current values to supersede those in the databases. The drift motion update algorithm computed intermediate datum positions every hour. For gridded data, the values for an intermediate datum position and time were obtained by interpolation. For user-supplied data, which did not have to be on a regular grid, environmental data from the location/time nearest that of the datum position was used for each step. Although the (teletype) user interface was very crude, for the times the tool was fast, reliable and as accurate as the data and computing only two (minimax) mean drift trajectories would allow.

1.5.2 Computer Assisted Search Planning (CASP)

The Computer Assisted Search Planning (CASP) system was first fielded for general Coast Guard use in 1974. This tool was a sophisticated Monte Carlo simulation that could account for simultaneous uncertainties in almost all of the major variables of the search planning problem. Like SARP it had automatic access to digital environmental databases. CASP worked by distributing thousands of simulated search objects (called “replications”) around an incident position, along a track or in an area, and then computing an independent drift trajectory for each by drawing independent random samples from each of the environmental and drift parameters according to their respective uncertainty distributions. The result was a “probability map” like that shown in Figure 1-1 below. It was a visual representation of the computed probability density distribution at the time the search was to be conducted. These “maps” could be crudely scaled within the mechanical limits of the teletypes (and later any textual output device connected to the user’s workstation) to match the scale of the search planner’s nautical chart. Dividing cell values by 100 yielded probabilities in percent. Thus the value 325 represents a 3.25% probability that the search object is in the center cell of the top row.

16	33	325	97	42
51	145	744	537	333
243	392	777	750	515
348	443	327	629	549
212	607	333	200	152
140	388	292	205	175

Figure 1-1: A CASP Probability Map

The probability map shown above is an example of a "Full Map." CASP had two other "map" formats. These were a "Quick Map" that showed the entire distribution in a concise format of one character per cell. Four characters were used to represent four sub-ranges of cell values from the lowest to the highest as shown in Figure 1-2 below.

```
--:--
-:X!:
::XX!
:!:X!
:X:::
:::::
```

Figure 1-2: CASP Quick Map

The other type of "map" was called a "Top Map." It was a listing of cells in descending order of probability density along with expected cumulative efforts and POS values for covering the cells in the listed order at coverages of 0.5, 1.0, and 1.5.

Due to the natural variability of winds and currents over areas of the ocean, the computed CASP distributions were quite generalized and showed the inadequacy of the circular bivariate normal assumption used by the manual methods and computerized versions of them. CASP made many other shortcomings of the (necessarily) oversimplified manual methods evident as well. CASP also had the capability to compute an optimal search plan, although at first such plans were not always operationally feasible. Finally, CASP could update the probability density distribution to reflect the effects of previous searching and provide values for the probability of success (POS) for any given search and the cumulative probability of success (POS_{cum}) for all searching done to date. This was the first time that SAR search planners had access to POS values for search planning and evaluation purposes.

Both SARP and CASP remained in use while being gradually improved until about 1982 when a major change in the Coast Guard's central computing center hardware was effected. Between 1982 and 1985 a hybrid tool that borrowed from both SARP and CASP was in use while CASP was re-written, but no trace of that hybrid software now remains in the Coast Guard. The new version of CASP became operational around 1985. The rewrite did an excellent job of addressing many software engineering issues and an improved optimal effort allocation module was developed and installed with contractor assistance. However, the rewrite did not advance CASP's basic design to address and remove certain important deficiencies or take advantage of new developments in search theory and algorithms to implement them. Although it has undergone a few changes, this version of CASP has not been under a program of continuous improvement like its predecessor—but it is still in use.

1.5.3 Automated Manual Method/Automated Manual Solution (AMM/AMS)

About 1995 a new computerized version of the manual method was developed and fielded as the Automated Manual Method (AMM). This software was combined with the Geographic Display Operations Computer (GDOC) software that was a geographic information system (GIS) for the Coast Guard's new IBM PC-compatible workstations. It's search planning solution technique was a far more literal rendition of the manual method than SARP had been, and it took little advantage of the available computing power or available digital environmental data products to make improvements in the solution itself. However, it did provide the user with a far more useful and powerful interface and geographical display capability than any previous system had done. These interface and display capabilities were also provided for CASP, although the behind-the-scenes interface between GDOC and CASP was extremely primitive—in fact it was a classic example of a kludge. Nevertheless, having the CASP probability maps plotted in color and overlaid onto nautical chart images is a powerful and concise way to combine CASP's previous three map formats into a single easily understood and information-rich presentation like that shown in Figure 1-3 below. Although not shown in Figure 1-3 due to limitations of scale, each cell's probability of containing the search object could be displayed as a numeric percentage in addition to the cell being color-coded. About 1998 AMM and the CASP interface (still a kludge behind the scenes) were ported to a new GIS environment called the Command and Control Personal Computer (C2PC) and the new version was fielded in mid-1999 as a replacement for GDOC/AMM. The AMM portion was renamed the Automated Manual Solution (AMS). Unlike CASP, AMS has been under a program of continuous improvement since its inception and new, improved, versions are fielded on a regular semi-annual schedule. However, its solutions to the search planning problem remain quite primitive. The CASP interface portion has remained "frozen" in place for the most part. There appears to be no mechanism for developing improvements that require changes to both the C2PC CASP interface and CASP (1.x) itself.

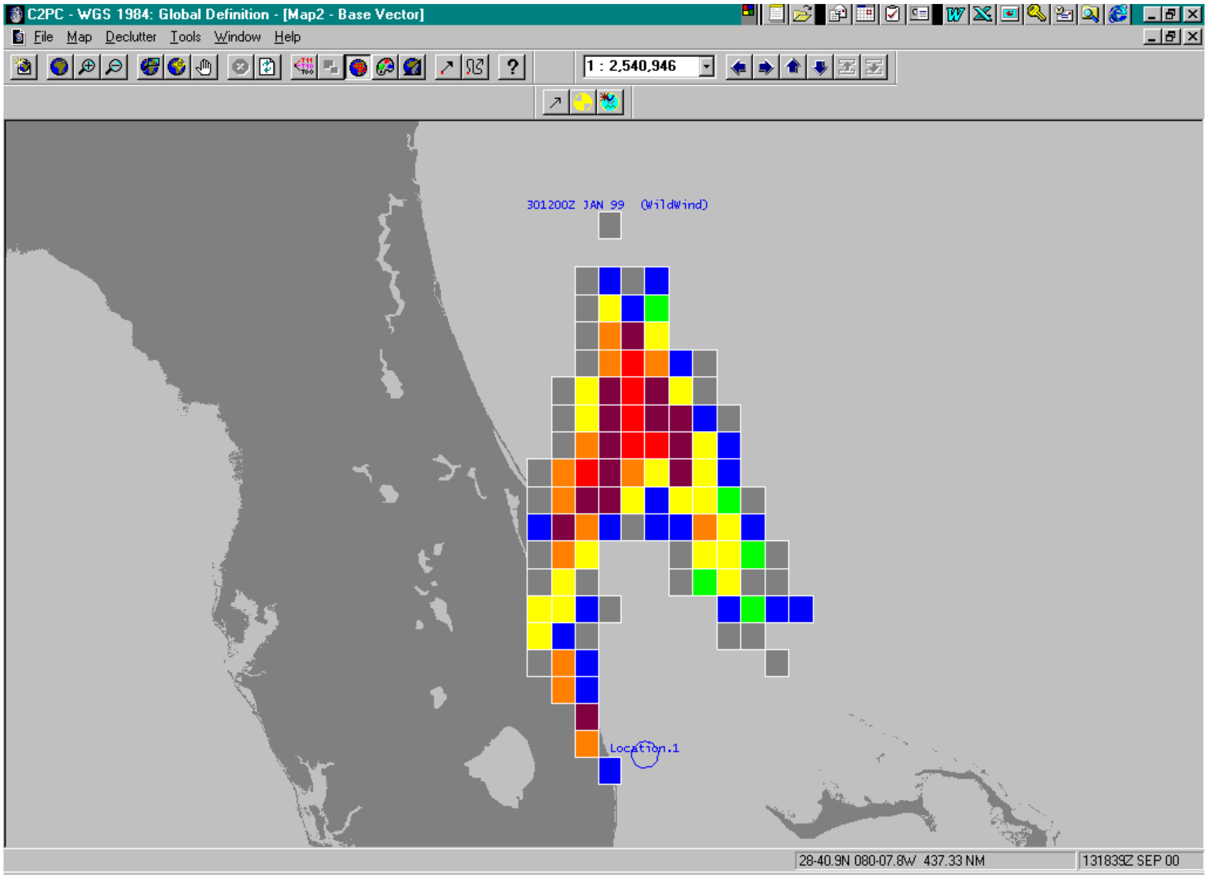


Figure 1-3: C2PC/AMS Rendition of a CASP Probability Map

2 Search Planning

2.1 Planning Operations

In general, there are several reasons for planning an operation rather than just issuing a general objective and directing assets to achieve it. These include, but are not limited to:

a. Making efficient, effective use of limited available resources.
b. Maximizing the probability of success.
c. Ensuring an adequate level of safety.

Planning is so important to achieving these goals that many operations require days, weeks, months or even years be devoted to the planning of activities that will be completed in much shorter lengths of time once they actually commence.

2.2 Special Aspects of SAR Search Planning

SAR search planning is essential for all of the same reasons for which the planning of any operation is essential. However, SAR search planning has some additional and in some cases more stringent requirements.

The SAR system is necessarily reactive in nature. Distress incidents are not planned in advance but it is the job of the SAR system to respond efficiently and effectively when alerted. Much of the planning needed for activities following an alert may be addressed through training requirements, readiness posture, standard operating procedures, and doctrine. When searching is required, these are still necessary but they are not sufficient.

A significant and continuing investigative effort must be mounted in situations where searching is required. The goal of such an investigation is to minimize the uncertainty as to the location and state of any survivors. This information can then be used to more effectively deploy search facilities and locate survivors more quickly. However, it is still necessary to make carefully considered assumptions to fill gaps in the available information and develop likely scenarios describing what may have occurred. Scenario development and analysis requires careful thought, making it an essentially human part of the investigative activity that cannot be performed by computers, although some powerful computational aids can be provided. When the case as a whole is viewed as an investigative effort, searching is just one more investigative tool, albeit the most complex and expensive one.

Unlike many carefully planned operations, the time available for planning a search is almost always much less than the time required for completing it. A few hours at most are available for planning purposes, and often the planner is also responsible for additional investigative efforts, other duties related to the case, and still other duties that are not related to the case. This makes time for planning a very scarce commodity.

The goal of search planning is to maximize the probability of successfully locating the survivors of a distress incident while it is still possible to save them. A search plan that maximizes the probability of achieving this goal is known as an *optimal survivor search plan*. However, allocating the available resources optimally is difficult and it requires a great deal of computation. Hence there is a need to bring a significant amount of computing power to bear on the problem in a way that allows easy use and quick results.

2.3 Search Planning Process

The first step in defining the requirements of a search planning tool is understanding the problem that the tool is supposed to help the planner solve. Like most problems, a thorough understanding requires that it be viewed from several perspectives and levels of detail. One of these perspectives is a "process view." At a high level of abstraction, the search planning process consists of the following steps:

1) Create a Case when Alerted.
2) Gather Data and enter it into the Case folder or database (investigation). Revise as needed.
3) Enter Assumptions about needed but unavailable data elements into the Case folder or database. Revise as needed.
4) Analyze the Data and Assumptions from steps 2) and 3) to Define and Weight Scenarios. Review frequently and Revise as needed.
5) Estimate "initial" Probability Density Distribution(s) (PDD) based on Scenario definitions. Re-compute as needed based on revisions in previous steps. (Note: Pre-distress Motion, interaction with Hazards, initial (distress incident) State transitions, etc., should be included in the "initial" PDD estimation process.)
6) Estimate PDD for next search (based on "initial" PDDs from Scenarios, post-distress Motion (drift), probable post-incident State changes, previous Searching, etc.) This may require re-evaluating all activity to date, depending on revisions, if any, to data from previous steps. Often steps 2) – 4) will provide sufficient information the first time and will not be significantly revised in ways that affect the remaining steps as the search progresses. In other cases, proper revision of the "initial conditions" may be crucial. The "Review" process of step 4) is vital and necessary to prevent the condition known as "scenario lock" where the search planner pursues one scenario to the exclusion of others that are also plausible.
7) Estimate resource availability and capability (available Effort) for the next search.
8) Plan the next Search (usually an Optimal Survivor Search (OSS) is desired).
9) Promulgate the Search Plan.
10) Execute the Search Plan.
11) Evaluate the completed Search based on actual search activity and search conditions.
12) Repeat steps 2) – 11) until all Survivors are found and rescued or until active search is suspended pending further developments.
13) Close or "suspend" the Case.

3 Modules of a Search Planning Support System

3.1 Elements of Optimal Search Planning

Koopman defined the elements of the basic problem of optimal search. They are:

- A probability density distribution on search object location and state (so the probability of containment, POC, for any subset of the possible locations and states can be estimated),

- A detection function relating the probability of detecting (POD) the object if it is in a searched area to the density of the searching effort expended there,

- A known finite amount of available searching effort, and

- An optimization criterion of maximizing probability of finding the object in a desirable state (probability of success or POS) subject to the constraint on effort availability.

Given these elements, it is possible to develop an optimal search plan. However, translating the facts and assumptions about the case into a probability density distribution on search object location and state may be a quite complex undertaking. Such distributions are the combined result of many, often interrelated, factors. Frequently the best way to address this complexity is with a computerized Monte Carlo simulation.

3.2 Subsystems and Modules

A search planning support system should have several major modules. These are

- A User Interface
- A SAR Scenario/Search Simulator
- Environmental Database Support

These major modules are actually subsystems that each must contain numerous modules and sub-modules. The present CASP system, which has a very limited user interface that forms only a small fraction of the total software package, has about 400 modules. However, it would not be productive to list those modules here, as the new tool will undoubtedly have a different design and one that is yet to be produced. Nevertheless, some general suggestions as to the more important modules can be made. These are listed below in a hierarchical fashion. This is an incomplete list and it is to be expected that the final design will produce a different arrangement than the one shown here.

User Interface Subsystem
- GIS/GUI (preferably with two screens—one for text, one for graphics)
- General Databases & geographical overlays for information of general SAR interest
  - AOR boundaries, points of interest, etc.
  - Past SAR Case Summary Data
  - Search objects and their characteristics
    - Object types
    - Motion attributes
    - Detection attributes
    - Survival attributes
  - SRU configurations and capabilities
    - Types
    - Speeds
    - Endurances
    - Sensors
    - Operational constraints/restrictions
- SAR Case Database & Associated Scenario Specification Tools
  - Checklists & Data Entry Tools
  - Scenario definition tools
    - Position/time and associated uncertainties (includes LKP + DR Cse/Spd/Distress Time, LOP from flare sighting, DF bearing [either from distressed craft or from station that hears distressed craft], Other LOPs, etc.)
    - Intended Track and associated uncertainties (positions/times/speeds, etc.)
    - Intended Operational Area and associated uncertainties (vertices, duration of operations, characteristics of craft motion while operating in the area)
    - Any realistic combination of the above (both positions and areas may be origin, waypoints, or destination of an intended track)
    - Hazardous Areas (vertices [some edges may be arcs, e.g. hurricanes], times, motion characteristics, probability of causing distress)
    - Possible Post-Distress Search Objects
      - Post-distress motion (leeway) characteristics
      - Detection characteristics
      - Survival characteristics
      - Other state change characteristics
  - “Theme” or “view” display tools
    - Geographic displays in various chart formats
    - Probability maps keyed to scenarios, search objects, states, etc., scaled and formatted to conform to the underlying chart/map display.
    - Effort/coverage maps (optimal, planned, and actual)
    - Environmental and Drift data display
  - Search Facility definition tools
    - Type
    - Location(s) (Staging/Recovery)
    - Endurance
    - Search Speed(s)
    - Sensor(s)

Search plan definition/display/modification tools
Search areas (geographic plots and textual specifications)
Search patterns/tracks (geographic plots and textual specifications)
Search effects (expected POS consequences of different plans/resource allocations)
SRU search activity description/display tools
Actual or estimated SRU tracks, speeds and times for each sortie
On scene conditions
Search Results
Communications module (human-readable & digital I/O as appropriate)

SAR Search Simulation Subsystem
Case database and data acquisition module (input module – obtain pertinent data from User Interface Subsystem)
Case-dependent "local" environmental database/module (Each case should have its own "mini" database for environmental data, even if no "local" data is available. If "local" data is available, this module should properly integrate it with the available "global" data. This is where the Ocean Prediction System [OPS] fits. If only one ["global" or "local"] is available, then the "mini" database should be built from that source alone. OPS would also be needed to build "mini" databases from "local" data alone.)
Scenario-to-PDD module ("initial" distress positions, dates/times, states – w/uncertainties)
Translate scenario data from the scenario definition tools into appropriate probability density distributions (PDDs) in space and time.
Motion Update module (pre- and post-distress)
Statistical Sampling modules
State Transition module (includes not-in-distress to distressed, survival, etc.)
Resource Allocation module (to produce recommended Optimal Survivor Search Plan)
Detection module (area-based)
Search Planning module (obtain user modifications to recommended plan, provide feedback)
Search Evaluation module (includes relative motion)
Detection module (SRU track-based using lateral range curves)
Statistics and Reporting module (output module – provide data needed to create appropriate displays on the User Interface.)

Environmental Data Support Subsystem
Case-independent "global" environmental database and data acquisition subsystem
Surface (10m) Wind, total surface water current (top 1-2 meters of water column), land, hazards, sea temperature, air temperature, sea state, etc. obtained from outside resources (e.g., FNMOC, NOAA, etc.) on a regular basis. Includes first and second order statistics (means and variances, standard deviations or probable errors). Also includes putting data in most efficient format for the search planning tool's usage.

4 Algorithms for a Search Planning Support System

Many and varied algorithms are required for a search planning support system. Algorithms are needed from a number of scientific and technical disciplines. Knowledge of these disciplines, along with basic operational knowledge about various types of vessels and aircraft, SRUs, operations in the marine environment, etc. is also needed. Topics include computer-based simulation and modeling technology, physical oceanography, search theory, optimization techniques, probability, statistics, random sampling, numerical methods, mathematics, cartography/navigation, sensor performance, standard operating procedures, capabilities and limitations of SRUs, general characteristics of the marine environment and operating in it, etc.

The following list contains a number of technical and algorithmic issues that need to be carefully considered when constructing a search planning support tool. They are not listed in any particular order since there is no natural ordering of these issues. All aspects of the search planning support system must work together simultaneously.

Statistical Sampling (General)

- Assumptions about independence and correlation among variables being sampled.

- Stratified Sampling if/when appropriate.

- Geographic Uncertainty Distributions (see also motion modeling)
  - Position
  - Line of Position (over time, e.g., DR; at a given time, e.g., DF bearing)
  - Track Line
  - Area

- Environmental Means and Uncertainty Distributions
  - Representation
    - Means and Uncertainties for a region (cell) of space and time, or
    - Means and Uncertainties for a point in space and time
  - Velocity Vectors (wind, current, etc. in two dimensions—u, v)
  - Scalars (temperatures, sea state, etc.)
  - Coefficients (leeway, wind current, etc.)
  - Given position and time, draw appropriate samples from the environmental uncertainty distributions to produce sample state change and sample total water current and leeway.

- Uncertainty Distribution Types
  - Uniform (minimum, maximum limits)
  - Normal (mean, standard deviation)
  - Bivariate Normal (means, standard deviations, covariance)
  - Poisson (mean, standard deviation)
  - Exponential (mean, standard deviation)

Time-series Sampling
- 100% Correlated ("minimax")
- Uncorrelated (random walk)
- Partially Correlated
  - Relaxation Times
  - Relaxation Distances

Sampling Frequency (based on all of the following taken together)
- Physical nature of the phenomenon being sampled
- Temporal resolution of the data
- Spatial resolution of the data
- Gradients
- Rates of motion of sampling points (e.g., simulated search objects)

Environmental Data Processing
- Conversions from external "product" formats to standard internal model format.
  - Digital formats (e.g., from FNMOC, NOAA GRIB to planning tool format)
  - SRU/other on-scene observations
  - Wind current computations (from same wind data as used for leeway)
  - Integrate data from different sources to produce coherent environmental "picture."
- Display of environmental data for users

Pre- and Post-Distress Motion Modeling
- Underway
  - Realistic voyage simulations from Last Known Position (LKP)
    - LKP and time of position, DR course, speed, time of distress with uncertainty distributions for each.
    - Intended track (sail plan) specified as a series of locations (starting with LKP) connected by rhumb lines or great circles (in any combination) with positional uncertainty distributions for locations and the track legs connecting them (e.g., mid-leg position error).
      - Estimated times (must be in same sequence as positions) with uncertainty distributions for each position, or
      - Estimated departure time and speeds on legs with uncertainty distributions.
    - Operating Area
      - General well-formed polygon, need not be strictly convex.
      - Operating speed (e.g., speed while fishing) with uncertainty distribution
      - Operating characteristics (e.g., continuous motion vs. sprint-and-drift)
    - Encounters with hazards during voyages
      - Static Hazards
      - Moving Hazards
- Adrift (Leeway, Total Water Current)
  - Search object type with leeway parameters and uncertainty distributions
  - Total water current and computed uncertainty distribution

- Navigational Computations
  - Rhumb Lines (Mercator sailing, mid-latitude, etc.)
    - Meridional parts
  - Great Circles
  - Appropriate chart datums (NAD27, WGS84, etc.)
  - Appropriate spheroid adjustment factors
  - Must handle dateline (180E & 180W), poles (90N, 90S), equator (0N & 0S) and prime meridian (0E & 0W) correctly.
  - Must handle various standard chart "projections" correctly.

State Change Modeling
- From Pre-Distress to Post-Distress
- Survival Modeling
- Other Post-Distress State Changes

Generalized Search Optimization (GSO) (Multiple Moving Objects in Multiple States)
- Optimal Survivor Search Plans
  - Moving search objects (Brown's Algorithm)
    - Stationary Search Objects (Stone's [1989] Theorems 2.2.4 and 2.2.5)
  - Search objects that change state
  - Uncertain sweep widths

Relative Motion and Detection Modeling for Search Evaluations
- Replication/SRU Closest Point of Approach (CPA) and Lateral Range Detection Curves

Mathematical Algorithms and related items
- Monte Carlo Simulation Techniques
  - Simulation Clock
  - Combined Continuous/Discrete event simulation
    - Event chains
- Standard Coordinate Systems
  - Internal computation
  - Geographic
  - Display
  - Conversions among coordinate systems
  - Rotation and Translation of coordinate system axes
  - Relative Space (e.g., relative to known or assumed SRU movements)
- Binary Search
- Sorting
- Linear Interpolation (multi-dimensional)
- Cubic Spline curve fitting/interpolation (multi-dimensional)
- Numerical Integration Techniques
  - Romberg Integration, Gaussian Quadrature, etc.
- Convolution Integrals
- Vector Operations
- Random Number Generation

Uniformly Distributed Random Numbers
Normally Distributed Random Numbers
Error Function – erf(x)
Normal Distribution (area under the normal distribution function or any defined portion)
Bivariate Normal Distribution (probability contained within ellipse centered on mean coordinates, probability contained within any specified polygon containing a portion of a specified bivariate normal distribution.)
Statistically correct combined distributions
Determine whether a point (lat, lng) is contained in a (not necessarily convex) polygon with vertices at (lat_1, lng_1), (lat_2, lng_2), …, (lat_n, lng_n)
Compute latitudes and longitudes of vertices of geometric shapes such as regular polygons, rectangles, parallelograms, etc., based on specification of center point, dimensions, orientation, etc. and whether sides are rhumb lines or great circles.
Compute waypoints (lat, lng) for all standard search patterns based on commence search point (CSP), area dimensions, track spacing, direction of creep, etc.
Compute weighted probability maps from files or tables of replication records.

The specialized algorithms and mathematical functions associated with detection models and optimal search effort allocation may be found in Stone [1989], Koopman [1980], Washburn [1996], and various professional/academic journals devoted to operations research, applied mathematics, naval research, management science, etc. Many pertinent references are provided in Appendix A. Those by Stone, Charnes and Cooper, Brown, Washburn, and of course Koopman deserve particular attention.

It should be noted that the Generalized Search Optimization algorithm described by Stone [1989] contains Brown's [1980] algorithm for moving objects, which in turn contains algorithms for stationary objects such as those of Charnes and Cooper [1958] and Stone's [1989] more general theorems 2.2.4 and 2.2.5. For example, Brown's [1980] algorithm for moving objects requires repetitive application of an optimal effort allocation algorithm for stationary search objects. Similarly, the Generalized Search Optimization algorithm uses Brown's algorithm to deal with search object movement while simultaneously using other techniques to deal with changes in the search object's state.

Appendix B contains a list of references on computer-based simulation techniques and high-level programming languages designed to facilitate development of simulation software. The simulation languages found in this reference list are aimed at human-created systems and processes as opposed to natural ones (such as search object drift). They are primarily used for scheduling, queuing, network and other discrete event simulation problems. While it is unlikely that any of these languages would prove especially suitable for this specific application, many of the general simulation principles they embody are applicable and should be studied with care. This is especially true of their treatment of ***time*** via a simulation clock.

As an aside, simulation of natural phenomena such as atmospheric or oceanic circulation is normally done with special-purpose programs written in computationally efficient "scientific" programming languages such as FORTRAN. Simulation of a SAR case involves both discrete events (e.g., distress incident occurrence, SRU movements, etc.) and natural processes (e.g., search object drift). Hence it must carefully combine techniques from both types of simulation.

Contrary to popular belief, FORTRAN is not a "dead" or obsolete language but one that is continuing to evolve. It is moving toward an object-oriented paradigm while retaining its well-deserved reputation for computational efficiency and the ease with which mathematical and scientific algorithms can be implemented in a clear, well-structured manner. It now contains all the data structure and control structure constructors found in other languages and the FORTRAN implementations are generally better in terms of both source code readability and capability. They are also better in terms of encouraging and enforcing sound programming principles and discipline that lead to correct, stable software. C-like capabilities such as dynamic memory allocation and pointers are also available but much less subject to misuse and abuse than they are in C. Ada-like capabilities have also been incorporated, such as the ability to designate subroutine arguments as input only, output only, or both input and output. FORTRAN is the only popular language that is designed to automatically take advantage of parallel processing capabilities when they are available.

Appendix C contains a number of references for mathematical functions, numerical algorithms and navigational computations that provide computational methods and techniques needed by a search planning support tool of the type being contemplated by the U. S. Coast Guard. The standard navigational reference for the U. S. and much of the world is Bowditch [2002].

5 Simulation/Modeling and Time

Most computer simulations attempt to emulate the behavior of a system over ***time***. This makes the treatment of ***time*** a critical issue that is normally addressed by maintaining a **simulation "clock."** A simulation to support search planning is no exception when it comes to the importance of ***time***, although there are some differences between this application and more typical simulations that emulate things like manufacturing processes, traffic flow, financial transaction processing, communications networks, etc. In the more typical cases, "transactions" are generated in time order according to some distribution. Each transaction then proceeds through the simulation until all "events" and processing related to that transaction have been completed, at which time the transaction is destroyed, leaving only the statistics associated with its historical path through the simulation behind. However, events are processed across all transactions in time order to assure appropriate interrelationships among events and transactions are preserved. The key to preserving proper order in the time dimension is the simulation clock. At the end of the simulation, all statistics are aggregated across all transactions and reported when the "stop simulation" transaction is processed.

In the sections that follow, we will describe how a typical simulation works using a very simple example. Although much of what follows will appear to have little direct application to simulating SAR situations, it will serve to establish a basis for reasoning about simulations in general and the handling of simulated ***time*** in particular. Comparisons with CASP will then be drawn by analogy and a significant source of CASP's past and present difficulties will be revealed so they may be avoided in the future.

5.1 Discrete Event Simulation

As its name implies, a discrete simulation views its "world" as a set of specific events that occur at specific times. Continuous phenomena are typically encapsulated between events and only the time it takes for them to occur is simulated, not the details of the phenomena themselves. For example, consider simulating a barbershop with only one barber, Joe. The mean time between customer arrivals (called the inter-arrival time) is described by some probability distribution. For example, the inter-arrival time between customers may be 18 minutes on average, uniformly distributed between ± 6 minutes of the mean value (i.e., inter-arrival times uniformly distributed between 12 and 24 minutes). These values may be assumed, based on empirical data gathered by observation, or estimated from other data such as population density in the immediate area of the barbershop. When Joe is free and a customer is present, the customer "seizes" Joe's services by occupying the barber chair. Joe then cuts the customer's hair, which is simulated as a service time (e.g., 16 ± 4 minutes) drawn from an appropriate probability distribution. Joe's continuous activity while cutting the customer's hair is not simulated. Customers who arrive while Joe is busy go into a queue or waiting line. When Joe finishes serving a customer, that customer departs and the next customer in line "seizes" the barber chair. Note that the only two scheduled times in this simulation are the arrival of the next customer and the departure of the customer currently being served. Although a customer may be waiting for service and will be served upon the current customer's already scheduled departure, the waiting customer does not "know" when that will be. In this sense, the departure from a queue is not scheduled. Hence there are "primary" events that are scheduled and "secondary" events or consequences of scheduled events that are not scheduled in advance. This distinction turns out to be an important one even if its importance is not obvious at this point.

In most of the more typical simulations, the two most common distributions for generating "new arrival" transactions with respect to time are the uniform and Poisson distributions. Each produces some mean time between arrivals with an associated uncertainty distribution about that mean. Transactions then move to "servers" and either "seize" them for some period of time (typical distributions are uniform and normal) or enter a queue associated with the server where they wait until the server is available and then seize it in a first-in, first-out (FIFO) sequence. This involves two "events" – seizing the server when it becomes free at time x (an unscheduled "secondary" event from the transaction's perspective) and releasing it at a later time y (a "primary" event that is scheduled as soon as the transaction seizes the server). More complex simulations allow for prioritized queues and preempting service. Logic for transactions to enter the shortest queue, as in people going to the shortest line at the grocery store, is also available. When finished with one server, a transaction then moves to the next server (or associated queue) that is appropriate or leaves the simulation after contributing to various statistics being computed and saved for output in the simulation report.

5.1.1 GPSS and Event Chains

The operation of programs written in the simulation language GPSS provide a good example of a discrete event simulation process. A GPSS model consists of a number of "blocks" representing various "real world" entities such as waiting lines (queues) and servers (e.g., bank tellers, machining stations in a factory, etc.), control logic, and data gathering procedures. The relationships among these blocks may be represented graphically by a flow chart depicting the path(s) transactions may take through the model. Transactions representing customers or items to be serviced, processed, etc., are generated and move through the blocks. These movements involve events that occur at specific times. To keep all events in the model in their proper relationship, GPSS maintains a simulation clock that keeps the simulated time.

Every transaction is uniquely identified. That is, each has a unique "serial number." Transactions have a number of other "standard numerical attributes" or SNAs. The "user" or programmer can add more. Some of these attributes are reserved for use by the simulation software itself. These include the transaction number, the time of its next scheduled event, the number of the block in the model where the transaction currently resides, and the number of the next block it will attempt to move into. Others may be modified during the simulation and may affect how the transaction moves through the simulation. For example, the first "server" at a repair facility might be an inspector who determines whether the item being inspected requires only a cleaning, a minor repair, or a major overhaul. (Medical triage is another example.) The results of the inspection (selected according to a distribution specified by the modeler) would set a user-defined SNA that would determine which server the transaction was sent to next. The transaction would then either "seize" the appropriate server and a service completion time would be "scheduled," or it would be placed in a queue waiting for that service. The important point is that the simulation clock is the key to the proper sequencing and duration of events. The importance of the simulation clock to proper sequencing and timing for "life-like" simulations such as flight simulators should be obvious.

GPSS also maintains two "event chains" – a current events chain and a future events chain. All transactions with events that are supposed to happen "now" in simulated time are on the current events chain. That is, the time of these events and the time showing on the simulation clock are the same. In addition, transactions in a "blocked" state waiting for an event to occur to another transaction (e.g., waiting for Joe to finish serving the current occupant of the barber chair) so they can move forward to the next block in the model are also on the current events chain. Transactions whose next event has been scheduled for some later time are on the future events chain in time order. Transactions having events with the same time appear in the order in which they were added to the current events chain or future events chain respectively. After all the transactions on the current events chain have been processed, the simulation clock is advanced to the time of the first event on the future events chain. All transactions with events that are supposed to occur at this time are moved from the future events chain to the current events chain for processing. This procedure is repeated throughout the simulation. As new events are scheduled during the simulation, they are inserted into the future events chain so as to preserve the time ordering unless the generated time is the same as the time showing on the simulation clock. In the latter case, the event is placed at the end of the current events chain and processed before the future events chain is examined.

At any given point in ***real time*** there are two views of the model. One of these is the "block view" showing which blocks contain transactions and what their states (attributes) are. The other is the "chain view" showing which transactions are currently being examined for processing (the contents of the current events chain) and which are "scheduled" for future processing (the contents of the future events chain). A transaction may occupy only one block and be on only one chain at any point in ***real time***. In ***simulated time***, transactions are moved between the two chains without advancing the simulation clock. They may also be moved through multiple "zero delay" blocks in the model without advancing the simulation clock. While the nature of single CPU computers does not permit simultaneous processing of transactions in real time, simultaneous events are allowed in simulated time. This has potential for causing conflicts in simulated time, but tools are provided to resolve such conflicts in a realistic manner. An example is provided in section 5.1.2 below.

> *Note: Negative ("backwards") times are not allowed in GPSS. However, for a SAR search simulation, when debris is found and identified as belonging to the missing craft, there is often the need to estimate distributions of when and where its drift could have originated in order to get a better estimate of the distress incident position and time. This issue will be visited in due course.*

5.1.2 A Simple Simulation Example

To give an example of a simple single-server simulation, consider the model of Joe's barbershop shown in Figure 5-1 below.

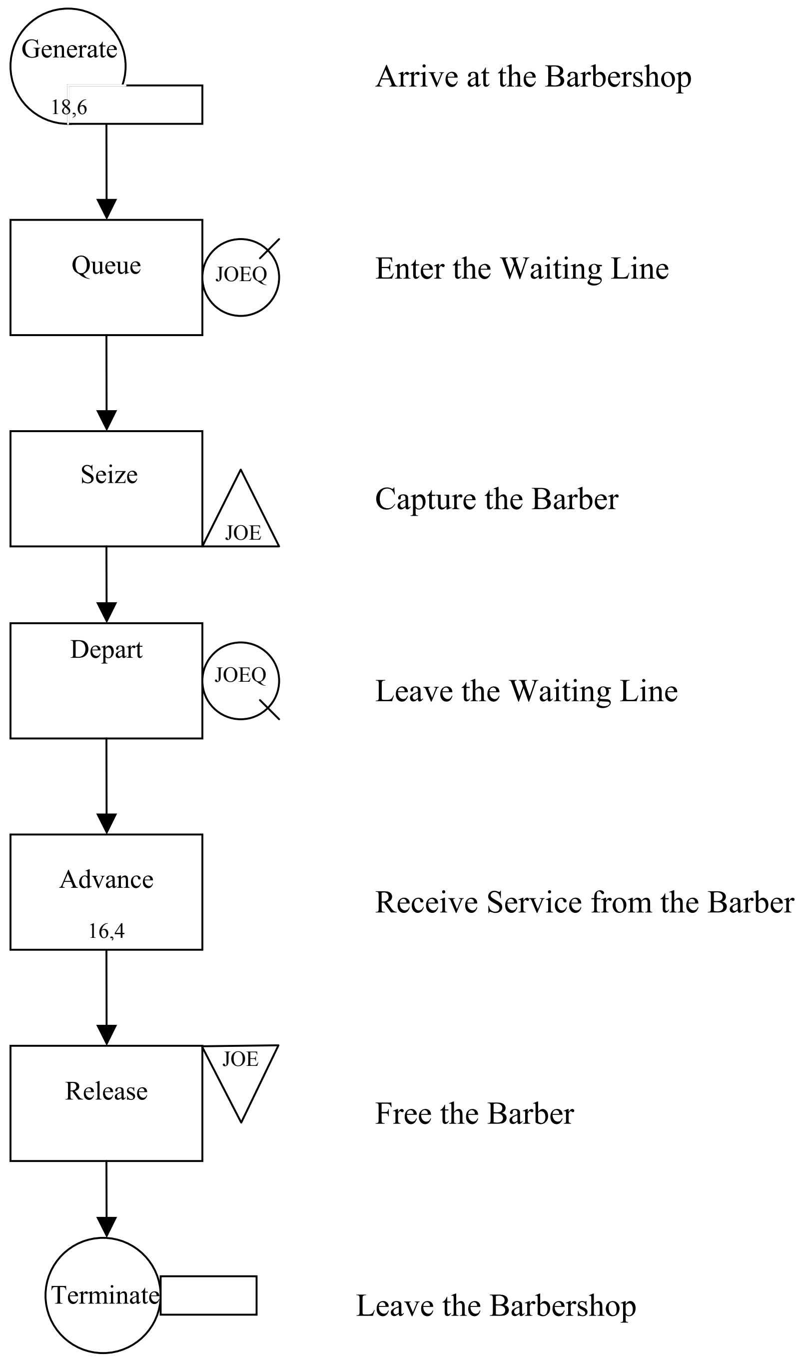


Figure 5-1: GPSS Model of Joe's Barbershop

Assume the simulation clock is initially set to 0800 in Joe's barbershop example described above and that time is measured in minutes. To generate the first transaction (customer), an inter-arrival time is drawn at random from the specified distribution (18 ± 6 minutes), added to the current simulated time (0800) and the transaction is placed on the future events chain. If the inter-arrival time drawn were 20 minutes, then the first customer would be scheduled to arrive at 0820. The simulation clock is then advanced to 0820 and the first customer is moved from the future events chain to the current events chain. The first customer is also moved from the Generate block to the waiting line (Queue block) for Joe in the block view of the model (5-1). The customer's arrival time at the "end" of the waiting line is recorded. Processing is temporarily interrupted to draw an inter-arrival time for the next customer and place that customer on the (now empty) future events chain. Assuming the randomly drawn time between the arrival of the first and second customers was 14 minutes, the second customer would be "scheduled" to arrive at 0834. Now the processing of the first customer resumes and it is moved from the Queue block to the Seize block making Joe unavailable to other customers, then to the Depart block where the its departure time from the waiting line is recorded and on to the Advance block (simulating the customer's time in the barber chair) where it will remain until Joe has finished the haircut. Note that because the simulation clock has not been advanced, the simulated entry and departure times from the waiting line were the same and the first customer would show a waiting time of zero. A service time is then randomly drawn from the specified distribution (16 ± 4 minutes) to simulate the time it will take Joe to cut the first customer's hair. Suppose that time is 18 minutes. The first customer is then moved from the current events chain to the future events chain with a scheduled service completion time of 0838, which places him behind customer 2 on that chain. In the block view of the model at this point in the processing, there is one transaction present and it is in the Advance block representing Joe the barber cutting the customer's hair. Note that although the next customer's arrival has already been "scheduled," that customer is not yet "in" the model. In the event chain view there are two transactions on the future events chain—customer number 2 with an arrival time of 0834 and customer number 1 with a "release Joe" time of 0838. The current events chain is empty.

Since there are no transactions on the current events chain that can be processed "as of" the current simulated time, the simulation clock is advanced from 0820 to the time of the next transaction event at the "top" or "front" of the future events chain. This is the arrival of customer number 2 at 0834. After the simulation clock has been advanced to 0834, customer 2 is moved from the future events chain to the current events chain. Since there are no other transactions on the future events chain for 0834, processing of the current events chain begins. In the block view of the model, customer number two is moved from the Generate block to the Queue block as a new arrival. As before, processing is temporarily interrupted to "schedule" the next arrival. Assuming the randomly drawn inter-arrival time is 19 minutes, the arrival of customer number 3 will "scheduled" for 0853 and it will be placed on the future events chain. The software resumes processing customer number 2 but finds it cannot move this customer forward at the current simulated time because Joe is still busy with customer number 1. Therefore, customer number 2 remains on the current events chain waiting to "seize" Joe's services as soon as possible. There is now one transaction on the current events chain in a "wait state" and two on the future events chain—customer number 1 with a scheduled release of Joe at 0838 and the arrival of customer number 3 at 0853. In the block view two transactions are present—customer number 1 still being served by Joe in the Advance block and customer number 2 waiting for service in the Queue block. Note that customer number 3 is not yet in the model.

Since there are no transactions on the current events chain that can be processed “as of” the current simulated time, the simulation clock is advanced from 0834 to the time of the next transaction event at the “top” or “front” of the future events chain. This is the completion of Joe’s service to customer number 1 at 0838. After the simulation clock has been advanced to 0838, customer 1 is moved from the future events chain to the current events chain. Since there are no other events scheduled for 0838, processing of the current events chain begins. In the block view of the model, this causes customer number 1 to be moved into the Release block, which frees Joe to service another customer and sets a flag to rescan the current events chain before advancing the simulation clock. Customer number 1 is then moved into the Terminate block, which removes it from the model after collecting certain statistics about its passage through the model. At this point there is one transaction on each event chain—customer number 2 on the current events chain waiting for Joe to be free and customer number 3 on the future events chain scheduled to arrive at 0853.

Because the Release block set a flag to rescan the current events chain before advancing the simulation clock, the software goes back to that chain and detects customer number 2 waiting for service. Since Joe is now free, customer number 2 moves from the Queue block to the Seize block where Joe is made unavailable to other customers. It then moves to the Depart block where its departure time from the waiting line is recorded. Note that this customer had to wait four minutes from his arrival at 0834 to the present time of 0838 to be served. Next, customer number 2 is moved into the Advance block where a service time is drawn at random. Assuming this time is 15 minutes, customer number 2 is scheduled to release Joe at 0853 and is moved from the current events chain to the future events chain. This event has the same time as the already scheduled arrival time of customer number 3. Because customer number 3 is already on the future events chain, customer number two is placed behind customer number 3. Now the current events chain is empty. In the block view, only one transaction is present in the model and that is customer number 2 currently being served in the Advance block.

Since there are no transactions on the current events chain that can be processed “as of” the current simulated time, the simulation clock is advanced from 0838 to the time of the next transaction event at the “top” or “front” of the future events chain. This is the arrival of customer number 3 at 0853. After the simulation clock has been advanced to 0853, customer 3 is moved from the future events chain to the current events chain. Customer number 2 is also moved to the current events chain behind customer 3 since it also has an event time of 0853. In the block view of the model, customer number 3 is moved into the waiting line as a new arrival. As before, processing is temporarily interrupted to “schedule” the next arrival. Assuming the randomly drawn inter-arrival time is 17 minutes, the arrival of customer number 4 will be “scheduled” for 0910 and it will be placed on the future events chain. The software resumes processing customer number 3 but finds it cannot move this customer forward at the current simulated time because Joe is still busy with customer number 2. Therefore, customer number 3 remains on the current events chain waiting to “seize” Joe’s services as soon as possible. Processing proceeds to the next transaction on the current events chain, customer 2. Because the simulation clock shows 0853 and customer 2 is scheduled to release Joe at that time, customer 2 is moved from the Advance block to the Release block where Joe is made free to serve another customer and a flag is set to rescan the current events chain before advancing the simulation clock. Customer 2 is then moved to the Terminate block where it is removed from the model after contributing to the various statistics being gathered for the simulation report.

Because the Release block set a flag to rescan the current events chain before advancing the simulation clock, the software goes back to that chain and detects customer number 3 waiting for service. Since Joe is now free, customer number 3 moves from the Queue block to the Seize block where Joe is made unavailable to other customers. It then moves to the Depart block where its departure time from the waiting line is recorded. Note that this customer did not have to wait for service as far as the simulation clock is concerned. Next, customer number 3 is moved into the Advance block where a service time is drawn at random. Assuming this time is 12 minutes, customer number two is scheduled to release Joe at 0905 and is moved from the current events chain to the future events chain ahead of the scheduled arrival of customer 4 at 0910. Now the current events chain is empty and there are two transactions on the future events chain. In the block view, only one transaction is present in the model and that is customer number 3 currently being served in the Advance block. Note that Joe will show an idle period from 0905 to 0910.

5.1.3 Important Features of the Simulation Example

Perhaps the most striking feature of the above example is the extremely careful and disciplined treatment of ***time*** using the simulation clock. No event was allowed to proceed until all previous "scheduled" events in simulated time had been processed. Although computers are generally not capable of processing multiple events simultaneously in real time, provisions were made for processing simultaneous events in simulated time. At a given point in simulated time, transactions with events scheduled for that time and transactions in a "wait state" were handled one by one and each was moved as far as possible through the model until a "blocking condition" prevented further progress. However, no simulated time passed until all events that could be processed "as of" the current time showing on the simulation clock were processed. Two types of blocking conditions were shown. One was when the transaction captured a server and some period of time was required to perform the service. The other was when a transaction had to wait for service because the server was occupied. In the first case a completion time for the service was "scheduled" and the transaction was placed on the future events chain, properly sequenced with other events already there. In case of a tie, the transaction was placed behind any already on the chain with that same next event time. In the second case, the transaction simply remained on the current events chain waiting for the blocking condition to be removed. Logic was provided to ensure proper processing of simultaneous events.

Although a great many details were left out of the above simple simulation example, as were many GPSS modeling capabilities, it should be clear that simulating real-world behaviors requires that simulated events occur in the proper order in simulated time and that the state of the system be displayed only before or after all events that are supposed to have occurred as of a given time (i.e., all "simultaneous" events) are processed, and not while only some fraction of the "current" events have been processed.

Another item to note is that in real life, many processes are continuous and are not made up of a series of discrete events. However, a discrete event simulation can still be quite realistic by dealing with continuous processes in two ways. One way is to encapsulate them between discrete events. For example, the barber's actions while cutting a customer's hair were not simulated—just the passage of time required to perform that service. The second way is to approximate continuous processes by using short, usually regular, time steps and compute the changes in the transaction's attributes (or "transaction state") that take place during each time step. In this way, the beginning and end of each time step become "events" in their own right.

Although the above example is very simple and probably not very useful, some interesting statistics might still be gathered. One would be the average number of customers served per (eight-hour) workday. Others would be the average amount of time customers spent waiting to be served and the longest time a customer had to wait to be served. The amount of time that Joe was idle could also be reported. In a more sophisticated model simulating, for example, a complex manufacturing process, such statistics can be used to identify and correct bottlenecks and under-utilized "stations" in a factory design well before the factory is built, thereby avoiding design errors and saving large sums. Simulations of this type have proven very valuable to both government and industry. Although GPSS does not provide the tools needed for SAR simulation, our simple barbershop simulation does provide clear and simple examples of certain principles all simulations need to observe.

5.2 Use of Time in CASP

CASP has not treated ***time*** in the careful, disciplined way that most simulations do. This has been a significant source of logical errors in the CASP software. It has also created significant difficulties for certain types of desirable enhancements and even made some of these virtually impossible. It should be possible, for example, to simulate a vessel's voyage in the absence of a distress incident and produce a probability map for any point in time during the voyage that describes what is known about the vessel's location at that time based on the data input by the user. CASP cannot do this now because it first uses an assumed spatial distribution to select locations for distress incidents to occur and then "retrofits" distress times to those locations, rather than simulating the progress of the voyage and distress incidents in the same sequence in simulated time as they might have occurred in real time.

Another problem for CASP has been simulating simultaneous events of more than one type. For example, some hazards, such as storms, are in motion at the same time as the missing craft. CASP has no mechanism for simulating when and where encounters between craft and storms might occur. In general, CASP has not helped the user determine when and where a missing craft may have become distressed. Instead, CASP has required the user to make such determinations and even then it has provided only limited tools for describing the probability density distribution of possible distress positions and times. In a similar vein, search objects, represented in the simulation by entities called "replications," and search facilities are both moving constantly during a search. However, CASP cannot simulate when and where opportunities to detect specific replications occur or what the probabilities of detection are on those occasions.

In CASP, the objects called “replications” correspond to what GPSS calls “transactions.” Each replication in CASP represents an instance of the search object (but not while it was attached to the missing craft that “spawned” it) and each replication’s history represents one possible example of the actual search object’s history since it went adrift.

5.3 The Original CASP

CASP was originally implemented on an obsolete (even in 1970) CDC 3300 mainframe computer with very limited memory (128K 6-bit bytes for program code and data plus another 128K for additional data) and similarly limited disk storage by today’s standards. Even though the replication records were short (24 bytes), CASP allowed up to 44,000 replications. Therefore, a complete replication file could be as large as 1,056K bytes (just over a megabyte) plus a little for the header record. Considerable memory space was also needed for environmental data. Because memory was at such a premium, replication files were processed sequentially. That is, a replication would be created or read in from the disk, processed for the next datum time and the updated record would be written out to a new disk file. Then the next replication would be processed in a similar fashion until all replications had been created or updated for the requested datum time. This was analogous to a GPSS-style simulation where the simulation clock was advanced from one datum time to the next in a single increment, precluding simulation of shorter-lived phenomena, such as search facility movements, that required finer time increments to resolve. The limitations of the CDC 3300 computer placed severe restrictions on how much of the search problem CASP could address and how sophisticated its algorithms could be. Many of the limitations cited below were painfully apparent to the original developers but were necessary compromises in light of the limited computing power then available. Nevertheless, for the 1970s, CASP was an extremely sophisticated and capable search planning tool.

In the original version of CASP, each replication was represented by an ordered triple of values: latitude, longitude, and “Pfail.” These values were equivalent to what GPSS calls “standard numerical attributes” or SNAs. The first two values gave the replication’s position and the “Pfail” value gave the probability that all searching done to date would have failed to detect the object represented by that replication. That is, if the replication represented a life raft that had moved along the same drift trajectory as that represented by the replication’s simulated drift history, the Pfail was supposed to be the probability of not detecting such an object given all searching done to date. The initial Pfail value for each replication was always 1.0 prior to any search updates.

The original version of CASP provided tools for describing where the distress might have occurred and generated replications accordingly. Three types of geometries were provided: possible locations in relation to a point, possible locations in relation to a series of connected line segments (e.g., an intended track), and possible locations within an area defined by a closed well-formed polygon, also represented as a series of connected line segments. Programs POS, TRK and ARA were invoked for each of these respectively. A drift update tool, Program DFT, was provided to compute new replication positions based on previous positions, search object type (for leeway characteristics), and environmental forces from data stored in CASP’s environmental database or supplied by the user. Two search update tools, Programs REC and PTH, were provided to evaluate area and track searches respectively. Finally, a display tool called Program MAP was provided to print probability maps to any desired scale that was within the mechanical capabilities of the teletypes in use at the time.

5.3.1 CASP "Generation" Modules (POS, ARA, TRK)

Three modules were provided to generate replications initially, one for each of the available geometries. Program POS generated replications around a point, Program ARA generated replications over the area of a polygon and Program TRK generated replications with respect to an intended track.

Distributions around distress positions were assumed to be of the bivariate normal type. The user specified the latitude and longitude of the center point, the length of the semi-major axis of the 50% elliptical containment (probable error) contour, the length of the semi-minor axis, and the orientation (direction) of the semi-major axis. Program POS used these inputs to generate replication positions and create an initial replication file. The user also specified the mean time of the distress and this time was taken as the initial file or datum time. All replication Pfail values were initialized to 1.0.

Distributions within areas were assumed to be uniform. The user specified a series of positions designating the corner points of a polygon in clockwise order. Initially, such polygons had to be strictly convex, but this restriction was later relaxed. An improved algorithm was installed to allow any well-formed polygon. Program ARA used these inputs to generate replication positions within the specified polygon. As with a distribution around a point, the user specified the mean time at which the distress was believed to have occurred and this time was taken as the initial file or datum time. All replication Pfail values were initialized to 1.0.

Distributions along tracks were assumed to be uniformly or linearly distributed along the length of the track with a cross-track "offset" that was normally distributed with respect to each of the track's legs. However, the need to correlate the distress or drift start time with the distress position relative to the starting position and time of the track created a problem when it came to assigning a single file or datum time for the output replication file. The means by which Program TRK dealt with this problem is described separately in paragraph 5.3.4 below. All replication Pfail values were initialized to 1.0.

It should be noted that the three generation programs operated independently of one another and there was no meaningful way to combine them even though the software allowed combinations. For example, a voyage from a point of departure to a fishing area could not be simulated even though Program COM would combine the replication files generated by Program TRK and Program ARA into a single file. Combining scenarios is discussed further in section 5.3.6 below.

5.3.2 Replication Files

Replications were stored in flat files as simple (non-linked) lists. Files were named according to the procedure that created them (POS, ARA, TRK, DFT, REC, and PTH) and stored in a directory named for the case. For example, CASE01-243 would be the directory for the First Coast Guard District's case number 243. Each file in the directory represented a "snapshot" of the state of all replications in the simulation at a specific "datum" time. The datum time was made part of the file name using a standard military date-time-group format and it also appeared in the file's header record, along with other data such as the minimum and maximum latitudes and longitudes of the replications contained in the file. DFT051500ZJUN78 would be a drift-updated file as of 1500Z on 05 June 1978. The replication positions within a file represented possible search object positions as of this "datum" time. Each replication file could also represent only one object type. As a search progressed from day to day, the replication records were read in from a file, updated for the effects of searching, updated for the effects of drift, and written to a new file. Thus each replication file used its predecessor from the day before as input. However, there were some very serious shortcomings to the original techniques, as described below.

5.3.3 CASP Drift Updates

The first step in the drift update process was associating a distress time with a distress position for each replication. When a drift update was requested, the user could specify an uncertainty in the amount of time the object had been adrift as an input to Program DFT. For example, a user might specify a drift time uncertainty of two hours. This was interpreted to mean that replications in the input file could have started drifting anytime between two hours before and two hours after the datum time of the input file. For the first drift update, this was tantamount to specifying an uncertainty about the distress time. Thus a two-hour uncertainty about a mean time of 1500Z meant the object could have started drifting at any time within two hours of 1500Z – that is, anytime in the range of 1300Z to 1700Z. Uncertainty about time adrift was not appropriate for use with any subsequent drift updates. However, there was nothing in the software to prevent inappropriate use of the drift time uncertainty feature.

When a drift time uncertainty was specified, CASP would choose an amount of time from a uniform distribution over the specified range (e.g. ± 2 hours) and would add that value to the datum time of the input file. This would determine the replication's drift start time for the update. CASP would then "drift" the replication up to the next requested datum time and write the replication with its new position out to a file for that datum time. This procedure would be repeated until all replications had been processed.

This treatment of ***time*** was at least marginally adequate for the initial drift updates of distributions of distress positions and times around points and within areas, even if the exact time of the distress was not known but it was known to fall within some relatively short interval. In such situations there was no requirement to correlate drift start times with drift start positions. Track distributions presented a more complex problem, however.

5.3.4 CASP Track Distributions

To define an intended track, CASP asked the user to specify it as a series of positions representing connected legs. First, a departure time and a "control" time were entered. This was followed by leeway parameters. Then the departure position and associated probable error were specified. This was followed by the type of sailing (rhumb line or great circle) and the mid-leg probable position error. Then a speed of advance on that leg and an associated percentage error was specified. Finally, the end point of the leg and associated probable error were given. The end point of one leg was assumed to be the starting point for the next leg, if there was one.

The user could also specify a most likely point along the entire track, as a fraction of total track length, for the distress to have occurred. The likelihood of a distress at that point was expressed as a multiple of the likelihood at either end of the track. Entries of 0.5 and 2 would mean that at the midpoint of the track, a distress was twice as likely as at either end. The probability of a distress increased linearly from the origin to the most likely point on the track and then decreased linearly from that point to the end. CASP would then generate a replication by first choosing a position along the track in accordance with the linear "distress likelihood" model just described. CASP would then choose an offset distance perpendicular to the track to get the distress position. The distribution perpendicular to the track was assumed to be normal with a probable error that was found by linear interpolation between the mid-leg position error and the position error of the nearest end point. In other words, CASP would first select a position along the intended track and then select a perpendicular offset distance to the right or left of the corresponding leg. This defined the drift start position for that replication.

Returning to the required input parameters, CASP also required that the user provide a departure time and a "control" time. The control time was the same as a "datum time" for a drift update. A full set of drift parameters (e.g., leeway factors, etc.) also had to be input just as in the case of a drift update for reasons that will become clear shortly.

The next step in Program TRK was to compute a drift start time. This was done by randomly choosing speeds from the specified speed distributions on all legs up to and including the one with which the chosen drift start position was associated. These speeds were then used to compute how long it would have taken the craft to reach the point on that leg used to generate the drift start position via the process just described. This amount of time was added to the departure time to produce the distress/drift start time.

However, before the replication could be added to the replication file for the requested "control" (datum) time, its position at that time had to be computed by "drifting" the replication from the distress time to the datum time. Once all the replications were processed, they would be synchronized and the file time would have some meaning. This requirement for drift computations made Program TRK the largest module in CASP since it had to incorporate a copy of the drift update and associated modules in addition to the modules used to compute drift start positions and times. This also had the major drawback of not allowing any probability maps for times prior to the latest time the distress could have occurred. In other words, CASP operated under the assumption that no search action would be taken until enough time had passed to make a distress incident involving the missing craft an absolute certainty according to the user's inputs. (A similar situation could arise with position and area distributions if Program DFT was given a datum time within the time adrift uncertainty interval. However, this almost never happened in practice.) If a probability map for an earlier time were requested, CASP would return an error message indicating that a map could not be produced. However, CASP also reported the probability that the craft had become distressed at some point prior to that time. This probability was simply the number of replications with drift start times less than or equal to the requested datum time divided by the total number of replications to be distributed along the track.

Another problem arose when the drift time uncertainty feature of Program DFT was used in conjunction with a distribution generated by Program TRK. Since there was no provision for uncertainty in the time of departure, such uncertainty could not be accommodated until the first drift update following the one built into Program TRK. This was tantamount to saying that it wasn't the actual drift start time that was uncertain but the "control" time of the TRK-generated distribution that was uncertain. Unfortunately, this was a poor reflection of reality. Environmental data had its own "as of" times and positions. A two hour uncertainty about the drift start time at the actual starting position was not the same as a two hour uncertainty about how long the replication had been adrift since the "control" time. It also was not the same thing as a two hour uncertainty about the departure time from the track's origin. A more correct simulation would have provided for uncertainty about the departure time and allowed that uncertainty, as well as the uncertainty about pre-distress speeds of advance, to affect the computed drift start times.

5.3.5 CASP Search Plans and Updates

Once all replications were adrift from any of the three types of initial distress location geometries, probability maps based on drift updates could be produced for any desired later time. These "maps" consisted of grids of cells where each cell contained a number related to the probability that the search object was in that cell. For example, a cell value of 1000 meant there was a 10% probability of containment (POC). The search planner then used these probability maps to plan searches. Planners were instructed to ensure the highest probability cells were covered, but few useful tools were provided to help the user allocate the available search effort in a truly optimal fashion. In addition, CASP lacked a simulation clock and logic to keep activities and events in their proper sequence and properly synchronize simultaneous activities. Drifting and searching are simultaneous activities in real life, but CASP could not simulate this situation. This significantly limited one of CASP's primary advantages over previous methods—the ability to account for the effects of previous searching when developing the next search plan.

CASP evaluated the effects of searching by applying estimated POD values to all the replications contained in the search area(s) as of the time of the replication file used as input to the search update module. This meant that CASP search updates worked as if the entire day's search efforts were expended at a single instant in time. (Users were instructed to use the mid-search times as the datum times for both planning and evaluating searches, but there were no checks to ensure this was done. In fact, there was no way, outside of external bookkeeping by the user, to ensure search (or drift) updates were applied to the correct replication files.) In reality, objects do not stop drifting during search activities nor are complete searches carried out at a single instant in time. Both activities occur simultaneously and the relative motion between searchers and search objects during a search is an important factor affecting search effectiveness. However, CASP's treatment of datum times as the only times when events could occur made it impossible to properly simulate the simultaneous motion of search objects and search facilities. As a result, a search that took six hours to complete was applied to a single "snapshot" replication file. If Program REC was used for the search update, then it might apply six hours of searching to the DFT051500ZJUN78 file above to create a REC051500ZJUN78 file that contained replications having the same positions as those in the DFT file with only their "Pfail" values updated. The REC051500ZJUN78 file was then used as input to Program DFT to create the replication file (e.g., DFT061500ZJUN78) and probability map for planning the next day's (06 June) search.

The search update process just described produced Pfail and POS values based on a single average POD for each search area, and made it completely impossible for CASP to show the effects of relative motion between search facilities and search objects. Such relative motion is extremely important as it can easily have devastating effects on search effectiveness, depending on search pattern orientation with respect to the general direction of drift. However, CASP could not show the effects of different pattern orientations since it could not simulate the simultaneous motions of search objects and search facilities.

5.3.6 Combining Scenarios

Program COM allowed users to combine replication files by appending one file to another. Several files could be combined in this manner with a single call to Program COM. The apparent intent of this feature was to allow users to create "non-standard" initial distributions by using the "standard" distributions from Programs POS, ARA, and TRK as building blocks. However, it seems that there were few instances when combining two different probability density distributions made sense in terms of describing real-world scenarios. One such case could have been a situation where a vessel was believed to have been operating in one of several areas when it became distressed, but it was not known which.

On the other hand, the opportunities to create distributions that made little sense or were not actually very useful seemed to abound. Examples included:

- Combining replication files with different file or datum times. The resulting COM file was assigned the later of the two times, so the time associated with the replications in the earlier file was lost. This would result in an erroneous drift update if the COM file were used as an input to Program DFT.
- Combining replication files that represented different search object types. Since all the leeway data was contained in the Program DFT input data set, if the COM file was input to Program DFT, all replications would be treated as the same search object type regardless of how they had been treated previously. The same was true with respect to search updates since all the detection parameters, such as sweep width, were inputs to Programs REC and PTH.
- Any set of replication files could be combined, regardless of whether combining them made any sense in terms of the real-world tactical situation. This provided almost unlimited opportunity for file management errors on the part of the user.
- Since a COM file could (but should not) have contained multiple object types, a probability map of such a file was difficult to interpret and impossible to use for search planning. For example, if two quite different objects such as a PIW and a small boat were mixed in the same file, the probability map would show only which cells were more likely to contain "something" (PIW or boat) but not what that "something" was. Without knowing which cells contained mostly PIWs and which contained mostly boats, planning an effective search was impossible.

No warnings about the potential misuse of Program COM were provided with the original CASP documentation and many file management and case management errors resulted.

5.3.7 Summary of the Original CASP

CASP originally provided that only two types of "events" could occur to each replication—position updates and search updates. Position updates were based on the environmental factors with respect to both space and time and on the amount of simulated time since the previous position update. This effectively simulated the replication's drift motion over the time interval from the last datum time to the current datum time. However, search updates had no associated "event times" of their own but were associated with the datum time of the input file. In effect, searches were simulated as being instantaneous at the datum time of the input file. No mechanism was provided for expressing an SRU's arrival and departure times at/from the scene and simulating its movements in relation to those of the replications during the search. Some other significant features and shortcomings of the original CASP are listed below.

- Programs were provided for distributing possible distress positions geographically but the distress times associated with those positions were not always realistic.
- The time dimension was handled in two quite different ways for drift start times. The "generation" programs POS, ARA and TRK did not address start time uncertainty but a mechanism was provided for this purpose in Program DFT. However, Program TRK was forced to do an initial drift update in order to compute a meaningful "as of" time for the resulting replication file. Hence use of the drift uncertainty time in Program DFT with an input file produced by Program TRK made less than complete sense. The uncertainty in the starting time should have been applied to the departure time at the track's origin. In all cases, the time uncertainty should have been applied in the generation programs, not in Program DFT. Given the lack of individual replication time tags, this would have meant providing "control" time inputs to Programs POS and ARA and incorporating the drift modules into them as done with Program TRK. Doing so would have satisfied the "regularity" principle of good software design and made incorrect usage of the time uncertainty feature less likely.
- No tools, such as the ability to specify the presence of hazards in both space and time and simulate potential craft/hazard encounters, were provided to help the user estimate the more likely locations for a distress to have occurred. The user was expected to make such estimates manually using offline (and labor-intensive) scenario analysis techniques.
- The only time-dependent "event" recognized by CASP was the drift update. This was the only "service" performed on replications that required a "service time" greater than zero in simulated time. All other "services" were performed as if they required no time at all.
- Searches were simulated as being instantaneous and were applied to a snapshot of the distribution as of a specific point in time.
- Replications only moved between "instantaneous" searches, not during actual searches.
- Like transactions in GPSS, replications were processed one by one and moved as far ahead in the model as possible. Unlike GPSS, there were no equivalents of the simulation clock or event chains to ensure events occurred in their proper sequence.
- The primary output was the probability map generated by Program Map. This represented an instantaneous "snapshot" of the probability density distribution on search object location at a single point in time.
- CASP generated a plethora of replication files. Properly tracking and managing these files was tedious, labor intensive and prone to error.

5.4 The Current CASP (1.x)

About 1982 the original computer that hosted CASP was taken out of service. CASP was rewritten and implemented on a much more capable minicomputer. The primary purpose of the rewrite was to better modularize the software, make it easier to maintain, and address at least some of the shortcomings in the original version now that considerably more computing power was available. The USCG undertook the rewrite using its own resources and a number of the original algorithms were either modified or omitted. Most of these changes were, at best, questionable and many of them were incorrect. This version of CASP has come to be known as CASP (1.x) since the level of sophistication was not much different from the original CASP. CASP (1.x) did a better job than the original software in some cases but fared less well in others.

The current version of CASP (1.x) has added a number of additional fields (similar to GPSS SNAs) to the replication record. These include ***time***, object type, scenario identifier, initial location identifier, status, and the time, latitude, longitude and Pfail as of the point where forecast, rather than analysis, environmental data began to be used. The "last analysis" fields are important because updated digital environmental products are received twice daily. Old forecast data is overwritten with newer analysis and forecast data. Planning a future search always requires the use of some forecast data. To keep drift estimates as accurate as possible for subsequent updates, CASP automatically restores the simulation to the state it was in when the previous latest available analysis data expired and runs forward from that point to the desired time using later analysis data and more recent forecasts.

Replications are still stored in flat files as simple (unlinked) lists. However, replication records are now longer (36 bytes) and contain much more information. The availability of a time tag on every replication, as well as the object, scenario, and initial location identifiers, greatly improved the flexibility and usefulness of the simulation. Unfortunately, this alone did not solve all of CASP's problems and it created opportunities for some new ones.

5.4.1 CASP (1.x) Inputs

Users begin a CASP (1.x) case by specifying a case name. This opens a directory of the same name in the user's workspace on the central computer. The user is then presented with a hierarchical menu system for data entry. Under "Input Case Data," the user may enter search object ("target") descriptions, location descriptions, situation descriptions, search area descriptions and environmental data. The first three descriptions provide the information needed to generate replications.

A search object ("target") is entered by first assigning it an identifier that is unique within the current case, such as "Raft1." Then an object type is selected from a menu of choices based on leeway characteristics. This completes the search object description.

A location is entered by first assigning it an identifier that is unique within the current case, such as "GulfStream1." A location may consist of one or more positions. Each position consists of a time of position, uncertainty about that time, the latitude and longitude of the position, and the positional uncertainty as a "containment radius." The time uncertainty is interpreted as ± two standard deviations on a normal distribution. The positional uncertainty is interpreted as 1.5

standard deviations on a normal distribution for the distance from the given position. A uniform distribution over the interval [0,360> degrees for the direction from the given position is assumed. It should be noted that this is ***not*** a statistically correct representation of a circular bivariate normal distribution about the given position. In fact, there is no known situation for which a distribution like the one currently in use makes sense. Comparing the two distributions, one finds that within a radius of one standard deviation, a true circular bivariate normal distribution contains about 39% of the total distribution while the incorrect CASP (1.x) version contains about 68% of the distribution. Thus the CASP (1.x) distribution produces an unrealistically high concentration of positions near the center and an unrealistically low density farther away.

If only one position is entered, CASP (1.x) assumes it is the mean distress position. If two positions are entered, CASP (1.x) assumes the user intended to enter a track consisting of one leg. If three or more positions are entered, CASP (1.x) asks the user whether a "line" (track) or "area" is being represented. Thus CASP (1.x) uses the same three basic location geometries as its predecessor—points, lines and areas.

Once the set of search objects and locations of interest have been described, the user combines a search object with one or more locations to form a "situation." Like search objects and locations, situations are given an identifier that is unique within the case. Locations and situations may each be "weighted." Within a situation, location weights are used to indicate the relative likelihood, in the user's judgment, that the distribution of positions defined by that location contained the distress position. The situation weight is used to indicate the relative likelihood that the given situation describes the actual search object and set of possible distress locations.

This ability to combine several "locations" in a single "situation" and to weight both locations and situations separately provides a great deal of flexibility but it also provides levels of complexity that can quickly outstrip a human's ability to clearly comprehend. Furthermore, there are no constraints to keep such combinations within the bounds of realism. For example, CASP (1.x), like the original CASP, cannot simulate a voyage that includes an operational area such as a fishing area. There is no provision for making an "area" a departure, destination, or "waypoint" of a voyage. However, it is possible to combine a track location and an area location in the same situation even though there is no way to connect them so they logically model a craft's motion from a last known position to the area, its activities in the area, or its return from the area.

Users specify search areas by first assigning them an identifier unique within the case, providing search area boundaries (e.g., by the corner point method), the primary search object identifier (must have been previously defined as described above), the commence search time, the on scene endurance of the SRU, the track spacing used, the SRU's navigational error, and a sweep width for each search object defined for the case.

Users may enter their own environmental data (wind and current) or they may elect for CASP (1.x) to use the data in its environmental databases. In most cases, users enter environmental data only if they have on scene observations available.

5.4.2 CASP (1.x) "Generation" Module (Buildrep)

CASP (1.x) generates all replications using a module called "Buildrep." This module iterates through all the defined situations and calls other modules to generate replications according to the search object and location descriptions used to form each situation. The modules that actually create the replications initially are called Gen_Posn, Gen_Trk, and Gen_Area.

5.4.2.1 Generating Replications Around a Position (Gen_Posn)

The original version of CASP distributed replications around a position in a true bivariate normal fashion. CASP (1.x) uses a different procedure. It assumes the distance from the given position is normally distributed with a mean of zero. The "containment radius" specified by the user is assumed to be 1.5 standard deviations. For a normal distribution, about 87% of the distribution is contained within 1.5 standard deviations of the mean. To prevent negative distances, the absolute value of the randomly selected distance is always used. CASP (1.x) assumes the direction of the selected position from the mean position is uniformly distributed between zero and 360 degrees. Once a distance and direction have been selected, the replication's initial position is computed using a rhumb line sailing.

This technique produces a quite different distribution from a true bivariate normal distribution. In the latter case, the x- and y-components of the replication positions would each be normally distributed. They might be completely independent of one another with identical standard deviations, in which case the distribution would be of the circular bivariate normal type—a subset of the more general elliptical shape. Otherwise, there might be some degree of correlation or covariance that would produce an elliptical distribution with the major axis oriented in some direction between zero and 360 degrees. The distribution currently used in CASP (1.x) has no known physical meaning, whereas true bivariate normal distributions are known to often represent the distribution of position errors about the mean position that can be attributed to navigational inaccuracies and other sources of uncertainty about a position's latitude and longitude.

Once a position has been computed in this fashion, a distress/drift start time is selected. The time uncertainty input by the user is interpreted as two standard deviations on a normal distribution with a mean time equal to the position time entered by the user. About 95% of a normal distribution is found within two standard deviations of the mean.

Now the replication has a position and a time assigned to it. The Pfail value is initialized to 1.0. The search object (target), location, and situation identifiers are all copied to the corresponding fields in the replication record. The replication's status is set to either "adrift" or "aground" depending on whether its position falls on land according to CASP's land location database. Finally, the "last analysis" position, time and Pfail values are initialized to be the same as the initial values just assigned. The replication record is then written to the "INITIAL" replication file.

5.4.2.2 Generating Replications Within an Area (Gen_Area)

In the case of an area, CASP (1.x) connects the points in the order in which they were entered with rhumb lines and adds an additional rhumb line to connect the last point entered with the first point entered. Normally a user would associate the same mean time and time uncertainty with each of the positions used to define an area, but this is not required. In the event these time values are not all the same, CASP (1.x) computes the earliest and latest times from the inputs and interprets these as the minimum and maximum values of a uniform distribution of times.

An initial position for a replication is generated by first enclosing the polygon in a larger "box" defined by meridians and parallels. Using the meridians as limiting minimum and maximum values, a longitude is chosen at random from a uniform distribution between the two meridians. A similar function is performed to choose a latitude at random that falls between the limiting parallels. A procedure is then called to determine whether the resulting (latitude, longitude) position is inside the polygon defined by the user's inputs. If so, it is used as the replication's initial position. Otherwise, a second check is made to determine whether the point is "close" to the defined polygon. "Close" is defined as being within a larger figure defined by the "containment radii" of the vertices of the polygon and connecting external tangents to the circles they define. If this test fails, the position is rejected and new values are chosen until a point is found that passes one test or the other. A time is chosen that falls between the minimum and maximum times associated with the polygon's vertices.

Now the replication has a position and a time assigned to it. The Pfail value is initialized to 1.0. The search object (target), location, and situation identifiers are all copied to the corresponding fields in the replication record. The replication's status is set to either "adrift" or "aground" depending on whether its location falls on land according to CASP's land location database. Finally, the "last analysis" position, time and Pfail values are initialized to be the same as the position, time and Pfail values just assigned. The replication record is then written to the "INITIAL" replication file.

5.4.2.3 Generating Replications Along a Track (Gen_Trk)

In the case of a line (track), CASP (1.x) assumes the points are connected by rhumb lines in the order in which they were entered, regardless of the times associated with each position. CASP (1.x) does not check to see whether the order of the times provided for the positions agrees with the order in which the positions were entered nor does it sort the positions based on their associated times before proceeding. Although a user would normally enter positions defining a track in time order, failure to do so, whether purposeful or in error, would produce unexpected and almost certainly unrealistic results. The means and distributions of speeds on legs could be inferred from the times associated with the end points and the lengths of the legs, but speeds are never actually computed.

A worthwhile question is whether the search planner is better able to estimate a craft's intended speeds and speed uncertainties along its intended track or ETAs and time uncertainties at turn points. Recall that the original CASP computed times for replication positions from the estimated departure time and the speeds and speed uncertainties along the legs of the intended track. Add uncertainty to the departure time as appropriate, and all the information needed for computing the probability density distribution of possible craft locations at any later time is complete. There would be no chance for ETAs at turn points to be in a different order from those of the turn points themselves. For CASP (1.x), search planners almost certainly have to estimate the craft's speed(s) and compute ETAs offline for entry into CASP (1.x) and there is an opportunity for inconsistencies between the order of position entry and the order of the associated times.

CASP (1.x) assumes the probability of distress is uniformly distributed along the track's length. However, Gen_Trk computes the total track length in two different ways that produce different results and neither one makes complete sense with respect to the real world. Prior to choosing a fraction of track from a uniform distribution over the interval [0,1>, Gen_Trk computes the total track length as the sum of the leg lengths plus the sum of all the "containment radii" around the departure, turn, and destination points. This length is multiplied by the randomly chosen fraction of track to get a distance along the track. However when computations are done to determine which of the base track legs is associated with this fraction of the track, problems arise. The first question is whether the distance along track falls either on the first leg or on extensions from either of its ends equal to the radii of the respective "containment" circles. If the answer is "No," then the same question is repeated for the second leg. Prior to July 2002 this had the effect of adding all the turn point containment radii to the total track length a second time. In other words, an along track distance value might have been chosen from the range of zero to 200 NM but the software might have then tried to match it to a range that extended to 250 NM as a result of "double counting" the turn point "containment radii." In effect, the last 50 NM section of the track would have had no replications associated with it, except the single "datum reps" placed at each position used to define the track. This problem was "corrected" by removing the double counting of turn point radii but this alone may not have corrected all of Gen_Trk's problems.

Once the selected distance along the track is associated with a specific leg of the track, Gen_Trk computes a distance along the leg and the fraction of the leg's length that this distance represents. This fraction is then applied to the differences in the times and time uncertainties of the end points to get a mean and uncertainty for the time at which the missing craft would have been at that point. A random selection is made from a normal distribution having this mean and a standard deviation equal to half the value of the computed time uncertainty. Next, the actual position for the replication is computed by selecting an "offset" value from a normal distribution in much the same way as the original CASP, except that a very complex algorithm is used in the vicinity of turn points in an attempt to avoid artificially high replication densities on the insides of turns and artificially low densities on the outsides of turns. Finally, a position is computed for the chosen point.

Now the replication has a position and a time assigned to it. The Pfail value is initialized to 1.0. The search object (target), location, and situation identifiers are all copied to the corresponding fields in the replication record. The replication's status is set to either "adrift" or "aground" depending on whether its location falls on land according to CASP's land location database. Finally, the "last analysis" position, time and Pfail values are initialized to be the same as the position, time and Pfail values just assigned. The replication record is then written to the "INITIAL" replication file.

It is clear that Gen_Trk is fraught with problems and produces distributions in space and time that do not reflect either the user's inputs or reality. It also lacks certain capabilities that were present in the original CASP. These include great circle routes and the ability to vary the probability of distress along the track. Although work-arounds are available for these two deficiencies, they are cumbersome and require additional offline work on the part of the user.

5.4.3 CASP (1.x) Replication Files

Like the original CASP, CASP (1.x) stores its replications in flat files as simple (non-linked) lists. However, unlike its predecessor, CASP (1.x) stores all the replications in the model in a single "INITIAL" file plus a single file for each datum time. This is possible since all the information needed to fully describe each replication is either part of the replication record itself or the record contains a pointer to the necessary additional descriptive information. As a result, a CASP (1.x) case has far fewer replication files to keep track of than an equivalent CASP case, making it less prone to file usage errors by search planners. However, this technique presents a potential problem. Like the original CASP, the datum date-time-group is made part of the file name for files that represent the results of drift updates. Although the "INITIAL" replication file for a case will contain replications with varying (drift start) times if the user has entered any time uncertainties greater than zero, the presumption is that all replications in a drift-updated file will have identical time tags. However, if the requested datum time is less than the latest drift start time, the resulting drift-updated file will contain replications that have yet to go adrift. Nevertheless, if a probability map of the file is displayed, these "future" replications will contribute to the cell probabilities just as if they were adrift at the datum time. The result will be an inaccurate representation of the tactical situation as of the requested datum time. In fact, it is possible to obtain a probability map for the "INITIAL" file even though such a map has no operational use. This capability has led to searches being erroneously planned based on a map of an "INITIAL" replication file—one of the few file usage errors a user can still make.

In short, an "INITIAL" replication file contains no replications that are actually adrift even though all of those not on land have a status of "adrift" assigned. In a drift-updated file for a time greater than the latest drift start time drawn by any replication, all replications not on land actually are adrift. In a drift updated file for a datum time less than the latest drift start time drawn by any replication, a mixture of replications that actually are adrift and replications that are not yet adrift results, even though all those not on land are assigned a status of "adrift." One could define the replications in the "INITIAL" file as scheduled for processing but not yet in the model, as in the GPSS example cited earlier. One could also define all replications with times later than the latest drift start time as being in the model. However, CASP (1.x) now tries to make these distinctions unambiguously only at the file level. A file with a mixture of "adrift" and "not yet adrift" replications makes no sense under the current CASP (1.x) paradigm even though such files may easily exist.

Also like its predecessor, CASP (1.x) replication files have header records containing aggregate information about their contents. This includes the minimum, middle, and maximum replication latitudes and longitudes, the number of replications that are adrift and "landed" as well as the total number in the file, and the minimum and maximum replication times.

5.4.4 CASP (1.x) Drift Updates

Once the inputs have been provided, the user proceeds to the second choice on the CASP (1.x) main menu—"Compute Datum Positions." A subordinate menu then allows the user to choose which source(s) of environmental data to use for the drift update. After that, the user enters a datum date-time-group and CASP proceeds to update all replication positions across all situations for the effects of drift. Since each replication record contains all the information needed to determine what type of object it represents, CASP (1.x) is able to obtain the appropriate leeway parameters on a replication-by-replication basis. Likewise, since each replication carries its last update time, CASP (1.x) "knows" when to start its drift as well. All replications with time tags earlier than the requested datum time have their positions updated for the effects of drift. All replications tagged with times equal to or later than the requested datum time are simply copied to the output file with no changes in their positions, times or status.

Any drift update datum time may be requested, including updates for times prior to the earliest drift start time drawn by any replication. In this case, the "drift-updated" file would be identical to the "INITIAL" replication file since no elapsed drift times greater than zero with which to compute a drift distance would be found. If the requested datum time falls between the earliest and latest drift start times, which cannot be precisely predicted due to their random selection, then the output replication file will contain a mixture of drift-updated replications and replications that are "scheduled" to go adrift at a later time. CASP (1.x) does not inform the user as to whether any of the replications in a replication file have times in the future (although this would be quite easy to do).

Like the original CASP, drift updates are done by processing replications serially one after another. A replication is read from the input file, its position, time and Pfail values are reset to their "last analysis" values, these quantities are then updated to reflect the effects of drift and any searching done since the last analysis time, and the replication is written to the output file. The term "last analysis" refers to the replication's state as of the point in time when the available analysis environmental data was exhausted and the use of forecast data commenced. The ability of CASP (1.x) to automatically return the state of the simulation to the time at which the analysis data originally expired and take advantage of analysis data received since the previous run is a significant improvement over the original CASP. The quality, quantity and resolution of the available environmental data are also much better than they were for the original CASP. However, further improvements in the environmental data available to CASP (1.x) are still needed, particularly near shore.

5.4.5 CASP (1.x) Search Planning and Search Updates

Although the vast majority of the CASP rewrite effort in the 1980s was undertaken by the Coast Guard itself, outside contract assistance was obtained to improve optimal effort allocation and subsequent search plan recommendations. A significantly improved effort allocation algorithm was developed that more realistically reflects the real-world operational constraints under which search craft operate. The user is required to supply sweep width values for all the SRU-search object combinations. The user is also required to designate one of the search objects as the primary search object. The results are near-optimal "myopic" search plans for the primary search object that are also usually operationally feasible. However, these algorithms, like the search update process, are applied to "snapshots" of the distribution at some instant in time. In other words, these plans do not account for search object motion during the search nor do they consider future search efforts should the planned search fail. This latter feature makes such plan "myopic" in that they are optimize only the allocation for the next search without regard to future consequences.

Although CASP (1.x) uses the same basic search update technique as the original CASP, some significant improvements, and omissions, have been made. Since the times associated with each search sub-area are available, the problem of applying a search update to the wrong replication file has been eliminated. CASP (1.x) computes the mid-search time for each search sub-area and then sorts the sub-areas accordingly. It then drift-updates the replication positions for the earliest mid-search time and applies the POD for that sub-area to all the replications contained in that sub-area as of the mid-search time. This procedure is repeated until all sub-areas have been processed. Also, since the inputs to the search update process require sweep width information for every search object type that is part of the case, correct POD values are applied to each replication based on its type. Both features are substantial improvements over the original CASP.

CASP (1.x) has lost the ability to evaluate track line searches. It also does not use the last and most sophisticated algorithms that were available for handling the effects of search craft navigational error that were available in the final CDC 3300 version of CASP. Inexplicably, earlier less capable algorithms are used instead.

Unfortunately, although drift motion is allowed "during" a search, it is limited to motion between mid-search times of sub-areas. This is insufficient for modeling the simultaneous and continuous motion of search objects and the facilities that are searching for them. CASP (1.x) is not significantly better than the original CASP in this regard.

5.4.6 CASP (1.x) Probability Maps and Optimal Search Plans

Recall that although the combining of replication files was permitted in the original CASP using Program COM, such combinations often made little sense when multiple search object types were involved. Such combined files were a source of potentially serious errors if used as inputs to the drift update, search planning or search update processes. CASP (1.x) improvements have dealt successfully with these issues by tagging every replication with sufficient information and asking the user for sweep width data on every SRU-search object pair for every search. However, probability map interpretation remains a problem.

The default mode is to produce a probability map where all search object types are represented at the same time. As before, a cell's probability value then represents the total probability that "something" of interest to the SAR case is in the cell regardless of what the "something" might be or whether several types of "somethings" are all contributing to the cell's value. Therefore, default maps are no more valid than those produced by mapping the combined files of Program COM for search planning purposes. CASP (1.x) optimal search plans get around this problem because they are currently based on a single search object type. However, when plotted on a "combined" probability map that includes all search object types, the plan may not appear optimal. This may confuse users and result in such plans losing credibility undeservedly.

There are two approaches to this problem. The first approach is to produce a probability map with the optimal search plan where the map is based only on the primary search object type. This would ensure agreement between the optimal plan and the probability map, and would also show the user exactly what criterion was used to develop the optimal plan. Although a corresponding probability map on primary search object location is not produced automatically by CASP (1.x), such a map can be produced manually by manipulating the situation weights. In fact, the optional inputs for producing a probability map include adjusting the weights temporarily for mapping purposes. If the weights for all situations except those involving the primary search object are set to zero using the "modify weights" option for probability maps, then a map that corresponds to the probabilities used for the optimal search plan will result. Although this technique for "isolating" a single search object type is described in the CASP (1.x) user documentation, no guidance is provided about when or why a user might want to modify the situation weights in this fashion.

The second approach is to produce an optimal search plan across all situations and their respective weights. Stone [1989] has shown how to develop optimal search plans when the sweep width is uncertain but its probability distribution is known. This could be applied to situations where there is a homogeneous mixture of possible search object types over the region of interest. However, different types of search objects tend to have differing drift rates as well as differing sweep widths. This means that over time they will tend to segregate themselves from one another geographically. Once the segregation is complete, optimal effort allocation again becomes more straightforward. The greatest potential for difficulty arises during the interval between initial homogeneous mixtures in the vicinity of the distress itself and the fully segregated situation at a possibly much later time. This interval is the most important one for SAR. During the "segregation period" there will be a non-homogeneous mixture of search object types. Care in the choice and implementation of optimal effort allocation algorithms will be necessary to ensure non-homogeneous mixtures of search object types are handled adequately.

5.4.7 CASP (1.x) Summary

Like its predecessor, CASP (1.x) has no overall "simulation clock" in the sense described in the GPSS discussion above. CASP (1.x) has also retained the process of first selecting a distress position from a static distribution and then associating a time with the position. For all scenario types, including track distributions, all replication records are generated with positions and times before any drift updating is done. These are placed in a file called "INITIAL" that does not have a specific time associated with it. Unless a replication falls on land according to CASP's land database, it is given a status of "adrift" as soon as it is generated. The "INITIAL" file may be mapped and displayed even though the records it contains span a range of times and no positions have been updated for drift. On some occasions, this file has been mistakenly used to plan a search. Similarly, if the user requests a probability map for a time that is earlier than the latest drift start time, map probabilities that include replications with drift start times in the future will appear. It is also erroneous to use such a map for search planning. Unlike the original version, CASP (1.x) provides no warning of these potential errors. In fact, for both an "INITIAL" map and a map for any subsequent time, CASP (1.x) indicates all replications are already either adrift or aground regardless of the relationship between "replication time" and "datum time." Since a single replication file will contain multiple search object types if more than one has been defined and the default probability maps are based on aggregate probabilities across all types, default maps should not be used for planning searches.

Despite the recent "bug fix," the CASP (1.x) Gen_Trk module likely contains significant errors and in many respects does not represent reality as well as the original CASP module that performed the same function.

In an attempt to use a "one size fits all" method for specifying positions, the function the positions serve and their relationship to one another was not realistically preserved. In other words, CASP (1.x) attempted to use a single data entry format for positional data regardless of its use. On the surface this appears to be a good idea. However, positions used to define the vertices of a polygon that represents an operational area are not related to one another in the same way as positions that represent waypoints along an intended track. CASP (1.x) does not adequately address these differences in that it does not require appropriate relationships among the times associated with such positions.

- In the original CASP, positions used to define an operational area had no associated uncertainties in space or time and all of the probability associated with the area was assumed to reside inside the defined polygon. However, it would have been reasonable to associate a time uncertainty with the area as a whole—a function that was available via the time uncertainty feature of Program DFT. CASP (1.x) is less explicit, but it still defines positions as being "vertices of the enclosed area" when the "area" option is chosen, although this is not strictly true given the way the software actually functions.
- For a craft's intended track, the original CASP required speeds and speed uncertainties on legs as input, thus ensuring the points defining the track were in the same time order as their implicit spatial order derived from the order in which the points were entered. CASP (1.x) requires times and time uncertainties for each point and does not require that the time order be consistent with the implicit spatial order. The original CASP also allowed legs to be either rhumb lines or great circles whereas CASP (1.x) assumes all legs are rhumb lines. Even if the CASP (1.x) Gen_Trk module worked correctly under the assumptions apparently made for its construction, some realism and flexibility have been lost compared to the original CASP.

CASP (1.x) drift updates, with the exception of "timing errors" associated with mixing "current" and "future" replications in the same file, are much better than those of the original CASP due to shorter time steps and great improvements in the environmental data obtained from the U. S. Navy's Fleet Numerical Meteorology and Oceanography Center (FNMOC).

The bookkeeping burden and chances for file management errors on the part of the user have been greatly reduced in CASP (1.x) as compared to the original CASP since time information is a required input for all significant activities. This is particularly helpful for search updates. However, the search updates themselves are little improved since they are still applied to instantaneous "snapshots" of the probability distributions and do not properly account for the relative motion between search facilities and search objects.

The CASP (1.x) recommended search plans are significantly improved over those of the original CASP in terms of both optimality and operational feasibility.

Historically, the techniques that led to CASP's development were first employed with high-cost searches for objects of national importance located on the deep ocean bottom. A great deal of effort and analysis went into establishing, with as much accuracy as possible, where the incident occurred. Since sensors for deep ocean search had small detection ranges and using them involved considerable expense, it was impractical to undertake such a search unless the location of the incident could be established within relatively close limits. Usually the time of the incident was known within close limits and the search area was based on detailed analyses of various long-range sensor data for that time. Together with the fact that objects do not move after sinking to the bottom, this meant that little or no motion modeling was needed. Therefore the pre-CASP techniques concentrated on optimally allocating effort over a fixed region of ocean bottom. This is probably why both the original CASP and CASP (1.x) have the feel of a tool that was originally designed for finding static objects but then had time and motion added as an afterthought.

CASP requires tens of thousands of "transactions" (called "replications") in order to assure an adequate sampling from the various possible combinations of values for parameters affecting the search problem. Often all replications are subject to the same "event" at the same time, such as computing an updated drift position every hour. However, there are situations where it is possible that every replication would have its next "scheduled" event set to occur at a different time. When simultaneous activities of different types must be simulated to determine their proper relationship, such as simulating search facilities searching for drifting objects, proper timing becomes an important issue and a source for large numbers of events, e.g. one event for the closest point of approach (CPA) for each (search leg, replication) pair. Proper timing is also important when simulating a craft's pre-distress behavior in relation to both specific hazards and the general hazard of being underway or in flight. Every replication will have a unique distress event time whenever there is uncertainty about when the distress actually occurred. Despite the addition of time tags to replications and other additional uses of time to avoid potential file management errors, the basic CASP paradigm used to date is not capable of handling the basic simulation requirement of having events occur in simulated time as they would in real time.

5.5 Comparison of CASP (1.x) and GPSS Paradigms

It is instructive to compare the general CASP updating process with the GPSS example above.

In CASP (1.x), there are as many "simulation clocks" as there are replications. However, the use of time is related almost exclusively to drift updates. There is no single clock that is keeping time for all activities in the simulation as a whole. In GPSS, transactions with events scheduled for some future time are kept separate from those with events being processed "as of" the current time. In particular, new transactions with future arrival times are not considered as part of the simulation until their respective arrival times are reached. In CASP (1.x), replications with times in the future are mixed in with replications that have been or are being processed "as of" a given time. Furthermore, "new" replications that have not yet "arrived" (i.e., started moving) are not separated from replications that represent search objects already adrift. It is possible to update a replication file for drift and still have replications present with later drift start times. A probability map constructed from such a file would be in error since it would show replications that did not yet exist as drifting search objects at the simulated time.

On the other hand, in all versions of CASP up to the present, each replication and its event history has been independent of every other replication and event history in the sense that what happens to one replication does not depend on or affect what happens to another. That is, replications do not interact with one another. In the GPSS example of Joe's barbershop, transactions did interact with one another. Only one customer at a time could be served so any other customers present in the barbershop while someone was being served had to wait for service. This situation does not yet arise in CASP. Consequently, CASP could continue to process each replication independently of the others. However, this does not remove the requirement to process the events that occur to each replication in the proper order, and at the proper times, with respect to the simulation clock.

The strict independence among replications and their event histories may not be preserved in future search planning tools. There is a need for replications that represent one type of object to "spawn" replications that represent other types of objects. This is roughly analogous to what GPSS does with a "Split" block to produce one or more "offspring" from a single "parent" transaction. For example, when a craft experiences a distress, several outcomes are possible. The craft might be disabled and adrift, the crew might have abandoned the original craft in life rafts, lifeboats or both, and there might be persons in the water (PIWs). The original craft in its pre-distress state would be the "parent" whereas all the post-distress search objects would be "offspring." However, there are ways to emulate the "spawning" process and still preserve independence among replications.

Another area where strict independence may not be preserved is when it is known or believed that a single distress incident produced multiple search objects. CASP has always operated on the presumption that there was only one search object to be found, even if the search planner did not know whether that object was a boat, a raft or a PIW. If a large ship founders or a large aircraft successfully ditches, there may be multiple search objects adrift. Whenever one of the objects is found, this will impact the probability density distribution for the remaining objects. In turn, this will affect the search planning for the remaining objects.

GPSS is not suitable for the needs of simulating search problems. However, there are valuable lessons to be gained by studying general GPSS techniques and those of other simulation languages (e.g., SIMSCRIPT, SIMULA, etc.) and drawing useful ideas from them. It is clear that great care must be taken in the development of the new SAR tool to carefully define "events" and sequencing mechanisms that provide sufficient realism within the available computing power while also allowing future enhancements.

6 Support for Scenario Generation/Analysis in Future Tools

6.1 Scenarios

A "scenario" is a "story" describing what may have happened to a missing craft from the time it was last known to be safe up to the time of the distress incident. Several such "scenarios" or "stories" may be possible. One of the most critical tasks of the search planner is to define the possible scenarios and weight them based on the available information and the search planner's judgment. It is very important that the search planner avoid "locking in" on any one scenario to the exclusion of the others. As the case proceeds, information will be acquired that will change the scenario weights, and may eliminate some scenarios altogether. The defining and weighting of scenarios is an inherently human activity requiring careful, thoughtful analysis of all the available information. It is not something that a computer can do, no matter how sophisticated the programming may be. Support for describing scenarios and their relative weights, step 4) of the search planning process described in section 2.3, is one of the most important parts of the search planning tool but one that has never been adequately addressed by CASP. This is where the search planner articulates the "initial conditions" for the search problem. The tool must provide an efficient and structured way for the search planner to describe possible craft behaviors and other data about the craft, survival equipment, persons on board, environmental parameters, and hazards that are pertinent to the problem of estimating where and when the craft became distressed and what kind(s) of search objects may have then been released.

Craft normally exhibit the following general behavior:

> Depart an initial location at a specific time and proceed to a destination with an intended estimated time of arrival (ETA) via some number of intermediate positions (waypoints) or operational areas (opareas) along either rhumb lines or great circles at various speeds of advance. There may be a significant number of intermediate positions and/or opareas. Generally craft that proceed to an oparea spend a significant amount of time (operational time or "optime") moving about in the area before departing.

A distress may occur at any time between departure from the position where the craft was last known to be safe and arrival at the final destination. All of the parameters of the journey may have significant uncertainties about their values. The "initial conditions" leading up to a distress situation may be any subset of this general behavior and will depend on what information the search planner has or is able to obtain about the craft, its intentions and likely alternative behaviors, potential hazards, the environment, and the distress incident itself.

6.1.1 Distress Position and Time

The distress position and time may be approximately known, as in the event of a call for help from a distressed craft giving its location. The time would be taken to be the time of the call and the position would be the position reported by the craft. However, distress position and time information may be provided in several forms and may not be complete.

a. **<u>Latitude, Longitude</u>** – A position may be specified by latitude and longitude. There are several recognized formats for specifying latitudes and longitudes. Often it is assumed that the uncertainty associated with a position specified in this form is distributed in a bivariate normal fashion about the given position. The traditional method for specifying the magnitude of the uncertainty is to give the parameters that define the probable error, or 50% containment ellipse. However, other criteria may be used such as standard deviations or sizes of ellipses having a different containment probability (e.g., the 99% containment contour). Once these parameters are chosen, a Monte Carlo simulation would create a distribution of possible distress positions around the given position accordingly, taking care to use appropriate navigational algorithms to compute each possible initial position's latitude and longitude.

b. **<u>Range, Bearing from Position, Landmark or Other Reference</u>** – A position may be specified as a range and bearing from some other position, such as 100 nautical miles southeast of Cape Fear. In this case the reference position may have some uncertainty, the distance (range) of the distressed craft from the reference position may have some uncertainty, as may the bearing. In the cases of range and bearing uncertainties, there are no "standard" uncertainty distributions. The two most obvious possibilities are uniform distributions between minimum and maximum values or univariate normal distributions about the given values. A decision needs to be reached on which of these to use or whether both distribution types should be offered. In any case, a Monte Carlo simulation would have to compute the possible distress positions by sampling from the uncertainties about the reference position, the range, and the bearing and then using those values in standard navigational algorithms to compute corresponding latitudes and longitudes.

c. **<u>Time</u>** – Times are generally expressed in terms of UTC (a.k.a. Universal Coordinated Time, Greenwich Mean Time, Zulu Time). However, times may be expressed as local times, with an appropriate letter suffix. Time uncertainties, like range and bearing uncertainties, may be either uniformly distributed between minimum and maximum values or normally distributed about the given time. Again, a decision is needed about which option to provide, or whether to provide both.

6.1.2 Last Known Position (LKP) Plus Dead Reckoning (DR)

The information available to the search planner may be the craft's last known or reported position at a point in time, its course and speed at that time, and the subsequent time of the distress. For this information to be useful, the difference between the time of the LKP and that of the distress should be relatively small, especially for craft that tend to change course frequently. For longer times, more information about the vessel's intended track or likely behavior should be obtained through investigative efforts and the "Intended Track or Sail/Flight Plan" option described in section 6.1.4 below should be used.

The last known position and its associated time may be in any of the forms cited in 7.1.1 above. As before, all of the parameters may have associated uncertainties, including the time of the distress. In this instance, the time of the distress must be later than the time of the LKP in all cases. The distribution of possible last known positions is generally assumed to be of the bivariate normal type centered on the given LKP. Course, Speed, and Time uncertainties may be distributed either uniformly or normally. The same decisions as before about which distributions to allow need to be reached. At this point it will be necessary to assume or allow the user to specify some pre-distress motion model to use when computing the distribution of possible distress locations and times. Some candidate pre-distress motion models are described in section 6.2 below.

The distribution of possible distress incident times and corresponding positions needs to be computed by sampling from the distributions about each of the parameters and then using appropriate navigational computations to compute sample pre-distress movements up to the distress time when a transition to a search object adrift is made. The distress position and time must be consistent with the available information and subject to physical constraints on craft movement. For example, craft cannot move backwards through time. Independent samples from LKP and distress time distributions could produce distress times earlier than LKP times unless the software explicitly prevents this unrealistic outcome. Some reasonable cutoffs on minimum and maximum speeds are also needed, especially if normal distributions about mean speeds are used.

6.1.3 Operational Area

A common behavior, especially in the fishing industry, is for a craft to proceed to an area, conduct operations there for some period of time, and then proceed to another location. Such areas may be described by well-formed polygons. Although there is often some uncertainty about the size, shape and location of the operational area (oparea), it is usually sufficient for the search planner to specify one or more exact polygons that are large enough to include the area(s) within which the missing craft was expected to operate. Within these polygons, the possible positions for the start of operations, subsequent waypoints, and departure positions are usually assumed to be uniformly distributed over the polygon's area. If there is reason to believe the distribution is not uniform, then the area may be subdivided and the portions weighted according to their relative probabilities.

In addition to the area boundaries, it is necessary to know either the time of the distress and its uncertainty or the arrival and departure times and their respective uncertainties. If only the arrival and departure times are known, it may be necessary to simulate pre-distress motion. In this case, additional information about the nature of the craft's movements while operating in the area will be needed. This could include the mean speed and time between turns when engaged in operations (e.g., trawling) and their uncertainties, the mean speed and time between turns and their uncertainties when transiting from one point to another within the area, etc. In the case of fixed gear, the speed during operations would be zero but the mean and uncertainty for the amount of time spent servicing each "station" would still be needed. For trawling and other operations conducted while the craft is in motion, an estimate of the mean operating speed and its uncertainty would be needed in addition to the mean time expended at that speed between turns or changes to transit speeds. Pre-distress motion modeling is discussed further in section 6.2.

6.1.4 Intended Track or Sail/Flight Plan

The SAR system may be alerted to a possible distress as a result of a craft being either overdue at its destination or unreported while en route. A craft is classified as unreported when it fails to communicate with an "agent" according to a previously arranged communications schedule. The "agent" may be a family member or friend, a company headquarters, a shipping agent, the air traffic control system, a base of operations, etc. Cases involving overdue or unreported craft are among the most difficult for SAR authorities. Initially, it is not clear whether a distress has occurred. Even the probability of a distress incident having occurred is not always obvious. One function of a search planning support tool should be to help the search planner evaluate the situation on the basis of the available information. This can be done by simulating the craft's intended movements along with some reasonable (in the search planner's judgment) alternatives due to changes in plans after the craft was last known to be safe. The intended movements would constitute a scenario, as would each of the alternatives. Each different scenario could then be "weighted" according to the likelihood that it represented what the craft actually tried to do. As with the scenarios themselves, the weights would be a matter of the search planner's judgment based on information obtained about the craft, the habits of the operator, conditions along the intended route and vicinity, etc.

Tracks may originate from distributions of positions centered on an estimated departure position or distributions of positions contained in opareas, as defined respectively in sections 6.1.1 and 6.1.3 above. Craft may also visit additional positions or opareas en route and tracks may end at either a position or an oparea type of location. In addition, it may be desirable to specify a mid-leg position or cross-track error for each leg of the track and distributions of possible variations between a base track course and speed and actual course and speed. (See sections 6.2.4 and 6.2.5 below for further information.) Voyages and flights do have events and these events must occur in a realistic sequence. Events include departure from the point of origin or LKP, course changes, speed changes, arrivals and departures from intermediate locations (which may be either positions or opareas), encounters with specific hazards, distress incidents (perhaps as a function of exposure time to hazards), etc.

When faced with an overdue or unreported situation, the information the search planner is most likely to have or be able to estimate includes the following:

a. <u>A time and position or oparea of origin.</u> This may be a time and port of departure, a last known position and time, or an operational area with an associated time and time uncertainty.

b. <u>A list of intended waypoints/opareas.</u> For opareas, the defining boundaries and the (remaining) amount of operational time (optime) the overdue/unreported craft intended to spend in the area are probably available or can be estimated.

c. <u>Intended or estimated transit speeds.</u> Ideally, these speeds should be speeds made good over the ground. If speeds through the water are to be used, then the model must include modifications based on currents and possibly even leeway to convert speed through the water to speed made good. The intended speed values, if known, may be used as estimates, or the search planner may revise them to account for weather and other factors. As with the other factors discussed so far, there is a need to decide whether to use uniform (min, max) distributions or normal distributions or to allow both as options. Mean and either minimum and maximum estimated times of arrival (ETA) and departure (ETD) or their standard deviations may be computed and displayed for the search planner's review. In the case of an overdue craft, this analysis may reveal that the ETA at the final destination was optimistic, thus reducing concern about whether the craft has become distressed. The opposite may also be revealed, showing that the operator made a pessimistic estimate of ETA, thus raising concern about the craft's well-being.

d. Estimated times of arrival (ETA) and departure (ETD). Estimated times for departure from the origin and arrival at the destination are often available. However a complete specification of the intended track based on times would require ETAs and ETDs for any waypoints or opareas visited en route. These are less commonly available than speeds on legs but the user should be given the option to use times if they are available. Given an ETD from the origin or LKP, an ETA at the destination, and "optimes" for locations en route, speeds on legs connecting the various locations may be computed. These can then be reviewed by the search planner for reasonableness and adjusted as needed. Given ETDs and ETAs for all locations en route, both "optimes" and speeds on legs may be computed. If both speeds and ETAs/ETDs are available, they can be compared for consistency. If ETAs/ETDs are provided, they must be in the same order as the order in which craft intended to visit the waypoints. (CASP (1.x) does not currently require this, creating an opportunity for undetected errors.)

e. Locations and times for major hazards. Hazards may be either fixed or moving. Examples of fixed hazards include busy shipping lanes, (hazardous to smaller vessels and regarded as fixed since movements of individual large vessels are not closely monitored—but this could change), reefs and shoals, etc. Examples of moving hazards or hazards of limited duration are primarily weather-related and include storms, fronts, high sea states, etc.

6.2 Pre-distress Motion Modeling

CASP has never adequately simulated pre-distress motion as a means for estimating when and where a missing craft might have experienced a distress incident. As we have seen, CASP first chooses a position from a static distribution and then assigns an associated drift start time. Whenever the position and time of the distress incident are not both specified, a reasonable model of the craft's pre-distress motion is needed to estimate when and where the craft may have experienced a distress incident. Pre-distress motion models may be as simple as no motion at all, simulating a distressed vessel anchored or aground or an aircraft's reported forced landing site on land. On the other hand, pre-distress motion may be relatively complex, depending on the craft's potential behavior and consideration of factors such as weather that might affect its motion. Some candidates for pre-distress motion models are discussed below.

6.2.1 Distress/Last Known Position and Time

If the position and time of the distress are known within reasonably close limits, there is no need to model pre-distress motion. The distributions of possible positions and times may be developed from the estimated accuracies of the given distress position and time respectively and sampled in Monte Carlo fashion.

6.2.2 Last Known Position (LKP) Plus Dead Reckoning (DR)

There are situations when the time of a distress is known within close limits (i.e., has a small uncertainty) but the position of the distress is not known. If the craft had reported its position, course, and speed as of some recent time, then it is possible to estimate the distribution of possible distress locations by means of dead reckoning. In this case, a starting position and time would be chosen at random from the distributions of possible LKPs and times. Then a course and speed would be chosen at random from the distributions of possible courses and speeds. This information would establish a sample track (of indeterminate length) from among all possible tracks following the time of the LKP. The sample track may be either a rhumb line or a great circle but it should be the same type as the intended track. A distress time would also be chosen from the distribution of possible distress times. Using these randomly chosen data elements, a distress position would be computed along the random sample track using appropriate navigational algorithms. This would produce a sample track length and hence a sample distress position. Repeating this process a sufficient number of times in Monte Carlo fashion would produce a distribution of possible distress positions and times as shown in Figure 6-1 below.

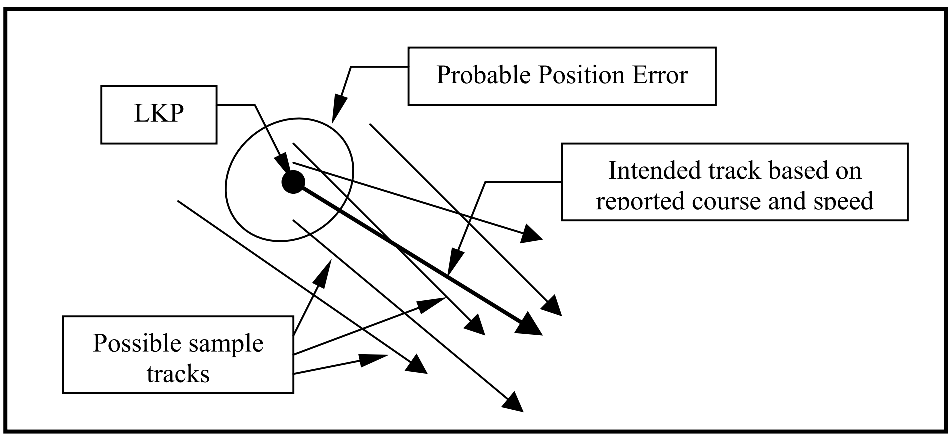


Figure 6-1 – Some Possible Tracks after LKP

6.2.3 Distress/Operational Area

When the distress is known to have occurred while the craft was operating in an area and the time of the distress is known within close limits, then the distribution of arrival, way-, and departure points and times over the oparea may be taken as the distribution of possible distress positions and times without simulating possible pre-distress movements. The distress times would be independently selected at random from the distribution of possible times around the apparent time of the distress. That is, the search problem may be initiated on the basis of the independent distributions of positions in the oparea and distress times alone.

If the time of the distress is not known within close limits but it is known or assumed that the distress occurred in the oparea, it may be necessary to simulate the craft's motion while operating in the area. There are generally two modes of operating while in an area:

- "Sprint and drift," where the craft moves at some transit speed to a position, operates there for some period of time and then moves at some transit speed to another location where a similar operation is performed. A vessel working with fixed fishing gear (e.g., lobster pots, crab traps, etc.) operates in this fashion.

- Continuous motion at various courses and speeds. Trawlers typically operate in this fashion. Trolling for game fish produces somewhat similar motion.

In either case, the motion model would involve a form of constrained random tour that would generate sample tracks like those shown in Figure 6-2.

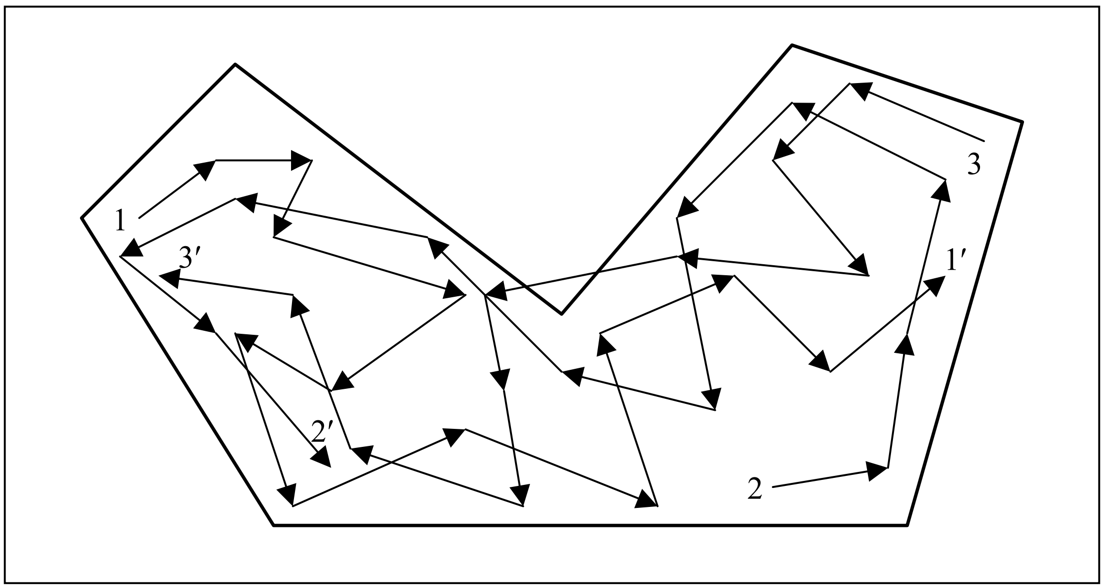


Figure 6-2: Three “Random Tours” of an OpArea

The following constraints are examples of what might be imposed on each leg of the random tour:

- Both of the leg’s end points must lie within the oparea.
- The leg must be characterized as “transit” or “operational” according to the specified ratio of transits to operations (e.g., if every third leg, on average, was a transit, then the ratio would be 1/3).
- The speed on the leg must be drawn from the specified distribution of transit or operating speeds, as appropriate. For vessels that service fixed gear, the operating speed is always zero, which means the lengths of “operational legs” for this type of activity are also zero.
- The time on the leg must be drawn from the specified distribution of transit or operating times on leg as appropriate.
- Each destination point is taken as the origin of the next leg of the random tour.

Algorithms to simulate a craft’s motion under these constraints are potentially quite complex. The following steps form a rough, incomplete, but instructive, example of an algorithm for simulating fishing vessel motion along a single sample tour like one of the three shown in Figure 6-2:

1) Select an origin position and time from their respective distributions.
2) Select a subsequent distress time from the appropriate distribution.

REPEAT

3) Set the origin position and time for the next leg to the position and time just computed.
4) Select another position in the area as an apparent destination.
5) Compute the rhumb line course from the origin to the apparent destination.
6) Select whether the leg is "transit" or "operational."
7) Select a time on leg from the appropriate distribution.
8) Add selected time on leg to the leg's origin time to get leg end time.
9) IF leg end time ≥ distress time, THEN
10) Use the distress time as the leg end time.
11) Select search object type(s) to be generated.
12) Set "distress" flag to generate search object(s) at leg end (and distress) time.
13) ENDIF
14) IF leg end time > datum time, THEN use the datum time as the leg end time.
15) Select a speed on leg from the appropriate distribution.
16) Compute the distance from origin that corresponds to the leg end time and speed on leg.
17) Compute the position along the leg that corresponds to the computed distance.
18) IF "distress flag" set, THEN generate search object(s) of the selected type(s).

UNTIL the distress or datum time is reached.

Note that this algorithm can generate positions outside of the oparea in two cases. First, even if the leg's origin and apparent destination are inside the area, the time and speed on leg that are drawn may carry the simulated craft beyond the apparent destination and possibly some distance outside of the oparea. However, the length of any one leg would be limited by the distributions of times and speeds, so simulated craft would not get too far outside the area. Second, if the area is represented as a concave polygon, as in Figure 6-2, this algorithm will allow "shortcuts" that pass out of the area and back in again. Therefore, some distress positions may occur outside the oparea even though the simulated craft did not "overshoot" the apparent destination. Whether these situations are considered realistic or additional logic is needed to prevent such "external" positions deserves consideration.

Before a random tour algorithm is developed for simulating pre-distress motions associated with opareas, a simplification should be considered. A highly simplified way to deal with opareas for the initial version of the new search planning tool is to make the distribution of "service times" at the initial positions in the oparea the same as the distribution of total "operational times" for craft to be in the area. The implication would be that a craft would go to a single position in the area and remain there until either it was time to depart the area or a distress occurred. Statistically speaking, this might even produce essentially the same distribution of distress locations and times as that obtained from simulating movements that were constrained to remain strictly within the area. However, this hypothesis should be confirmed by appropriate analysis and not just assumed. The technique of making the distribution of "service times" equal to that of the total "operational times" is recommended for the initial version of the new tool, provided sufficient analysis is done to ensure that no absurd results will ensue. If it is later discovered to be either necessary or desirable to simulate motion within opareas, it should be possible to reliably add this feature at reasonable cost without disrupting the rest of the model.

6.2.4 Simulating Complete Voyages/Flights

There are times when the only indicator that a craft has experienced a distress incident is lack of contact. If a craft fails to report when scheduled or fails to arrive at an intended destination when planned, concern about its well-being may be communicated to a Rescue Coordination Center. At some point, continued lack of contact will raise sufficient concern to require that a search be undertaken. In this type of situation, pre-distress motion modeling becomes especially important for two reasons. First, motion modeling can help the search planner assess the gravity of the situation by estimating the chances that the craft could still be en route and not in immediate danger. Second, motion modeling will provide an estimate of the distribution of possible distress times and locations. In both cases, the pre-distress motion modeling needs to be done in conjunction with modeling encounters with hazards and the outcomes of those encounters.

Before attempting to establish the distribution of possible distress times and locations, let us first develop a method for simulating an uneventful voyage (or flight) where no distress incident occurs, such as that depicted in Figure 6-3.

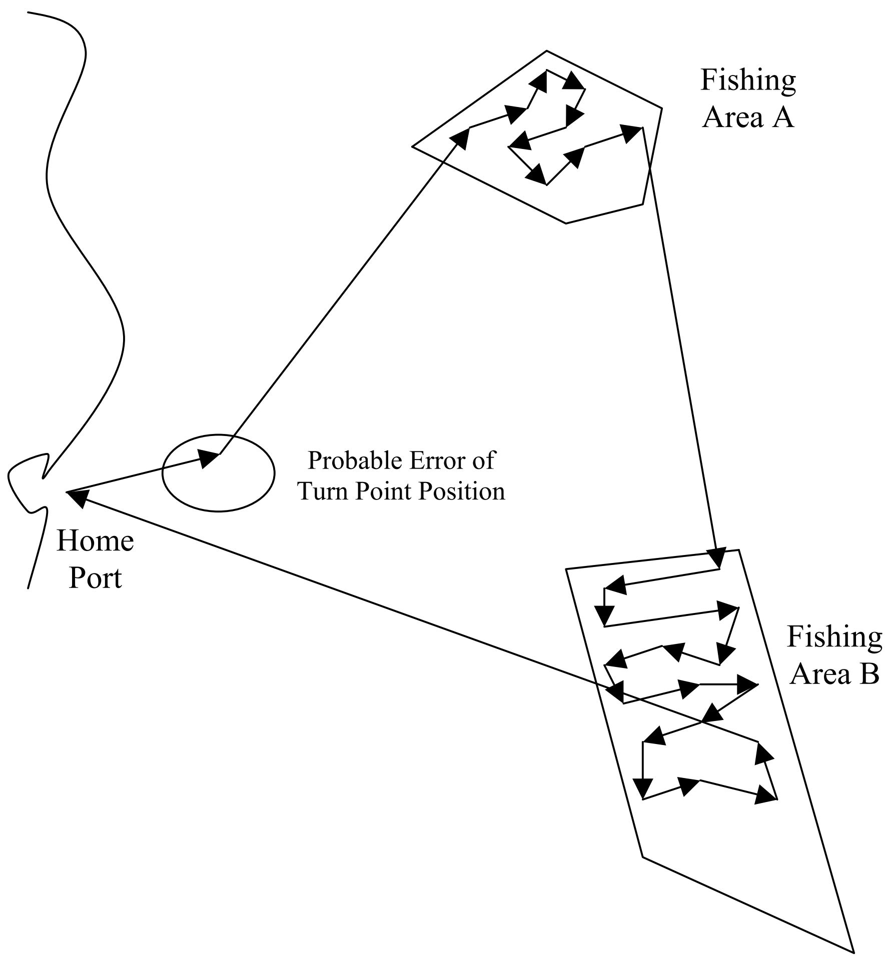


Figure 6-3 – Sample Track
for a Fishing Voyage

As described in 6.1.4 above, a voyage or flight may consist of several elements. Figure 6-3 shows a sample voyage where a fishing vessel departs its home port, proceeds to a fishing area via a turn point, engages in fishing for some period of time within that area, departs the area for a second fishing area some distance away, fishes that area for some period of time, and then returns to home port.

For the sake of simplicity, we will assume that the departure and return points used for the home port correspond to the location of the sea buoy at the entrance to the channel leading into the port and that the departure time from that location is known or can be reasonably estimated. Figure 6-3 shows how one of the infinitely many possible tracks for such a voyage might appear if motion while operating in the opareas were modeled as well as motion to, between, and from the areas. If motion within the opareas were not modeled, then the diagram would simplify to only four legs, with "service times" at single positions inside each area to simulate operations in that area.

The information required to develop sample tracks may not be as extensive or difficult to enter as one might imagine if the user is given the right tools. Use of appropriate graphics tools could make it quite easy to quickly enter a substantial amount of information. For example, the user could first establish a set of locations the missing craft intended to visit using "drawing" tools to create an overlay of "points" and areas on a nautical chart image. These would be the origin and destination locations and any intermediate locations. The user would then select these objects in the order visited, allowing the computer to connect them with nominal track legs. A complete set of geographic objects needed to define the where the craft might have gone in a particular scenario would thus be defined. The user would then select these objects and provide additional information about them needed to model the craft's motion.

For example, a departure position would have attributes of position uncertainty, departure time and time uncertainty. The nominal leg emanating from that position would have attributes of speed on leg and speed uncertainty, a track type (e.g., rhumb line or great circle), and possibly a motion type (e.g., sailing vessel vs. power vessel). That is, nominal track legs would really act as "containers" for descriptive information about how the craft would move from one location to the next rather than representing a specific line along which positions would be computed. An operational area might have attributes of amount of time spent in the area as well as type(s) of motion while operating in the area. And so forth. This information would then allow the software to compute sample paths along which movements of the craft could be realistically simulated and would allow a probability map of the craft's location at any time during the voyage to be computed and displayed.

6.2.5 Simulating Distress Incidents During a Voyage/Flight

Once it is possible to realistically simulate the possible/probable movements of a missing craft, the next issue is to simulate where and when a distress incident may have occurred. The time and place of a simulated distress forms the terminus for pre-distress motion modeling, hence its inclusion in this section.

Craft become distressed as a result of an encounter with some sort of hazard. The "hazard" may be internal, as in a mechanical failure aboard the craft, or it may be external as in an encounter with a storm, or it may be a combination of the two. "Hazards" may be quite general, as the general hazards of just being underway or in flight, or they may be quite specific as in storms, fronts, shipping lanes, etc. Since some hazards move, especially those associated with weather phenomena, just choosing locations and associating distress probabilities with them is not sufficient. Whether hazards are fixed or in motion, just choosing locations does not address when the distress might have occurred. Thus the simulation of distress incidents to terminate pre-distress motion modeling is a complex issue and must be approached with great care.

For example, if one were to make the ***prediction*** that the chances for a vessel to become distressed in a particular voyage scenario were 10%, then simulating the voyage on the basis of the prediction would result in 90% of the simulated craft completing it safely, leaving only 10% in distress. However, given the ***observation*** that the missing craft has not yet arrived at the destination even though the simulation shows that it should have already arrived absent a distress incident, the probability that the craft experienced a distress incident is greatly increased and may be reasonably presumed to be 100%.

This example illustrates an important difference between the simulation techniques needed for the new search planning tool and those used for many other simulations. In many simulations, the general paradigm is to build a model and then see how it behaves as various parameters are modified. Such models are generally used for their predictive value. In SAR, there is usually a mixture of predictive assumptions and observed facts. The model's results as of a given real-world clock time must be consistent with the known facts as of that time. Hence there is the need to simulate possible events that could have already occurred subject to the constraint that they must agree with the known facts of the case. The model also must have predictive value so that efficient, effective searches may be planned.

Consider the following. A vessel's intended track is a rhumb line between points **A** and **B**. A storm crosses the vessel's intended track between the vessel's ETD from point **A** and ETA at point **B**. The ETA at point **B** has passed and non-arrival as of that time has been confirmed, hence the vessel is classified as "overdue." The search planner needs to consider a number of factors, including but not limited to:

- The probability that the vessel encountered the storm.
- The probability that such an encounter would have caused a distress incident.
- The probability of a distress incident without encountering the storm.
- The overall probability that the vessel has experienced a distress incident.

Estimating some of these probabilities by manual means is extremely difficult. However, given sufficient information about the vessel's departure time, intended track, speeds on legs, the storm's movements, size, and intensity, etc., a simulation can provide a good estimate of these probabilities, including estimates of when and where distress incidents were probable. Given information about possible survival craft and a drift update capability, such a simulation can also provide probability maps on search object locations. Given an optimal effort allocation algorithm and information about search craft and sensor capabilities, the simulation can provide optimal or near-optimal search plans for finding such objects. Given information about search facility movements, the simulation can provide post-search estimates of search effectiveness by simulating and evaluating search facility/search object encounters (detection opportunities). While all of the above must be accommodated to have a complete search planning tool, we will concentrate for the moment on just simulating a sample of the possible times and positions of distress incidents.

Let us look at how a simulation of a vessel's voyage under the above circumstances might work. One way to approach the simulation would be to choose a sample path from the departure location to the destination along with a departure time and initial speed under "normal" conditions. Similar information would be selected for the storm, which we will assume can be represented by a circle containing the region of dangerously high wind speeds. A CPA could then be computed between the storm center and the simulated craft. If the CPA distance was less than the radius of the storm, an encounter took place. In this case, the initial time and position where the simulated craft entered the storm would be computed, a "speed while underway in a storm" would be selected for the vessel, and the time and place where the craft departed the storm would also be computed. Finally, the last leg from the storm to the destination would be computed and would yield an arrival time. This would complete the data set needed to describe a successful voyage.

Next, the question of whether the simulated voyage could represent the known facts would be addressed. If the computed arrival time was earlier than the latest time at which non-arrival had been confirmed, then the simulated voyage's outcome could not represent the actual voyage. It would be reasonable to presume that a distress had occurred somewhere along the sample track. In this case, it would be necessary to estimate when and where the distress occurred. This could be done as follows.

Suppose that in the search planner's judgment there was one chance in ten (10%) that the vessel would experience a distress incident under normal conditions and two chances in ten (20%) that it would experience a distress under storm conditions. This would make a distress twice as likely to occur during an encounter with the storm as it was otherwise. Based on this assessment and the presumption that a distress had occurred, choose a uniform random variate between 0.0 and 3.0. If the randomly chosen value falls between zero and one, the distress occurred under "normal" conditions. Otherwise, it occurred under storm conditions. Suppose that the simulated voyage along this particular sample track would have required 100 hours for successful completion, and that 80 of those hours were spent under normal conditions while the remaining 20 hours were spent under storm conditions. Further suppose that the 80 hours were expended in two parts, 50 hours prior to encountering the storm and 30 hours after leaving the storm. If the distress took place under normal conditions, a uniform random variate between 0 and 80 would be chosen. If the chosen value were between 0 and 50, then the time of the distress would be the departure time from the origin plus the chosen value. Otherwise, it would be the departure time from the storm plus the chosen value minus 50. If the distress occurred in the storm, then a uniform random variate between 0 and 20 would be chosen and added to the time at which the vessel first encountered the storm to obtain the distress time. A similar technique would be used in cases where there was no encounter with the storm and the computed arrival time was earlier than the time of the latest non-arrival confirmation. Once the time of the distress was determined for this sample track, the position would be computed using dead reckoning (DR) techniques and standard navigational algorithms.

In the event that the computed arrival time for a sample track was later than the time of the latest non-arrival confirmation, the simulated craft could be shown as "underway" at its DR position for the requested datum time or as "arrived safely" if the computed arrival time was earlier than the datum time. On the other hand, simulated craft in the second situation could be shown as distressed (using the logic described above) if the search planner were to assume that any computed arrival times earlier than the current real-world clock time implied a distress had occurred, regardless of whether non-arrival as of that time had been confirmed. In this case, only craft with arrival times later than the current real-world time would be shown as "underway." This choice might be considered a more cautious assessment of the available data than the one based on the time of the latest non-arrival confirmation. The most cautious option would be to assume that all simulated craft with arrival times earlier than the requested datum time (which is usually in the future with respect to the current real-world clock time) either already had or would become distressed prior to the datum time. Only craft with arrival times later than the requested datum time would be shown as "underway." Although not strictly realistic as of the real-world time of the simulation run, this is a reasonable alternative for search planning purposes. If the vessel arrives prior to the requested datum time, its arrival will no doubt be reported and the search will be called off. On the other hand, if it does not arrive, then the "premature" presumption of distress for all sample tracks with arrival times earlier than the datum time was correct and the search needs to proceed on that basis.

By gathering various statistics and counts, the overall probability of a distress could be estimated, along with the probabilities of encountering the storm, still being underway and not in immediate danger, etc. Such statistics might prove both useful in evaluating the urgency of the situation and in evaluating different scenarios.

The technique just described has several advantages over earlier methods used in CASP (1.x).

- It provides a more realistic model of craft behavior.
- It provides a means for estimating the probabilities of encounters with hazards.
- It provides a means for estimating the overall probability that a distress has occurred.
- It helps prevent users from jumping to invalid conclusions, such as: A storm crossed the intended track, therefore it must have caused the distress. The simulation might show an encounter with the storm was actually unlikely, making a distress in the storm unlikely.

7 Describing Pre-Distress Behavior and Hazards

The usefulness of simulations lies in their ability to represent the behavior of complex systems. Simulations then allow users to explore how a system behaves under a variety of circumstances. In some cases, the user may choose to vary one or more parameters and observe the results. In other cases, the user will want to know how random unpredictable variations in a number of parameters whose values are uncertain affects the system's behavior and the range of possible outcomes over some period of time. Although variations in the various factors affecting a simulation may be random, it is usually possible to make some reasonable assumptions about the nature of the uncertainty in each factor's value. This knowledge normally takes the form of probability density distributions of "errors" or deviations in the factor's value.

The entire search problem revolves around the question of uncertainty—specifically, uncertainty about the survivors' location(s). The uncertainty about the survivors' location(s) is a function of the uncertainty about all those factors that are used to assess where the survivors are at any given time. These factors include:

- Time and place of the distress incident
- Type of search object (affects object's drift motion and "detectability")
- Environmental factors (affects object's drift motion and "detectability")
- Sensor/Platform factors (affects detection performance used for evaluating prior searches)

It is crucial that the various uncertainties and the statistics used to describe them be appropriate to the nature of the phenomena to which they apply. It is equally crucial that whenever factors are combined to simulate an effect (e.g., drift motion) that samples be drawn from their respective uncertainty distributions in a statistically correct fashion that also produces realistic results. These two requirements, statistical correctness and consistency with the real world, require considerable expertise in both statistics and in the field of science (e.g., oceanography) that deals with the phenomena being sampled or modeled. Failure to obtain the necessary expertise and exercise the necessary care in dealing with computations involving the various uncertainties of the search problem will lead to a dysfunctional model that tends to lead search planners astray.

7.1 Types and Sources of Uncertainties ("Errors")

It is important to have a basic understanding of the types and sources of uncertainties. It is also important to understand exactly what the data and associated uncertainties represent as well as where and when they apply.

The terms "uncertainty" and "error" are often used interchangeably because they have the same basic meaning—variation from a mean, predicted, or standard value. In this context, an "error" is not the same thing as a "mistake." An error is the difference between an actual value and an expected or mean value. A mistake is a blunder, such as misreading a table or chart.

The two basic types of error are systematic and random. Systematic errors produce predictable variations and are therefore subject to corrective action once they are discovered. Random errors are not predictable nor are they subject to correction. However, it is possible to design systems, including search planning support systems, to properly account for random errors and provide good decision-making advice in spite of them.

Random errors typically have two sources—measurement error and the natural random variability of the phenomenon being measured. Their relative importance depends on a number of factors and must be evaluated on a case-by-case basis by an expert. For purposes of illustration, we provide two simple examples below—one where measurement error is probably the dominant factor and one where natural random variability is probably the dominant factor.

Consider a representation of a near-shore water current field generated hourly on a one nautical mile grid by direct observation using Coastal Ocean Dynamics Applications Radar (CODAR). The short time interval between updates means that natural variations will be quickly detected and measured. The relatively fine spatial grid greatly reduces the opportunity for large variations between adjacent interior data points in the grid. Therefore the primary source of error for this data set is probably the measurement error of the CODAR system.

Now consider a climatological database of water current information on a one-degree grid at one-month intervals. Entries in such databases are formed by computing statistics for sets of observations made over many years inside each cell of the one-degree by one-degree by one-month grid. The mean current vector is computed, along with the variance of the u- and v-components and the covariance between them. Although most observations come from ship set-and-drift reports and are therefore potentially subject to significant measurement error, the primary source of error when using such data in a simulation is the natural variability of the currents in most parts of the world over such large areas (thousands of square nautical miles) and time spans (28-31 days). In other words, when only long-term macro-scale averages are available, there is a large uncertainty about the current flow at a specific location and time. This large uncertainty reflects the fact that natural variations over shorter distances and time scales are not predictable from the available macro-scale statistics and often produce values quite different from the macro-scale averages.

How sample current values are drawn, interpreted and used may differ between two such significantly different data sets as CODAR and climatology, as we shall explore in section 8.

7.2 Probability Density Distributions

The search problem solution depends on a number of factors and each of these has some uncertainty, or error, associated with it. The associated error is generally described by two things—the characteristic size of the error and the manner in which it is distributed. Statistics such as standard deviation or probable error are often used to characterize the size of normal distributions while the maximum (±) range from the mean or bounding (minimum, maximum) values are used to characterize the size or extent of uniform distributions.

Many factors of the maritime search problem are two-dimensional. Positions, velocity vectors, and distance vectors are all two-dimensional data that have two-dimensional (bivariate) error distributions associated with them. The two most commonly used bivariate distributions are the bivariate uniform distribution and the bivariate normal distribution.

The bivariate uniform distribution assumes components in each of the two coordinate directions of the Cartesian plane are distributed uniformly and independently. The maximum range or the bounds in each of these two directions may be, and usually are, different. Bivariate uniform distributions are often used to distribute points randomly and uniformly over a plane area. With some additional care and computation, they may be used to distribute points randomly and uniformly over a portion of a sphere's surface. Bivariate uniform distributions may also be used to describe vector distributions having a range of magnitudes uniformly distributed between a minimum and maximum value and a direction uniformly distributed between limiting left and right directions.

The bivariate normal distribution assumes components in each of the coordinate directions are distributed according to (univariate) normal distributions. The variances (squares of the standard deviations) in each of these two directions may be, and usually are, different. If the two distributions with respect to the coordinate axes are independent of one another, then a scatter plot of points corresponding to the bivariate normal distribution will have an elliptical shape with axes parallel to the coordinate axes. In fact, contours of equal probability density are ellipses. If the two distributions with respect to the coordinate axes are not independent, then the covariance and correlation are not zero. A scatter plot of the corresponding bivariate distribution will still be elliptical, but the axes of the ellipses of equal density will not be parallel to the coordinate axes.

A special case is the circular bivariate normal distribution where the standard deviations in each of the coordinate directions are equal and the two univariate distributions with respect to these directions are independent (covariance = correlation = 0). A scatter plot of a circular bivariate normal distribution will be, as the name implies, circular. However, rotating a univariate normal distribution through 360 degrees about its mean value does ***not*** produce a circular bivariate normal distribution. Figure 7-1 shows the substantial differences in the probabilities contained within various radii for the two types of distributions.

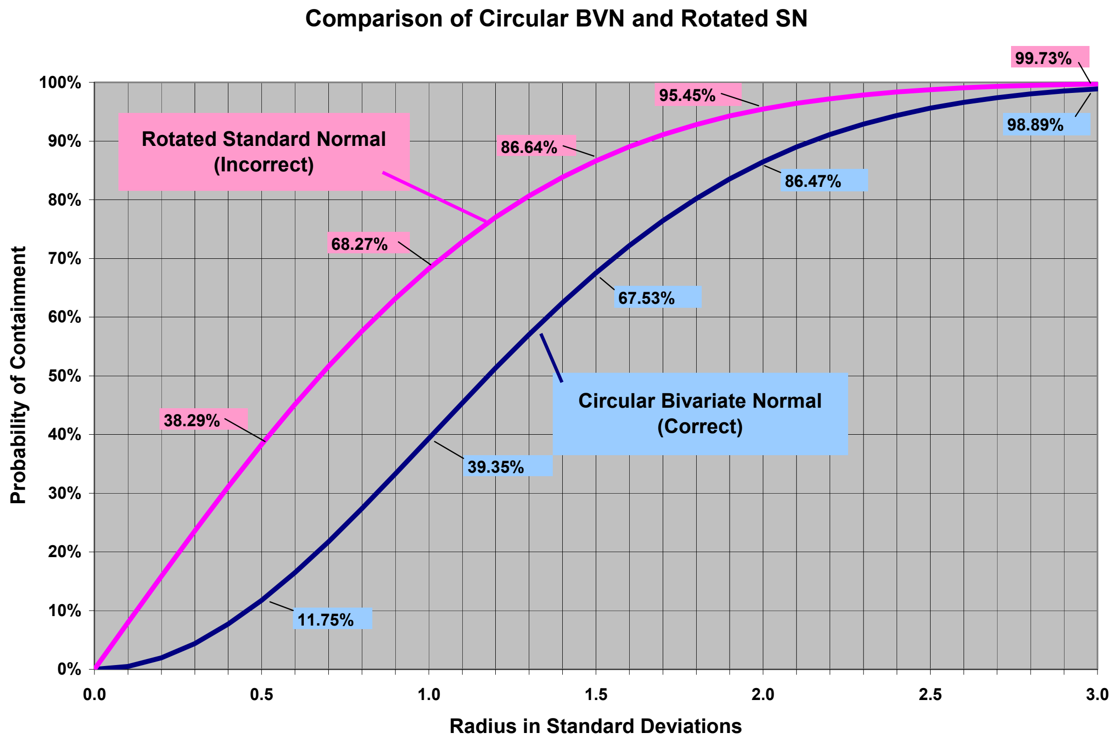


Figure 7-1: Comparing POC Values for Circular Bivariate Normal and Rotated Standard Normal Distributions

7.3 Pre-distress Locations and Motion

Section 6 above discusses this topic in some detail in the context of simulating how craft operate. We will revisit the uncertainty issues here with more attention to how they may be represented in terms of user inputs.

7.3.1 Uncertainty in General

Uncertainty describes the search planner's level of confidence in the accuracy of the information. If a craft reports its position at the time of the distress incident and provides the means by which the position was determined, then the probable error associated with that means of navigation applies, assuming the position was reported to the precision allowed by the means of navigation. However, if the vessel's usual method of fixing its position is known but the location of the distress incident is not obtained with the same precision, the accuracy of the craft's usual method of navigation may not be a good indicator of positional uncertainty. The same is true of intended waypoints along an intended track. Only if it is virtually certain that the operator would follow the intended route as accurately as the method of navigation would allow is the uncertainty in that method a good indicator of the uncertainty about the vessel's actual track versus the intended track. Otherwise, the search planner will have to base uncertainty values on other criteria and use his/her best judgment based on the information available at the time.

It is crucial that search planners understand the difference between a craft's ability to navigate and the quality of the available information they have about the craft's location, course and speed at any given time.

7.3.2 Distress/Last Known Position and Time

Positions on the earth's surface are specified in two dimensions, usually latitude and longitude. Hence positions may have some error in both latitude and longitude, giving rise to a bivariate distribution centered on the specified position. For many types of navigation, the distribution of possible actual positions around a reported or estimated position is of the bivariate normal type. A bivariate normal distribution is specified by the mean and standard deviation (or probable error) of each variable (latitude and longitude). To be more precise, the standard deviation would not be specified in terms of latitude and longitude but in terms of distances north/south and east/west in nautical miles. For small position errors that are not near the poles, choosing "easting" and "northing" sample distances from the respective distributions and computing the location of the corresponding sample point by either plane sailing or rhumb line navigation is sufficiently accurate. For larger errors, the easting and northing sample distances should be used to define the direction of a line of bearing and a distance, but great circle navigation should be used to compute the position of the corresponding sample point. The great circle technique is the most mathematically correct way to transform a distribution in the Cartesian plane to one on the surface of a sphere.

If the errors in the two coordinate directions are independent, then drawing independent samples from each of the error distributions and using them together to define a sample position is appropriate. However, if the two error distributions are not independent of one another but are correlated, then additional steps must be taken to account for the degree of dependency of one on the other. This degree of dependency is usually characterized by the *covariance*. When the covariance is "normalized" to fall in the interval [-1,1], another value called *correlation* or *correlation coefficient* is produced.

From a software perspective, a complete description of a probability density distribution about an object's distress position and time would consist of the following elements:

- The mean, or datum, position coordinates expressed as latitude and longitude.
- The variances associated with the mean position in each of the coordinate directions in nautical miles and the covariance between them.
- The distribution type (e.g., bivariate normal).
- The mean, or datum, time of the mean position.
- The variance about the mean datum time.
- The distribution type (e.g., uniform or normal) about the mean datum time.

For users, who are not generally accustomed to working with variances or standard deviations (standard deviation is defined as the square root of the variance), the uncertainties should be expressed in some easily understood fashion that may be converted to variances automatically by the software. The method used for expressing positional uncertainty in the manual method is probable error, or the dimensions of the 50% probability of containment contour centered on the mean position. However, many users in the past were unaware of this definition and often entered their estimate of the 100% containment contour about the mean position. In theory, a normal distribution has no maximum limit and so, in theory, the 100% containment contour would have to be infinitely large.

A solution that would allow users to think in terms approximating 100% containment would be to assume the user's values represent some probability of containment slightly less than 100%. For example, if it were assumed that the distribution of positions about the mean position was of the circular bivariate normal type and it were further assumed that the user's estimate of position error was the radius of a circle centered on the datum position and containing 99% of the possible positions, this would mean the value entered by the user represented about three standard deviations. Therefore, if the user entered a "containment radius" of 30 nautical miles, the standard deviations in each of the coordinate directions would be about 10 nautical miles under these assumptions. The radius of 30 nautical miles could be made a true containment radius, without a significant impact on the validity of the simulation, by rejecting any sample positions the software might initially generate outside the specified radius.

A decision is needed on what the user's estimates of position and time errors are supposed to represent. The two obvious choices are probable error (50%) and essentially complete containment. The manual method has always relied on probable error and, presumably, the position error values provided in the SAR manuals for various modes of navigation are also probable errors. On the other hand, probable error seems never to have been well understood by field personnel in its statistical context and it was not applied to anything other than position data in the manual method. If asked to estimate a position's error and then asked how likely the object was to be within that radius of the mean position, many users would probably respond with values at or near 100% simply because it seems easier and more intuitive to estimate the maximum error. The same is true for other quantities. It seems easier and more intuitive to estimate minimum and maximum values or the maximum amount an item could vary either side of the mean value than to estimate another statistic like probable error, standard deviation, or variance. Therefore, it is recommended that the new search planning tool standardize on the use of essentially 100% containment for all uncertainty estimates, including position error estimates. For normal distributions, the users' error estimates would be assumed to represent three standard deviations. This would include 99.7% of the probability for a univariate normal distribution and 98.9% of the distribution for a circular bivariate normal distribution, such as position error. It may be appropriate to display a note to users when requesting position data reminding them of the difference between probable error and essentially 100% containment.

As the above discussion implies, users rarely have sufficient information about the nature of the errors affecting position data to make separate estimates of the north/south error and east/west errors and any correlation between them. Therefore, position errors provided by users are generally assumed to be of the circular bivariate normal type characterized by a single value—either the probable error or the containment radius. If a distribution type for times was assumed (e.g., either uniform or normal) then the data elements required to define a distribution of positions about a mean position could be reduced to a more "user friendly" set:

- The mean, or datum, position coordinates expressed as latitude and longitude.
- The “maximum” error radius (in nautical miles) associated with the mean position.
- The earliest and latest possible times for the distribution of positions.

Although this would be sufficient for user-entered data sets, it is recommended that storage space be kept available for the more complete data set given previously. This will allow data from other sources, such as COSPAS/SARSAT, to be properly represented when such systems can automatically provide more complete descriptions of positional uncertainty. The software could automatically expand the more limited user version to fill the larger data set so that subsequent modules would always be dealing with the same data format.

Sampling from the positional and temporal distributions would be accomplished by standard methods for sampling from bivariate normal, univariate normal, and uniform distributions. Appendix C contains several pertinent references. For a single estimated mean distress position and time, it should be safe to assume that the positional and temporal distributions are independent of one another so there would be no requirement to correlate sample times with sample positions. That is, sample positions and sample times may be drawn separately and independently. It should be noted that zero error or uncertainty is a valid value indicating certainty on the part of the user. Therefore, the software must properly handle uncertainty values of zero in all cases.

As mentioned earlier, CASP (1.x) does not currently use the correct type of probability density distribution for positional errors. The developer of the current version apparently misinterpreted the term “circular normal distribution” to mean simply rotating a standard univariate normal “bell curve” about the mean value to distribute points in the plane. This produces a substantially incorrect distribution that has no known meaningful physical interpretation in the real world. CASP (1.x) also calls the entered position error a “containment radius” but assumes the position error entered by the user corresponds to 1.5 standard deviations, which in turn contains only about 87% of the incorrect distribution and would contain only about 68% of a true circular bivariate normal distribution. These attributes mean that the module GEN_POSN is not suitable for inclusion in the new tool.

7.3.3 Distress/Operational Area

There are times when it is necessary to assume that a distress occurred while a craft was operating in some defined area. There are also times when the only information available about a craft’s “last known position” is that it was operating in some area even if the distress is believed to have occurred after the craft departed the area for some destination. Such areas may be assumed to be polygons defined by the positions of their vertices, which are provided by the user in latitude and longitude. Under a scenario where the distress or LKP occurred in such an area, the polygon defined by these points is assumed to contain all of the associated probability. Usually there is no additional information to indicate a higher probability of distress or LKP in one portion of the area versus another. In this case, the distribution is assumed to be of the bivariate uniform type. Such a distribution is one where the possible distress positions are spread over the area with a uniform density—i.e., the same number of points per unit area everywhere within the polygon’s boundaries.

The data elements required for a user to define a distribution of distress positions and times in an area, such as a fishing or other operational area that can be represented as a polygon, are:

- A list of ordered (latitude, longitude) pairs giving the positions, in either clockwise or counterclockwise order, of the vertices of the desired polygon.
- The earliest and latest times associated with positions in the polygon.

In addition to allowing the user to enter vertices as (latitude, longitude) pairs, aids should be provided for computing and displaying vertices that are implied by other data. For example, if an approximate range and bearing are given from an approximate reference position, the user should be able to enter the approximate reference position and its error (if any) along with the minimum and maximum bearings and ranges from that location. The software should then provide a geographic display of this data showing all of the area the positional data and its uncertainties imply. This aid should also be able to handle multiple sets of such information in the event that approximate ranges and bearings are available from multiple reference positions. Given this information in an appropriate geographic display format, the user could then define an appropriate polygon by using the cursor to select positions to be used for vertices. In this case, the craft's motion while operating in the area would not be simulated.

For areas and uncertainties about the distress or departure time that are not too large, it should be safe to assume there is no correlation between the position within the area and the time of that position. For larger areas and time spans, and when it is known or assumed that the craft intended to "work" the area in a particular direction or be in different parts of the area during specific time periods, the larger area could be subdivided into several smaller areas, each with its own earliest and latest times that may overlap with other those of other sub-areas.

If a hazard, such as a front or storm moves through the area and a distress is deemed to be more likely based on an encounter with the hazard, then the distress times will not be distributed in the same way they would be if no specific hazard were present. In the case of a hazard, the search planner would have to indicate how likely it would be to cause a distress incident as well as describe its size, shape, and movements.

For example, suppose the search planner believed that a distress in a storm passing through the area was twice as likely as to "cause" a distress as normal conditions were, then when a sample position in the area was computed, a value would be drawn from a uniform distribution over the interval [0,3>. If the random value fell between zero and one, then the replication would be marked for becoming distressed sometime when conditions were normal. Otherwise, it would be marked for becoming distressed sometime when the storm was passing over the replication's position. The exact time for the sample distress would then be selected accordingly after the periods of "normal" and "storm" conditions were computed for that position.

Note that the individual positions used to describe the vertices of the polygon do not have position errors or time information associated with them. This is because the polygon these vertices represent defines the extent of the distribution. Associating position errors with the individual vertices does not make sense because the vertices do not represent positions the missing craft intended to visit nor are they subject to errors in the normal sense. It is presumed that the search planner placed the vertices exactly where he/she wanted them. For the similar reasons it does not make sense to associate different times with each of the vertices. The set of vertices should be treated as a single entity, namely the polygon they define and within which the distress or "LKP" is known or believed to have occurred. Earliest and latest times may then be associated with this entity, just as in the case of a distress or a last known position.

CASP (1.x) currently creates a uniform distribution of replications over a polygon's area by first enclosing the polygon in a quadrilateral whose sides are parallels and meridians, as shown in Figure 7-2 below. It then chooses a latitude value at random from a uniform distribution between the bounding parallels and independently chooses a random longitude from a uniform distribution between the bounding meridians. The position thus generated is then checked to see whether it falls inside the polygon. The algorithm that performs this check is embodied in the CASP (1.x) subroutines PTINI$CG and PTIN$CG. This algorithm is suitable for use in the new search planning tool for relatively small areas not too close to either pole.

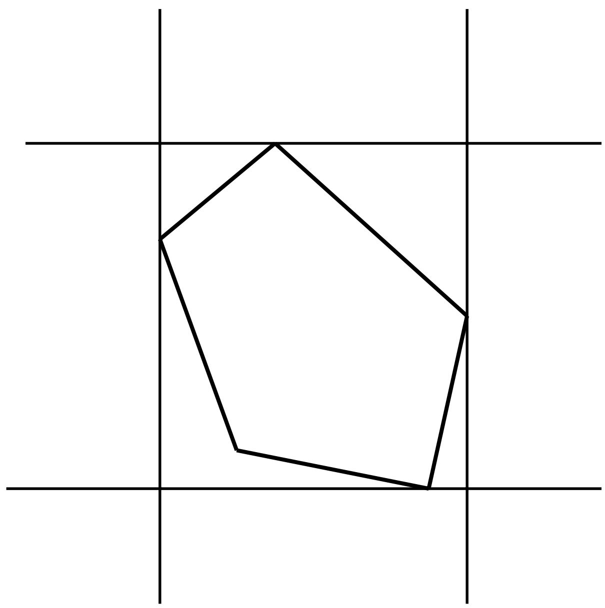


Figure 7-2 – A Well-formed Polygon and
Its Bounding Parallels and Meridians
(Mercator Projection)

For small areas, the technique of selecting longitudes from a uniform distribution between the bounding meridians is adequate but it does not produce a precisely uniform distribution in the sense of points per unit area. The distance between the meridians at the higher of the two bounding latitudes is smaller than the distance between them at the lower bounding latitude. Hence the point density near the higher latitude will be somewhat higher than the point density near the lower latitude. On a probability map, this will tend to give cells at higher latitudes higher probability densities than they are due. These artificially higher densities may adversely impact the optimal allocation of the available searching effort.

The problem just cited is likely to become important only for areas that span several degrees of latitude or areas that occur in polar regions. Since sea lanes in the Arctic are expected to open during the summers beginning as early as 2015, the need to accommodate SAR search planning much closer to the North Pole than previously considered necessary is a real possibility. An analysis is needed to determine whether resulting search plans will be sufficiently affected for this problem to require solution. If so, an appropriate algorithm for uniformly distributing points over a portion of a sphere's surface will have to be developed.

Another issue that will require attention is ensuring the positions entered as vertices of a polygon define a "well-formed" polygon where none of the "sides" cross. For example, it is possible to enter the pentagram depicted in Figure 7-3 into CASP (1.x) as an "area" without causing any error or warning message.

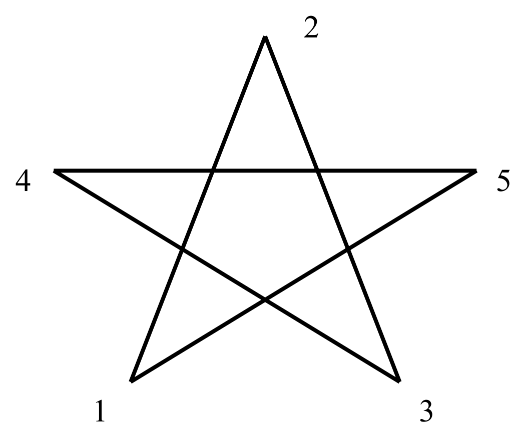


Figure 7-3 – An Ill-formed "Polygon"

The numbers near the vertices in Figure 7-3 indicate the order in which the user entered the positions. The algorithm that checks to see whether a generated sample position falls within this "polygon" would fill the triangular "points" of the star but would leave the central pentagon empty.

Unfortunately, the GEN_AREA module in CASP (1.x) is **not** suitable for use in the new search planning tool for several reasons. Apparently in an attempt to create a "one size fits all" representation for position data, regardless of use or context, CASP (1.x) requires that position errors, times and time uncertainties be associated with every position. When these positions are then used to define the vertices of a polygon, their associated errors are not ignored but are used to enlarge the polygon and round its corners. Since the minimum allowed position error is 5.0 NM, the user cannot defeat this "feature" by entering position errors of zero. If the first "is it in?" test using PTINI$CG and PTIN$CG fails, CASP (1.x) invokes another subroutine, PTINB. This module first checks to see whether the sample position falls within the "containment radius" around any vertex. If it does, it is added to the distribution. If not, PTINB then checks to see whether the distance from the sample position to the nearest point on the line segment connecting any two sequential vertices (in the order they were entered) is less than the "containment radius" computed for that point on the line segment. This "containment radius" is computed by linear interpolation between the "containment radii" at either end. If the position passes this test, it is added to the distribution. Otherwise, it is discarded.

Although the CASP (1.x) developers could have chosen to just ignore position uncertainties when the positions were used to define the vertices of a polygon, they could not ignore the times associated with the positions. No mechanism was provided for the user to enter a time and time uncertainty associated with the area as a whole. Hence CASP (1.x) computes the earliest time associated with each vertex and takes the earliest of these to be the earliest time associated with the area. It finds the latest time in a similar fashion. Then, the sample time associated with a specific sample position is drawn from a uniform distribution between the earliest and latest of all the vertex times.

If it is deemed necessary to simulate the craft's motion while operating in the area in order to obtain a valid distribution of distress locations and times, then positions within the polygon could be selected from a bivariate uniform distribution as before, but they would represent starting positions for the commencement of operations rather than distress positions. It would also be necessary to specify a distribution of starting times and a distribution of amounts of time spent operating in the area. As discussed in section 6.2.3, speed and time on leg profiles for operating behavior would be needed along with a description of the nature of that behavior (e.g., "sprint-and-drift" or "continuous operation"). However, since it was recommended earlier that simulating craft motion inside operational areas be deferred, possibly indefinitely, a detailed list of required input data elements is not provided here.

7.3.4 Last Known Position (LKP) Plus Dead Reckoning (DR)

As described in section 6.2.2, the distress position itself may not be given but there may be reason to believe the distress occurred close to a certain time. This would be the case, for example, if the stricken craft transmitted a distress call but only the craft's identity could be determined from the transmission and the time of the transmission was noted. Given this information, it might be possible for a search planner, through investigative efforts, to locate an agent for the craft or some other interested party who had recent knowledge of the craft and its intentions. If the craft had recently reported its position, course and speed as of a specific time, it would be possible to estimate its position at the time of the distress as described in section 6.2.2. Note that the "LKP" could be an operational area. In either case, the information required by the software to describe the distribution of possible distress locations and times would consist of all the data elements necessary for describing the distribution of possible last known positions and times, plus the following:

- The craft's last known, intended, or assumed mean course.
- Estimated variance about the mean course for movement after LKP.
- Type of distribution about the mean course (e.g., uniform or normal).
- The craft's last known, intended, or assumed mean speed.
- Estimated variance about the mean speed for movement after LKP.
- Type of distribution about the mean speed (e.g., uniform or normal).

- Estimated mean time of the distress.
- Estimated variance about the mean distress time.
- Type of distribution about the mean distress time (e.g., uniform or normal).
- Type of track (e.g., rhumb line or great circle).

As with the distress/last known position data in the previous section, this data set can be substantially reduced and made more palatable to the user by assuming a distribution type for each of the factors and using limiting values to describe uncertainties rather than statistics like variance. For user inputs, the following data would be sufficient when added to the LKP data:

- The left-most and right-most courses from LKP.
- The minimum and maximum speeds from LKP.
- The earliest and latest distress times.
- The type of track (e.g., rhumb line or great circle).

This method for estimating the distribution of possible distress locations should be limited to relatively short intervals between the time of the LKP and the time of the distress. Such intervals should be measured in hours, not days, for vessels and minutes, not hours, for aircraft. Although the type of track over such a short interval would make little difference in most cases, it could start becoming significant for higher speed craft even within the time limitations just described. In any case, the type of track should match the missing craft's intentions. For longer periods it would be necessary to ascertain the craft's intended track from the last known position, enter the track's parameters (see 8.3.5 below) and the minimum and maximum distress times. The ability to model interactions with hazards at and subsequent to LKP should be preserved.

7.3.5 Intended Track

When search planners find themselves dealing with an overdue or unreported craft, they are facing a real mystery. Like most mysteries, the solution requires expending a great deal of investigative effort to obtain evidence and clues, assess their accuracy, relevance and reliability, and piece them together to form one or more coherent scenarios about what may have happened to the missing craft. A simulation can be very useful in helping the search planner assess the various scenarios and develop an optimal or near-optimal search plan based on that assessment.

There are basically three situations where simulation of a voyage or flight are needed:

- The craft is unreported, i.e., failed to communicate as scheduled.
- The craft is overdue at its intended destination.
- The craft transmitted a distress call with its identity but its position was not copied and sufficiently recent position, course and speed data are not available.

In the first situation, an unreported craft, the search planner might choose to do any of the following based on the circumstances surrounding the craft's voyage/flight:

  - Monitor the case until the craft misses another scheduled communication or becomes overdue.

- Simulate possible motions of the craft since LKP and those of hazards in the area to assess the likelihood that the craft experienced a distress incident.
- Estimate where the craft would be if not in distress and search for it to confirm its safety.
- Assume the craft became distressed since its LKP and mount a full-scale search.

In the first case (monitoring) there is nothing for the new search planning tool to do. In the second case, the tool would be run in "predictive" mode with no presumption that a distress had occurred. Instead, the presumption would be that there was a communications equipment failure, the operator simply forgot to communicate as planned, or there was some other non-distress casualty that prevented the craft from communicating. The third case (estimate the craft's location if not in distress) also presumes no distress has occurred. The same "run" of the new search planning tool should be able to address both the second and third cases. Although rarely done, there may be cases where investing a relatively small amount of search effort to locate a vessel and confirm the its safety based on the presumption that it is not in distress is more than offset by the added risk and potential for search area enlargement associated with simply waiting. Finally, the last case is the primary situation for which the new search planning tool is being designed.

An overdue situation is similar to an unreported situation except that "monitoring" would be reduced or eliminated. In "predictive" mode, the simulation would be able to report the probability that the craft should have arrived by the requested "datum time," which could be the craft's original ETA, the current real world clock time, or any other time after departure. Runs for different times could provide information such as the probability that the craft would have arrived by its original ETA if no distress had occurred, the probability that it will arrive by some future time if there has not been a distress, etc. In this way, the search planner can assess whether the ETA is consistent with other information about the craft's intentions and capabilities. If such a simulation shows a high probability of arrival prior to the current real-world time and non-arrival has been confirmed, then the probability that a distress has occurred is also high and it may be time to plan a full-scale search using the search planning tool in "distress" mode.

The last situation is one where the time of the distress is approximately known but the distress position is not known. Given the identity of the craft, it may be possible to ascertain, through investigative efforts, the craft's last known position and intentions. Simulating the craft's intended movements after LKP within the uncertainties indicated by the available information and its quality would provide a distribution of possible distress positions and times when combined with hazard information.

We have already discussed most of the elements involved in simulating a vessel's voyage in section 6.2.4. Figure 6-3 illustrated one possible track of a fishing vessel that departed port, changed course at a turn point, arrived at one fishing area and operated there for a time, departed that area for a second area where it operated for some more time, and then returned to home port. Uncertainties about the location of the actual turn point and the locations of points in the fishing areas that were to be visited were represented by appropriate distributions and statistics. However, there are two alternative methods for computing a craft's movements along each leg of the sample path.

- If the distributions of times of departure from one end of the leg and arrival at the other end are known, then an average speed along the leg may be computed based on the sample times drawn from each distribution. The position of the craft along the sample leg at any intermediate time may be computed using this average speed in the familiar formula relating speed, time, and distance. Alternatively, the time of arrival at any intermediate position along the leg may be computed using the same relationship.
- If the distribution of departure times from one end of the leg is known and the distribution of speeds on the leg is known, then the time of arrival at the other end of the leg may be computed using the sample departure time and sample speed drawn from their respective distributions. As before, either the position of the craft along sample leg at any intermediate time, or the time of arrival at any intermediate position along the leg may be computed.

Consistent use of either method over the entire track is computationally acceptable so the question of which to use revolves around what information is available. In the example of section 6.2.4 it was assumed that the distribution of departure times from LKP and the distributions of speeds on all legs were available. If CASP (1.x) used the sample path technique with the inputs it now accepts it would assume that distributions of times about the endpoints of sample legs were known. It is recommended that both options be provided. It is recommended that if speeds on legs are entered that the ranges of computed times at the turn points and destination be provided as information to aid the user in evaluating the reasonableness of the track description. Similarly, it is recommended that if times at positions are entered that the range of computed speeds on legs be provided for the user's use in performing reasonableness checks.

Another input that will be needed is information about the amount of time spent at each location along the sample path. Normally the amount of time a craft spends at a turn point of a track is zero if it is in transit from one place to another. However, for the sake of regularity and because there may be some situations where a craft proceeds to a position and remains in the vicinity for some period of time, it is recommended that the user have the option of entering the amount of time spent at each position, with the default value being zero. For operational areas, it is clearly necessary for the user to provide the amount of time the craft was expected to spend in the area.

The list below shows the information necessary to specify an intended track and its uncertainty parameters when estimated speeds on legs are used.

- Location parameters for the departure/last known "position" (LKP as defined in section 7.3.2 or operational area as defined in section 7.3.3 above).
- Location parameters for the next turn point of the intended track (position or operational area without associated time data—times will be computed based on the sample speed and length of the sample leg).
- Estimated minimum and maximum speeds to the next turn point.
- Estimated minimum and maximum amounts of time spent at the next turn point.
- Type of path followed to the next turn point (e.g., rhumb line or great circle).
- Additional next "turn point" location and time spent data and next leg speed data until all legs of the intended track are specified and the intended destination is reached.

Ranges of arrival and departure times at each location and the average time spent there should be computed and provided for the user's information.

The list below shows the information necessary to specify an intended track and its uncertainty parameters when ETAs at turn points are used.

- Location parameters for the departure/last known "position" (LKP as defined in section 7.3.2 or operational area as defined in section 7.3.3 above).
- Location parameters for the next "turn point" of the intended track (position or operational area without associated time data—ETAs are provided as described below).
- Estimated earliest and latest arrival times at the next turn point.
- Estimated minimum and maximum amounts of time spent at the next turn point.
- Type of path followed to the next turn point (e.g., rhumb line or great circle).
- Enter next "turn point" location, time spent, and ETA data until all legs of the intended track are specified and the intended destination is reached.

Average and ranges of speeds between each pair of locations should be computed and provided for the user's information.

Once the intended track has been described, the user may want to make a number of different queries. Some examples are:

- If the craft did not experience a distress, where should it be/have been as of time t? (Show as a probability map.)
- If the craft experienced a distress between times t_1 and t_2, what does the probability map of distress positions look like?
- Given that the craft has/had not arrived at its destination as of time t and a distress incident is the presumed cause of non-arrival up to that time, what is the probability that the craft experienced a distress incident prior to time t?

The module GEN_TRK in CASP (1.x) does not use the sample path technique. It also has many other problems too numerous and, in some cases too complex, to list and explain here. Consequently, GEN_TRK is not suitable for use in the new tool.

7.4 Hazards

Any location where a distress may occur can be considered hazardous to at least some degree. Hence we must begin by assuming the whole world is continually hazardous, as indeed it is. However, there may be some regions of space and time that are or would have been more hazardous to a particular missing craft that is the subject of a SAR case than other regions. Hazards may be either fixed (e.g., shipping lanes, reefs, etc.) or moving (e.g., weather fronts, storms, etc.). There are several ways to approach the representation of hazards. Since this is new territory for a SAR application, we will examine some alternatives.

In abstract terms, a geometric shape such as a circle or a polygon may be used to represent the area affected by a hazard. The time period for which conditions in a fixed hazard area are dangerous may be represented by earliest and latest times. If the hazard is moving but maintains its size and shape, then the user may specify its trajectory in much the same way as a craft's intended track by picking some representative "hazard datum" point and describing its movements as a series of positions and associated times or speeds between positions. The software could then move that point along the given trajectory while moving any other points used in the hazard's description so that they "maintain station" on the "hazard datum" point. Finally, the most general case would be to specify the movements of each of the points used in the hazard's description. This would allow changes in the size and shape of the hazard over time as well as allow movement. The technique would be especially useful for describing weather fronts, which tend to "flex" as they move. However, this generality will have to be weighed against the added volume and complexity of the inputs required from the user, although a good geographic interface should ease the user's burden. An alternative solution would be a digital product from FNMOC or NOAA that describes weather fronts and their motion, if such a product already exists or can be provided upon request.

Since most moving hazards are weather-related, it may be possible to infer hazardous areas and times from gridded environmental data. For example, wind speed and sea state are closely coupled and hazardous conditions are often related to wind speed. In fact, wind speed defines gales, storms and hurricanes. If the wind data files have sufficient spatial and temporal resolution, it may not be necessary to ask users or weather data product providers for additional information on weather-related hazards. Unfortunately, the products currently received probably do not have sufficient resolution for defining where and when hazardous weather conditions are in effect.

In addition to describing the hazard itself it is also necessary to describe its effects. In particular, it is necessary to describe the probability that an encounter with it will cause the missing craft in question to suffer a distress incident. In addition to this "primary" transition from not-in-distress to distressed, the probability of each particular outcome of a distress is needed. For example, if the missing vessel encounters a storm, the search planner might judge that there is a 30% chance of a distress. However, when a distress occurs, there may be only a 20% chance that the vessel will remain afloat and adrift while the chances of the crew abandoning ship in a life raft may be 40% with an equal chance that they will become persons in the water. These three transition probabilities (20%, 40%, and 40%) would then be used to either control the number of replications representing each object type that are generated from simulated encounters with the storm or to assign weights to each object type. Ultimately, for each (hazard, craft) pair and each (hazard, search object) pair a set of transition probabilities is needed that describe the chances of the craft or object transitioning into another type of object, possibly as a function of the amount of time the craft or object has been exposed to the hazard. Considering exposure time would also allow survival modeling, such as simulating cold water survival times, as a natural part of modeling hazards and their consequences. The Canadian Cold Exposure Survival Model shows excellent potential for modeling survival times for both low air temperatures and immersion in water.

The need to specify transition probabilities brings up the interesting question of whether to make these probabilities part of the missing craft's description, part of the hazard description, or define a new data entity for relating craft, objects and hazards. It seems logical to include the types of objects a craft or object may become as part of the craft or object description. For example, a vessel that carried both life rafts and lifeboats could transition into one of the following:

- The vessel itself, disabled and adrift,
- One or more life rafts (depending on the number carried),
- One or more lifeboats (depending on the number carried),
- One or more persons in the water.

However, the transition probability should probably be a data entity unto itself that connects or relates the craft and object to the hazard in much the same way that CASP (1.x) relates "locations" and "targets" to create "situations." The user could select the craft and be presented with a list of objects it could become (taken from the craft description entered earlier) and a list of hazards (also entered earlier). The user would then select an object and a hazard and specify the corresponding transition probability that a distress incident "caused" by an encounter between the craft and the selected hazard would result in the selected object going adrift. A similar technique could be used to specify a craft's speed parameters while engaged with a hazard.

8 Post-Distress Motion (Drift) Modeling

Pre-distress motion and hazard modeling produce a sample set from the set of possible distress positions, times, and search objects. The post-distress motion, or drift, then becomes a function of currents, winds, and search object leeway characteristics. Given a starting position, time, and search object type for a specific replication (rep), a new position may be computed for a later time using the following steps:

- Access the environmental database (or a selected subset for the region of space and time that contains the case) based on the starting position and time, and draw sample wind and total water current vectors from their respective uncertainty distributions.
- Based on search object type, draw sample leeway parameters and compute a sample leeway using the sample wind just drawn.
- Using these sample values, compute the sample direction and rate (or u and v components) of drift for this rep.
- Based on the length of the drift interval (e.g., one hour) and the rep's drift velocity vector, compute a new position using appropriate navigational algorithms.

These steps would then be repeated for each rep for each drift interval, where the starting position and time for an interval is identical to the ending position and time from the previous interval. However, there are questions about when and how to draw new samples from the environmental data and leeway parameters. These will be explored presently.

The last two steps in the above list are already well-defined. They involve only simple vector addition and navigational computations familiar to any professional mariner. These steps will employ techniques very similar to the drift computations of the classical manual search planning method.

The first two steps are not so straightforward. Accessing the environmental database using a position and time is straightforward enough, as is accessing a database of leeway parameters using search object type. However, sampling techniques are another matter. Proper handling of them will determine, in large part, the validity of the model for updating distributions on search object location to properly reflect the effects of the means and especially the uncertainties associated with the factors affecting drift.

8.1 Environmental Data

There are at least two alternative ways to interpret gridded environmental data and their associated statistics, and several ways to sample from those statistics. The two alternative interpretations are

- Each data item characterizes a behavior (e.g., wind or current) within a cell of the grid. That is, the data is interpreted as applying over the entire "volume" of the cell in space and time.

- Each data item characterizes a behavior (e.g., wind or current) at a single point in space and time corresponding to where one set of grid lines intersect.

Although there are standard algorithms for sampling from uniform, bivariate uniform, normal, and bivariate normal distributions there are several other issues relevant to sampling environmental data. These additional issues revolve around ensuring each replication's drift trajectory is reasonably realistic and they include such things as

- Sampling frequency.
- Correlation among samples in a series used to compute a drift trajectory.
- Maintaining the "structural integrity" of the phenomenon from which samples are drawn.

How these issues are addressed depends on the characteristics of the data set from which the samples are drawn. For example, if a climatological database of sea currents is used that has a temporal resolution of one month, but the drift update frequency in the model is hourly, then drawing a new independent random sample of the current every hour would have two effects. First, the sea current samples could vary suddenly from one extreme to another in just one hour. Since climatological uncertainties tend to be large, the difference between two extreme values could likewise be very large. Sea currents do not vary by such large amounts over such a short span of time; hence the two samples when taken together would not be a realistic representation of sea current behavior. Second, after a period of several hours, the resulting average sea current for the sample set would always be close to the average value for the population. This means that frequent sampling under these conditions would be a waste of computing resources since the mean value is already known.

At one time, CASP (1.x) actually demonstrated these characteristics. The latter principle is also known as the law of large numbers or more informally as the law of averages. Given a distribution of minimum geographic size about a point, a single mean sea current vector with an associated uncertainty distribution, and a drift update period of one hour, the updated distribution would expand for the first few hours of drift but would then remain about the same size thereafter instead of continuing to expand. This violates both a mariner's common sense about the uncertainty of how objects drift and the second law of thermodynamics.

Not only is it unrealistic for a sea current value to suddenly go from one extreme to another in the space of an hour, two successive samples in a series that are only a short distance apart in space and time should normally have a high degree of correlation. The degree of correlation between any two samples (successive or otherwise) should decrease as the distance between them in time and space increases. The same is true, in theory, for wind data. However, the "relaxation" times and distances for wind sample correlation may be much less than those for sea currents since winds tend to be less stable and more variable than currents. That is, it may be appropriate for wind samples to become uncorrelated with previous samples over shorter distances in space and time.

Finally, wind and current over an area generally exhibit some structure or organization. The sampling technique should not produce series of samples that violate known structures, even when the data grid is quite coarse.

8.1.1 Interpreting Gridded Vector Data

When data is associated with a geographic grid, there are at least two ways to interpret it. One of these is to assume the data applies to the volume inside a cell of the grid. For our purposes, this is called "block" data. The other interpretation is to assume the data applies to the intersection of (usually perpendicular) lines used to form the grid. We will call this "point" data. Figures 8-1 and 8-2 provide a limited illustration of the difference. The figures below are limited because they depict only the two spatial dimensions. There is the third equally important dimension of ***time***. Therefore, each cell is not a "square" but a "cube" and the grid lines form a three-dimensional (***x, y, t***) lattice.

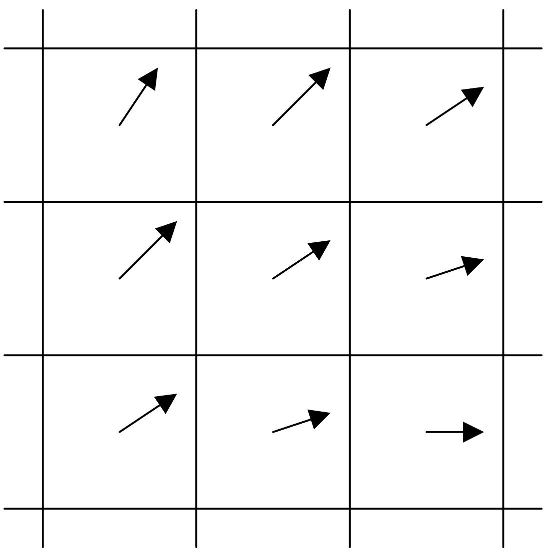


Figure 8-1 – "Block" Data

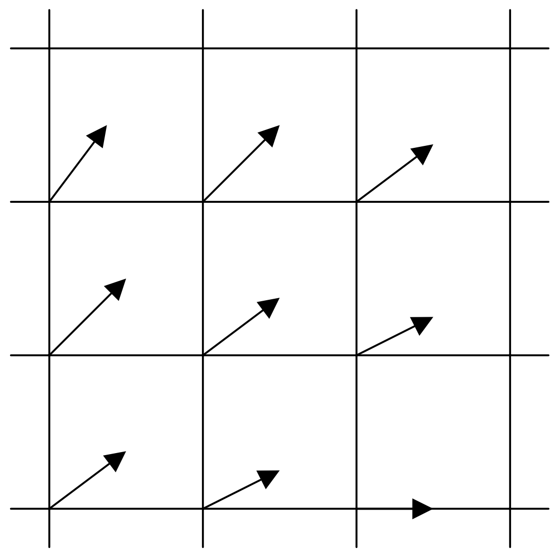


Figure 8-2 – "Point" Data

8.1.1.1 "Block" Data

Earlier in section 7.1 we described how sea current data for a gridded climatological database of sea current values were computed. Observations, mostly from ship set and drift reports, were accumulated for various positions and times of opportunity. These were then accumulated over many years and sorted into one-degree by one-degree by one-month cells. In each cell, these data were then processed to obtain a mean current for the cell along with the second-order statistics for the data contained in the cell. It is clear, therefore, that such climatological databases contain what we have chosen to call "block" data.

In theory, a replication should draw environmental data from the cell it is in for use in computing its drift over the next drift interval. This will create a new position for the replication that may or may not be in the same or an adjacent cell of the environmental data grid. Furthermore, different types of environmental data may be represented on different grids. Hence there are questions about how, when, and where to draw new environmental data samples in order to generate a realistic drift trajectory. Some of these questions are:

- If the replication remains in the same cell, should a completely new and independent set of environmental data samples be drawn, should use of the previously drawn values continue until the replication leaves the cell, or should there be some compromise between these two extremes?
- If the replication's new position falls in another cell, should that position be discarded in favor of the time and place of departure from the cell where the environmental data was drawn so new sample data may be drawn from the corresponding adjacent cell at that point? If so, should the new sample data be correlated in any way with the samples drawn for the previous segment of the drift trajectory?

The answers to these questions depend on several things all at once. These are the nature of the environmental data being sampled, the size of the grid cells in space and time, the length of the drift interval, and the rate of drift. To illustrate the situation faced by the modeler, consider the following: CASP (1.x) has a number of environmental data files available to it at several different resolutions in time and space. Global wind data is available on a one-degree by one-degree by 12-hour grid. Global sea current data is available on the same grid, except that every other value in the time dimension is a copy of its predecessor. That is, the data is actually on a one-degree by one-degree by 24-hour grid but data is copied to intermediate 12-hour grids for compatibility with the wind data. A global climatological sea current file is also available on a one-degree by one-degree by one-month grid. Regional sea current data for the western North Atlantic Ocean is also available on a 0.2 degree by 0.2 degree by 12/24-hour grid. Finally, climatological sea current data for the Florida Strait is available on a 0.1-degree by 0.1-degree by one-month grid.

If a sample current is drawn from the global climatological sea current file, then there is no good reason to draw another sample before the replication's drift takes it across a cell boundary (in either space or time). This could require anywhere from a few to many hours of drift, depending on the time, where the replication is situated in relation to the cell boundaries, and the direction of drift. However, since wind also contributes to drift and the wind data has a 12-hour resolution, there is good reason to re-sample the wind at least once every 12 hours. For the sake of simplicity and regularity, the modeler would like to re-sample all the environmental parameters every time a new position is computed. Otherwise, some bookkeeping functions are needed to keep track of and predict "sampling events" for each environmental data set as the replication drifts along.

One might think that all problems would be solved if the wind and sea current data were on the same one-degree by one-degree by 12-hour grid. As we can see from the earlier description of the available data files, this is possible. In fact, CASP (1.x) gives the global 12-hour sea current file precedence over the global climatological sea current file. However, this does not solve all problems. At higher drift rates, a replication near the boundary of two cells could penetrate quite some distance into the "next" cell in a 12-hour period based on data from the cell of origin, and perhaps even cross the "next" cell entirely without sampling from its data. This problem could become much more severe in regions where sea current data is available with much higher resolutions such as the western North Atlantic or Florida Strait. In those areas, 12 hours of drift could carry a replication across several cells, producing a drift trajectory that is a very poor, even unrealistic, representation of the available data.

At this point, it appears that the modeler would have to arrange for a sampling event to occur whenever a temporal or spatial cell boundary is encountered. This means the software will have to project the drift trajectory, compute the earliest time and place where a cell boundary will be encountered in any of the data files in use, place the replication at that boundary (or slightly further to get it unambiguously inside the next cell), draw new samples as appropriate, and then repeat the process until the desired datum time is reached. This would add considerable complexity and bookkeeping overhead to the software.

However, even this technique will not solve all the problems encountered with "block" data. When a new sample is drawn, it may send the replication back into the cell it just left. Normally this would not be a realistic drift trajectory. Even when this extreme is avoided, it will be difficult to avoid sudden unrealistically large changes near the cell boundaries, even when correlated sampling from the second-order statistics is used.

Finally, there is no way to meaningfully interpolate block data. The modeler is forced to use the data only in the discrete "chunks" provided.

CASP (1.x) treats all environmental data as "block" data even when the data is more properly interpreted as "point" data, such as the output of circulation models like those used to produce the wind and sea current products received from FNMOC. This has been the source of many of the problems CASP (1.x) has had in producing realistic drift trajectories.

8.1.1.2 "Point" Data

"Point" data is interpreted as being associated with a point where grid lines intersect in space and time. Most wind and current observations that are performed by either *in situ* or remote sensors are of the "point" type. The outputs of circulation models, whether on a global, regional, or local scale, are of the "point" data type.

"Point" data allow values to be interpolated to estimate the mean value at a time and place within the interior of a cell's volume. Interpolation schemes may be as simple as multi-dimensional linear interpolation or as complex as interpolation based on sophisticated models of natural structures (e.g., the Gulf Stream) present in the area for which the gridded data is being provided.

While interpolation of mean values seems straightforward enough, interpolating second-order statistics such as standard deviations does not seem so straightforward or meaningful. For example, one would normally be more uncertain about an interpolated mean value between data points in a grid than about the mean value at any of the surrounding data points, but interpolation would produce some intermediate, rather than greater, uncertainty estimate.

The following sampling algorithm is recommended for either uncorrelated or 100% correlated sampling when the covariance of the uncertainties is zero.

1. Draw a set of random variates from the appropriate distributions sufficient to sample each environmental parameter once, e.g., two independent normally distributed random variates for wind and two more for sea current.
2. Draw samples from the data associated with each of the eight vertices surrounding the rep's position in space and time using the same "single sample" set of random variates for each draw.
3. Linearly interpolate among the eight sample values in three dimensions (x, y, t) to obtain a sample value for the rep's position in space and time.

In more complex situations involving either non-zero covariances or a requirement for time-series "partial" correlation of samples for individual replications or both, the services of a professional statistician or stochastic modeler will be needed to develop the necessary algorithms.

8.1.1.3 Converting Between "Block" Data and "Point" Data

Although it may not be strictly correct to do so in a mathematical or statistical sense, it may nevertheless be useful from a practical standpoint to consider how "block" data may be re-interpreted as "point" data and vice versa.

"Point" data may be converted to "block" data by the simple expedient of assuming the point to which the data is referenced is the center of a cell that extends half the distance from the point to all immediately adjacent grid points. On a one-degree x one-degree x 12-hour grid, for example, data for a point (x, y, t) would be applied to the volume of a cell defined by (x±0.5°, y±0.5°, t±6hrs). This is how CASP (1.x) currently interprets "point" data from FNMOC products.

Going the other way, data for a cell bounded by (x_1,y_1,t_1), (x_1,y_2,t_1), (x_2,y_2,t_1), (x_2,y_1,t_1), (x_1,y_1,t_2), (x_1,y_2,t_2), (x_2,y_2,t_2), and (x_2,y_1,t_2) would be associated with the center point of the cell at (x,y,t) where $x = (x_1+x_2)/2$, $y = (y_1+y_2)/2$, and $t = (t_1+t_2)/2$. As discussed above, "point" data has some important computational and modeling advantages over "block" data.

8.1.2 Computing and Interpreting Sample Vectors

Second-order statistics describe the amount of uncertainty about the mean value. Given a distribution type for this uncertainty and its size, as represented by the second-order statistics, a random sample is generated by first drawing a random sample from the uncertainty distribution and then adding it to the mean value to get a sample value. For example, suppose the mean current was 000T at 1.0 knot and the uncertainty distributions of the u- and v-components were normal with a standard deviation of 0.25 knots in each case. Random draws from each of these distributions might produce an easterly sample "error" of 0.4 knots and a southerly sample "error" of 0.3 knots for a resultant sample "error" of 0.5 knots in the direction of 127T. Adding these errors to the mean current would produce a sample current having a u-component of 090T/0.4 knots and a v-component of 000T/0.7 knots. The resultant sample current for use in computing drift would then be about 060T/0.8 knots. Figure 8-3 illustrates the sampling method just described.

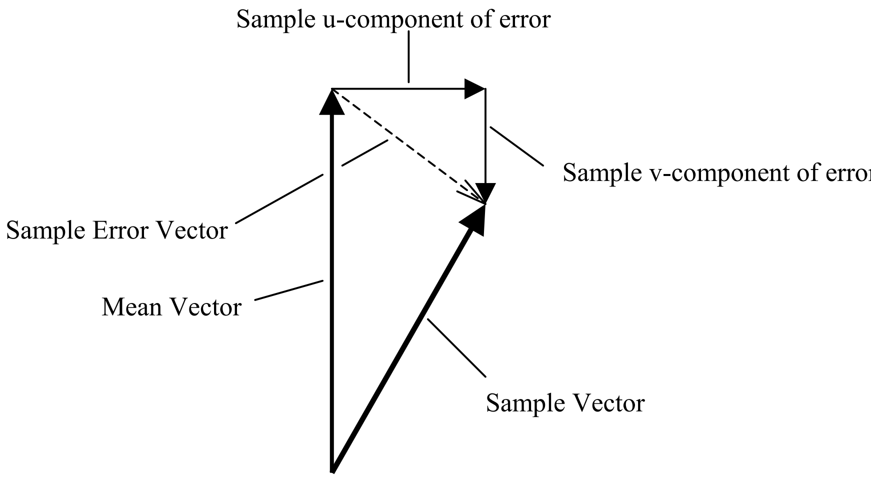


Figure 8-3 – Computing a Sample Vector

The sample current of 060T/0.8 knots would then be used for computing the replication's drift until it was time to re-interpolate from the present sample set or draw another sample set.

The next issue concerns the meaning that should be attached to the sample in the context of computing a drift trajectory. Having drawn the sample, the interpretation is that at the time and place for which the current was drawn, we are assuming that the actual current had the sample value. This begs the questions, "If the actual current at (x_1,y_1,t_1) was initially the same as the sample value, what does this imply about the current field in the vicinity of (x_1,y_1,t_1)? In particular, just how much correlation must exist between the initial sample and the sample drawn for (x_2,y_2,t_2) to produce a realistic drift trajectory?" These are critical, but not necessarily easy, questions to answer. As stated previously, a great deal depends on the nature of the phenomenon being sampled, the spatial and temporal resolutions of the data grid, the rate of drift, and the time step or interval between samples. The answers also depend, of course, on how much "distance" in time and space separate (x_1, y_1, t_1) from (x_2, y_2, t_2).

To illustrate two possible interpretations that could be applied to the same sample current, consider the following situation. A life raft goes adrift between Key West and the north coast of Cuba. The mean current there is in an easterly direction. Later it will make a left turn as it encounters the Bahama Banks and will flow in a northerly direction between Florida and the Bahamas before striking out for northern Europe in a generally northeasterly direction (with numerous meanders along the way). Suppose the climatological mean current at the raft's initial location was 090T/3.0 knots. Also suppose an error vector is drawn that has a value of 090T/0.2 knots. The sample current would then be 090/3.2 knots. The following two equally plausible interpretations may be given to the error vector that was drawn:

- The sample current represents an occasion when the mean currents in the strait were 0.2 knots faster in the mean direction of flow than the climatological averages.
- The sample current represents an occasion when the currents in the strait had a more easterly component, by 0.2 knots, than the climatological average values.

Initially, the two methods will produce the same result, but as the current turns northward, the two drift trajectories will become increasingly different. At the same time, both trajectories seem like reasonable interpretations of the data and random draws. In one case, the interpretation is that the actual current in the strait is stronger than the long-term average value along the average lines of flow and in the other case the interpretation is that the actual current is more easterly than the long-term average current.

CASP (1.x) "solved" this issue in 1994 by arranging for each replication to initially draw its own set of random values for sampling from the error distributions, and then retain that same set for all future samples. Furthermore, half the replications assumed the circular normal error distributions for wind and current had axes aligned with parallels and meridians, while the other half assumed the axes were aligned with and perpendicular to the mean vector's direction. Hence both of the above interpretations were covered. This technique is still in use. However, it is probably not the most statistically correct sampling technique. For one thing, it assumes that, on an individual replication-by-replication basis, the initial sea current error sample characterizes the "error" across the entire set of sea current databases. The "random walk" alternative, on the other hand, assumes the initial error sample provides no indication at all regarding what may be reasonably assumed about nearby currents no matter how close they might be to the sample location in space and time, thereby allowing substantial unrealistic discontinuities in local current flow to appear in a replication's drift trajectory. There may be no perfectly correct solution, but there is almost certainly a better solution somewhere between the two extremes of 100% correlated sampling for all time on an individual replication basis, and completely uncorrelated sampling over all time and space intervals, no matter how short. Finding an improved solution will require the services of an expert in the modeling of natural phenomena as a stochastic process.

8.1.3 Wind Current

Currents in the ocean, and also near shore, are affected by the local wind blowing across the water's surface. The friction between the wind and the water creates waves and also tends to create or change water movement in the "mixed layer" from the surface down to the depth affected by the wind-induced turbulence near the surface, except in shallower water where the "mixed layer" depth is truncated by the bottom and bottom friction comes into play. The component of the surface current due to the stress placed on the surface water by local winds is called local wind-driven current, or just wind current for short.

When, where, and how to estimate the wind current for SAR situations has always been a difficult proposition, and several methods have been tried. The following list contains pertinent, and generally unanswered (for SAR personnel) questions about wind current.

- Sea current databases that are based on climatology are generally assumed to have no wind current component. This is because it is assumed that the wind varies sufficiently over the time scales involved to effectively cancel out any wind driven component when the water current observations are averaged. However, this is not true in areas covered by a trade winds regime where the wind blows steadily at the same speed and from the

- same direction almost all the time. Should a wind driven component be added to currents under the trade winds?
- Do the sea current products from FNMOC and other sources contain the wind driven component? That is, do these products provide the total water current?

Another problem is that regardless of the answer to the second question above, CASP (1.x) computes a database of wind currents from FNMOC wind data twice a day following receipt of the FNMOC product set. When drift trajectories are computed, wind current from this database is used with the FNMOC global and regional sea current products. It is not clear that this should be the case. In addition, there is no provision for computing wind current from wind data supplied by the user. Even if such wind data is used to compute leeway, the wind current still comes from the wind current database that was built with wind data from FNMOC, not the user's wind data. Only if the user can supply total water current, such as that obtained from datum marker buoys, can the wind current issue be resolved—by eliminating it from the drift computations altogether.

The unsatisfactory situation with wind current needs to be resolved with the aid of professional oceanographers. Oceanographic models use wind current estimation algorithms that are different from any of those that have ever been used for SAR. If wind current were to be used at all, it would seem appropriate to compute its value using standard oceanographic algorithms.

8.1.4 Land Recognition

An important environmental feature that is often overlooked in maritime SAR models is land. The original CASP had no land recognition, just as AMS has no land recognition today. It was up to the user to recognize when a portion of a CASP probability map fell over land and to ignore that portion. However, replications could drift ashore a considerable distance and then drift back to sea again, drift across islands or peninsulas, etc. This was clearly unsatisfactory.

In the late 1970s the land recognition issue was addressed, at least in a crude form. Land was encoded into the global one-degree by one-degree by one-month climatological sea current database. If the majority of a cell was on land, it was assigned a sea current direction of 999. Similar changes were made to the regional climatological sea current databases in use at the time. This made it possible for the CASP drift update module to recognize when a replication's position was on land and to leave it there. In the mid-1990s a separate "bit-mapped" land database on a 0.1 x 0.1 degree grid was implemented for CASP (1.x) and used to set a rep's status to "landed" whenever a drift update placed it ashore.

The next step would be to use a vector-based shoreline model consisting of connected line segments rather than "pixels." This would allow the options of making the shoreline either "sticky" or "slippery." These options are frequently part of oil-spill trajectory and other near-shore drift models. If the shoreline were "sticky," then replications would stop drifting at the water's edge and remain wherever they "ran aground." If the shoreline were "slippery," then replications would be allowed to "slide" along it in the general direction the drift would have taken them otherwise as if caught in a long-shore current. They would also be subject to being carried back out to sea if winds and/or currents were encountered that would produce such an effect.

8.1.5 Tidal Currents

Near shore, and certainly in bays, sounds and estuaries, tidal influences can be significant and are often the most significant component of an object's drift. CASP has never done an adequate job of representing tidal influences. Even AMS uses only the equivalent of NOAA tide tables where the primary emphasis is on factors important for safe navigation in restricted waters rather than on how objects drift. NOAA tables provide data for irregularly placed "stations" that may be appropriate for navigation purposes but which make estimating accurate drift trajectories quite difficult. Such tables are woefully inadequate for search planning purposes.

Fortunately, hydrodynamic models do exist for many major bays, sounds, and estuaries around the United States. The primary reasons for their development have been academic study and drift trajectory modeling for various pollutants. Like the NOAA tide tables, these models are based on the influences of sun, moon, and hydrography and values may be computed well into the future. Unlike the NOAA tables, these models produce data for points on regular grids that cover the entire body of water rather than at just a few "stations." Usually wind stress can be added to modify the outputs of these models to account for local wind influences on shallow water, which are quite different from local wind influences in the deep ocean.

The replacement search planning tool should take full advantage of the considerable advances in near-shore current modeling made in recent years. This could take the form of either purchasing data sets for coastal areas or purchasing the models themselves and operating them as "black boxes" to produce products for the new search planning tool well in advance of when they might be needed. Recent interest in the near-shore environment shown by the U. S. Navy might also allow the Coast Guard to obtain near-shore data from them on a regular basis just as offshore products are now obtained from FNMOC.

8.1.6 Resolving Environmental Data Boundary Conditions

At present, the CASP (1.x) sea current database consists of a mixture of climatological data and data from models that produce both analyses of recent observations from many sources and predictions of what the currents will be in the next few days. Both climatological and model data have subsets of grids of different resolutions. Furthermore, these different subsets often overlap and always at least form some type of boundary across which a search planning tool must be able to compute a reasonable drift trajectory.

CASP (1.x) currently uses a geographic-based hierarchical approach to this issue. For example, if the replication's position is in the area covered by the Florida Strait climatological file south of 28N, then Florida Strait climatology is used (0.1-degree grid). High spatial resolution climatology in this region has so far proven superior to more nearly real-time products at lower spatial resolutions. North of 28N and west of 50W up to the U.S. coast, the regional FNMOC "Gulf Stream" file is used (0.2-degree grid). Between 26N and 28N, a replication leaving the Florida Strait file to the east will also draw data from the regional FNMOC "Gulf Stream" product. Elsewhere, a replication leaving the area covered by the Florida Strait file will find itself drawing data from the global FNMOC sea current file (one-degree grid). There is great potential for problems to arise when drift trajectories cross any of the boundaries as there is no mechanism currently in place for ensuring the transition from one file to the next is smooth and realistic.

The ideal situation, of course, would be either a single high-resolution product covering the globe or, since that is not feasible at this time, a single consistent set of products at various resolutions that smoothly blend into one another. Until such a product set is available, some method for "smoothing the edges" between sea current files will be needed. Probably a simple averaging or interpolation scheme along the boundaries is the best that can be done under the present circumstances.

Another related issue arises when on scene observations are made. The winds reported by SRUs actually at the scene may not be the same as those obtained from FNMOC. Similarly for data reported by nearby NOAA weather buoys over short time intervals. A datum marker buoy may show the current along its trajectory to be different from that contained in the sea current files covering that area. In all these situations, the search planner is faced with a dilemma. At the time and place of an actual and reliably accurate observation, that observation is clearly superior to either climatology or the outputs of any model. However, on scene observations constitute only single, or at most a handful, of data points. The search planner is not generally in a position to assess over what region of space and time the observations should be applied in place of data from other sources. He also is not generally in a position to "blend" the on scene data with data from other sources into a single coherent environmental picture.

In an ideal world, on scene observations would all be rapidly and accurately reported to places like FNMOC and NOAA who would then see a "data bloom" over the search area. This additional data would then go into their models, along with data from their many usual sources, to produce a better product in and near the search area than elsewhere.

The USCG Research and Development Center was examining at one point the feasibility of developing gridded data over limited regions of space and time from on scene observations using an objective analysis technique. Another objective of the Ocean Prediction System (OPS) was to examine the feasibility of blending on scene observations with other products, such as those from FNMOC, in an automated fashion.

The U. S. Navy uses a different solution. Meteorologists and oceanographers at sea with the fleet modify FNMOC model outputs according to local observations to produce a more accurate tactical environmental picture for the local region of space and time. An equivalent function for the Coast Guard would require, at a minimum, a small tactical staff of meteorologists and oceanographers so at least one would always be ready to respond to the needs of a SAR case. Within the Coast Guard, the International Ice Patrol staff performs a similar function for its own purposes during the ice season as they track and predict the drift of icebergs near the Grand Banks.

Given better products that actually incorporate on scene observations, the search planner would not be faced with decisions about data usage that he is ill-equipped to make. In the meanwhile, the Coast Guard needs to develop and promulgate policies on environmental data sources and methods of use that are deemed suitable for SAR search planning.

8.2 Drift Modeling

Once the necessary decisions have been made on how to interpret the available environmental data, along with any new environmental data sets that may become available, a method for using that data to compute sample drift trajectories is needed. The drift problem is very difficult to solve using just the physics of the situation. This is due to the complex nature of drift. Stated as a physics problem, we wish to predict the motion of a very small solid object that is suspended at the turbulent interface between two huge fluid masses—the ocean and the atmosphere. There are no exact solutions to this problem because there is insufficient information about all the forces acting on the object at every point in time.

Fortunately, there are adequate empirical solutions to the drift problem available, and these have changed relatively little over the years. The manual method recognizes total water current and leeway as the primary factors affecting search object drift. Total water current may be obtained from direct observations, such as datum marker buoy trajectories, or from models. However it may also require computation from constituent parts, such as average geostrophic sea current from climatology and local wind driven current. In any case, the basic drift update algorithm is to obtain an estimate of the velocity vectors in knots for total water current (e.g., by random sampling and interpolation) and leeway (e.g., by random sampling and interpolation of wind data that is then used along with estimated leeway parameters in an appropriate leeway formula) for the replication's location in time and space, add the two vectors, multiply the resultant by the number of hours in the drift sub-interval to get a distance, and compute the latitude and longitude of the new position based on the computed direction and distance the object would have drifted from its position at the beginning of the interval. For earth coordinates, appropriate navigational algorithms must be used to compute the updated position. This process is then repeated for as many sub-intervals as necessary to reach the desired datum time. In short, all search planning tools use the same basic drift update algorithm although some details may vary regarding how certain subordinate computations are done. A good example of such a "detail" is how wind current is estimated. An even better example is the time step used for computing intermediate datums. The time step for a true manual method (and also AMS) tends to be on the order of hours to days, depending on how soon after an incident SRUs can arrive on scene. SARP used a one-hour time step just as CASP (1.x) does today.

The basic CASP (1.x) drift update algorithms are acceptable for continued use except in three specific areas. One of these is the issue of correlated sampling of environmental data along individual replication drift trajectories discussed in paragraph 8.1.2 above. The other is the handling of leeway. Recent research done by the USCG Research and Development Center [Allen & Plourde, 1999] has produced new leeway modeling techniques, new uncertainty distributions, new leeway parameters, and a substantially increased "taxonomy" of objects for which leeway parameters are available. The results of this research need to be incorporated into the new search planning tool. Finally there is the issue of "point" vs. "block" data. Interpreting gridded data as "point" data and interpolating to obtain values for positions inside grid cells is recommended.

CASP (1.x) currently operates on a one-hour time step for drift sub-interval updates. This is adequate for all of the environmental data sets presently available to CASP and more than adequate for most of them. However, the time step for drift updates should be kept adjustable so that higher resolution data may be used if and when it becomes available.

8.3 "Reverse" Drift

Whenever identifiable debris related to a missing craft is discovered, it is an important clue. First, it usually removes any doubt that may have remained about whether a distress has occurred. Second, close examination of the debris may provide clues about the nature of the distress and possibly (though not usually) an indication of when the distress occurred. Finally, locating debris may provide some information about where (and when) the distress occurred if a way can be found to answer the questions, "Where did this object come from after being separated from the missing craft and, if not already known from other sources, when did it start?" Answers to these questions may substantially reduce the uncertainty about where and when the distress took place.

Since all drift update algorithms used to date have been linear in a mathematical sense, the "reverse drift" problem has been addressed by simply reversing all the wind and current vectors and running the drift algorithms "backward" in time. This has actually worked remarkably well in some cases. However, one gets the uncomfortable feeling that this technique may not always be reliable. It is possible to make a movie of a delicate vase falling from a table on to a hard floor and shattering. It is then possible to run the movie backwards and watch all the pieces jump up from the floor and reassemble themselves into a complete, unblemished vase on the table. However, such things do not occur in nature and it would be impossible, without the film or other evidence, to reconstruct the path a particular shard followed given only the shard's location on the floor. The reverse drift problem is something like finding a single shard after all the others have been swept up and then trying to ascertain exactly where the vase struck the floor based only on that one piece of information and estimates of the forces that may have been involved. There are probably a great many places where a vase could have struck the floor and left a shard in exactly the same position where our shard was found.

Perhaps a better way to approach the reverse drift problem is to add the debris to the list of objects associated with each scenario. Then the drift problem for the debris could be run forward through time. Those replications coming close to the location where the debris was found could then be queried for their scenarios and starting locations and times. This would give a more complete picture of where and when the debris could have originated. Such a picture could then be analyzed for correlations between it and the various scenarios being considered. A high degree of correlation between a scenario or some proper subset of it and the set of possible debris trajectories would indicate the scenario (or a subset) should be more heavily weighted while a low correlation would indicate the opposite. Low correlation with all scenarios might indicate that another scenario needed to be added. A high correlation with all scenarios would tend to confirm the scenarios were valid but it would also mean that finding the debris did not add much if any additional information about what happened to the missing craft.

Both of the above solutions presume that a reasonably accurate estimate of the leeway characteristics of the debris is available. This may or may not be the case. In the event that it is not the case, then a range of values for leeway parameters may be needed as well as ranges of starting locations and times. Probably the easiest method for the users is to ask them to choose two objects from the existing taxonomy that represent, in their judgment, the range of debris leeway characteristics. If the leeway of the debris could be observed or estimated from its total drift over some period following its initial discovery, a more accurate leeway estimate could be made.

8.4 *Tuning Issues*

Like any complex machine, a complex simulation requires tuning in order to be at its best. Furthermore, all the "pieces and parts" must be consistent and properly "mated." In some respects this is far more difficult to do with software than with machinery. If parts of a machine do not fit properly, either the machine cannot be assembled in the first place or, if it can be built, it will not operate. In other words, there is usually a strong indication that something is wrong. With software, systems can be built that appear to operate properly but do not, and the problem may not manifest itself until a catastrophic event occurs. A few years ago such an event did occur when a multi-million dollar unmanned mission to Mars crashed on the surface due to a programming error in the software that was supposed to control the landing. The error was simply a failure to use consistent units of measure throughout the software system. In at least one place there was a failure to convert from metric to English units or vice versa. Similar lapses have appeared in CASP's history from time to time.

8.4.1 Replication Density

CASP (1.x) attempts to achieve an average replication density of ten replications per square nautical mile for each distribution representing a situation. However, CASP (1.x) also has a ceiling value of 20,000 randomly placed replications per situation, plus one "datum rep" at each position used to define the situation's location(s). Thus if the distribution covers one square degree at the equator (3,600 square nautical miles), the average density would only be about 5.6 replications per square nautical mile. Simulating the possible distress locations of a vessel overdue on a voyage from New York to Bermuda could easily result in an average replication density of much less than one per square mile.

In addition to large search problems there is also the variation in replication density that naturally comes with non-uniform distributions like the bivariate normal distribution. As distance from the center increases, replication density necessarily must decrease to the vanishing point. So, even if the average density over the area containing replications is relatively high, the densities over small sub-areas will vary from very high to quite low.

Finally, replication density will generally decrease everywhere over time as the distribution spreads out and becomes increasingly diffuse. This is unavoidable unless a mechanism is found for replenishing replication densities without affecting the probability density distribution.

Replication density is important when it comes to sampling from the various distributions describing the environment and the properties of the object the replication represents. The desirable condition is to have enough replications everywhere so that the combined distributions of wind, current, leeway, and search object state transitions may be fairly represented. If there are only two or three replications in a cell of the environmental data grid, it will be impossible to obtain a representative sample set for even one variable. It is not possible, for example, to adequately represent a circular bivariate normal distribution of wind error vectors about the mean wind with only three random samples from the error distribution.

Heretofore, CASP has ignored the replication density issue as it relates to the ability to adequately sample from the environmental and leeway parameters. It has also ignored state transitions since no attempt has yet been made to model them. In spite of this, CASP has actually worked reasonably well over the years when relatively bug-free (which has not been often). Therefore, ignoring the replication density issue in an environmental sampling context may still be a viable alternative.

Other possible, but not necessarily feasible, alternatives include:

- Somehow separating probability density from replication density so a sufficient number of samples may be drawn everywhere, even in areas having a low probability density.
- Adjusting the sampling technique as a function of replication density on a cell-by-cell basis over the environmental data grids. For example, if there were only a single replication in a cell, it would be forced to use only the mean values for environmental parameters since the means would be the most representative. However, this might cause problems if correlated sampling is used to create drift trajectories that represent a particular environmental state (e.g., faster than average current flow).

It is worth noting that as environmental data sets migrate to finer and finer grids in space and time, the error distributions associated with individual values at grid points should become smaller and smaller. At some point the environmental features may become sufficiently well defined on a sufficiently small scale that sampling from the correspondingly small error distributions would not substantially affect the simulation results. If samples from error distributions about environmental data are eliminated due to the availability of high-resolution data sets, then so are all the sampling issues discussed above.

For the time being, it is recommended that the replication density versus sampling issue be ignored and that the replacement search planning tool simply use as many replications as the computing resources and response time requirements will permit.

8.4.2 Time Step and Relaxation Times

The time step, drift rate, and environmental data cell size are closely related to the accuracy of individual drift trajectories and the resulting probability density distribution on search object location. Adjusting the length of the time step serves two purposes. When using point data and interpolation, a small time step generally produces a more realistic and accurate drift trajectory.

If new samples are drawn for every time step, then it becomes the effective relaxation time for uncorrelated sampling. If the time step is too long, a replication may use data from one location to compute a drift vector that carries it across several locations in the environmental data grid without allowing those values to have an appropriate effect on the drift trajectory. The only remedy for this situation is to shorten the time step. This in turn would increase the need for correlated sampling due to decreased distances between samples in space and time.

On the other hand, having a time step that is too short without correlated sampling could cause the replication to draw many successive values from the same error distributions. This would cause mean errors to tend toward zero. The net effect would be the same as just using the mean environmental parameters without bothering to sample from the error distributions at all. This issue could be addressed with a correlated sampling scheme where the correlation with previous samples faded with the replication's distance from them in space and time and at a rate commensurate with the volume of the environmental data cells. Alternatively, a separate relaxation time could be introduced that would cause new samples to be drawn only at time intervals appropriately longer than the time step.

The essential requirement is that every computed drift trajectory must be realistic and the collection of all computed drift trajectories must result in a reasonably accurate representation of the distribution of possible search object locations and states at any given time. From another perspective, we might say that no drift trajectory (including any state transitions) may be obviously absurd in view of the combined uncertainties and known properties of the factors involved.

8.5 Random Number Generators

Perhaps the most important basic function for any Monte Carlo simulation is a reliable and statistically valid source of random numbers. Many texts on the subject of generating random (or more precisely pseudo-random) numbers contain strong cautions about using system-supplied random number generators. Almost all such generators use the linear congruential method for generating uniform random deviates. Deviates with other distributions (e.g., normal) are generally produced by performing transformations on uniform random deviates. This means the uniform random number generator is the key to all random numbers in the entire system. The linear congruential method is very fast, but it has certain weaknesses. For example, it is not free of sequential correlation on successive calls. If k random numbers are used to plot points in k-dimensional space, the points will not tend to fill the k-dimensional space uniformly but will lie on "planes" in k-1 dimensions. There will be at most about $m^{1/k}$ such planes, where m is the modulus of the linear congruential method in use. If the parameters for the method are not chosen with great care, there will be many fewer than that. In addition, linear congruential generators tend to produce numbers where the low-order bits are much less random than the high-order bits.

For drift updates, CASP (1.x) draws random numbers in groups of six for each replication—two for drawing a wind sample, two for drawing a current sample, and two for sampling the leeway parameters (the multiplier and the angle off the down wind direction). The weaknesses of the linear congruential method could have a significant adverse impact on the simulation by producing unintentional and unwanted correlations among the six values and between groups of values.

CASP (1.x) uses the Hewlett-Packard system-supplied random number generators RAND for uniformly distributed random deviates and GRAN for normally distributed random deviates. It is unlikely that the Coast Guard has subjected either to the standard list of statistical tests for such generators to ferret out any correlations that might present problems for CASP's use of the values they produce. RAND almost certainly uses the linear congruential method and GRAN almost certainly adds a standard transformation to RAND outputs to produce a normal distribution. Therefore, CASP (1.x) is at least potentially susceptible to the weaknesses of the linear congruential method.

It is strongly recommended that the new search planning tool use a proven portable random number generator rather than continuing to rely on system-supplied generators. Algorithms for such generators may be found in a number of texts, including Press, et al [1992,1993] and Knuth [1997,1998]. Further discussions may be found at several web sites devoted to random number generation with computers.

It is also recommended that provision be made for several independent random number strings to be available for the developer's use. Simulation languages like GPSS generally make several random number generators available, or more precisely, several instantiations of the same algorithm with a different, but carefully chosen, set of defining constants (e.g., multiplier, increment, and modulus for the linear congruential method). This provides added protection against unintended correlations among samples.

Although it may sound counterintuitive or even oxymoronic, the random number algorithm chosen should be capable of being reset so it can be made to repeatedly generate the same sequence of "random" numbers. This feature is important for developers and others who may need to study specific aspects of the model not related to the random number sequence. It is much easier to isolate and view the effects of a change to an input or an individual module if one does not have to deal with a new set of random samples every time. As an aside, the repeatability feature of the current system-supplied random number generators is used to achieve the 100% correlated sampling along individual drift trajectories described earlier.

9 Geodesy and Coordinate Systems

The new search planning tool will necessarily consist of a great many mathematical algorithms and computations. The ideal situation would be to establish a single standard coordinate system along with a single standard set of units of measure that all modules would use. However, there will be a need to use at least two standard coordinate systems. One of these is the standard navigational coordinate system familiar to users. This will be the one used for communications between the users and the new search planning tool. The other coordinate system will be standard Cartesian coordinates to facilitate implementation of the mathematical algorithms. A well-defined transformation from one to the other and back again will have to be established. This transformation will be completely internal to the software and will be seen by none but the software developers. Data received from the user and other sources would be converted to the "computational" coordinates used by the mathematical algorithms. Once the computing was complete, the outputs of the mathematical algorithms would be transformed back into normal navigational coordinates for display to the user. This is the most straightforward and least error-prone way to handle the differences between the normal navigational coordinates familiar to the user and the standard coordinates used for expressing mathematical formulas and algorithms. Standard Cartesian coordinates are used by all of the procedures in the mathematical libraries supplied with computer programming languages.

The need for additional coordinate systems may arise, but the number of coordinate systems and units of measure in use should be kept to a minimum. Conversions among coordinate systems should be explicitly defined and handled through specific modules designed for the purpose. Major subsystems should strive for the ideal situation of maintaining a single coordinate system and set of units within that subsystem.

Failure to establish standards for coordinate systems and units of measure at the outset will only serve to complicate both the initial development and future maintenance of the tool. The result will be (and has been) increased risk of programming errors over the entire software lifecycle.

9.1 Earth Coordinates

Positions on the Earth's surface are generally expressed in terms of latitude and longitude, which are spherical coordinates. Latitude is the angle formed by a line segment from the position to the center of the Earth and the plane that cuts the earth in half at the equator. Longitude is the angle formed by the same line segment and the plane that cuts the earth in half along the prime meridian/international date line (0 and 180 degrees east and west longitude).

Although latitude and longitude, together with the earth's radius, are spherical coordinates for points on the surface, the Earth is not a perfect sphere. That is, the Earth's radius is not the same everywhere. For example, the mean equatorial radius and mean polar radius are not exactly the same. The mathematical shape that best describes the shape of the Earth yet is relatively easy to deal with mathematically is the ellipsoid of revolution, also known as an oblate spheroid. A *datum*, in the cartographic sense, is the combination of a model for the Earth's shape and a reference point on which the locations of all other points are based. For many years charts and maps for North America were based on the Clarke ellipsoid of 1866 and a reference position at Meades

Ranch in Kansas. This is generally referred to as the North American Datum of 1927 (NAD 27). The most recent standard, and the one to which virtually all others are now referenced, is the World Geodetic Reference System of 1984 (WGS-84). This system is referenced to the center of the Earth and is therefore called a geocentric datum. The U. S. National Imagery and Mapping Agency (NIMA) uses this datum, which is based on surveys performed with electronic technology that is still partially classified.

The WGS-84 Coordinate System is three-dimensional, right-handed, Earth-centered Earth-fixed orthogonal (Cartesian) coordinate system. The origin lies at the center of mass for the whole Earth, including oceans and atmosphere. For all practical navigational purposes, the positive Z-axis pierces the Earth's surface at the north geographic pole, the positive X-axis pierces the Earth's surface where the prime meridian and equator intersect, and the positive Y-axis pierces the Earth's surface where 90 degrees east longitude intersects the equator. For high-precision geodetic work, the definition of the WGS-84 Coordinate System is somewhat more complex.

The WGS-84 system, which includes local adjustments to the basic ellipsoid model of the earth, allows for very accurate positioning on the order of meters or even centimeters. This level of accuracy is not needed for the new search planning tool. For the bulk of the computations, it will be sufficient to regard the Earth as a perfect sphere having a circumference of exactly 21,600 nautical miles (the number of minutes of arc in 360 degrees) along any great circle. For long intended tracks of overdue or unreported craft, it may be necessary to use a more accurate model of the Earth's shape, but only if a sensitivity analysis shows this to be the case.

9.2 Chart and Map "Projections"

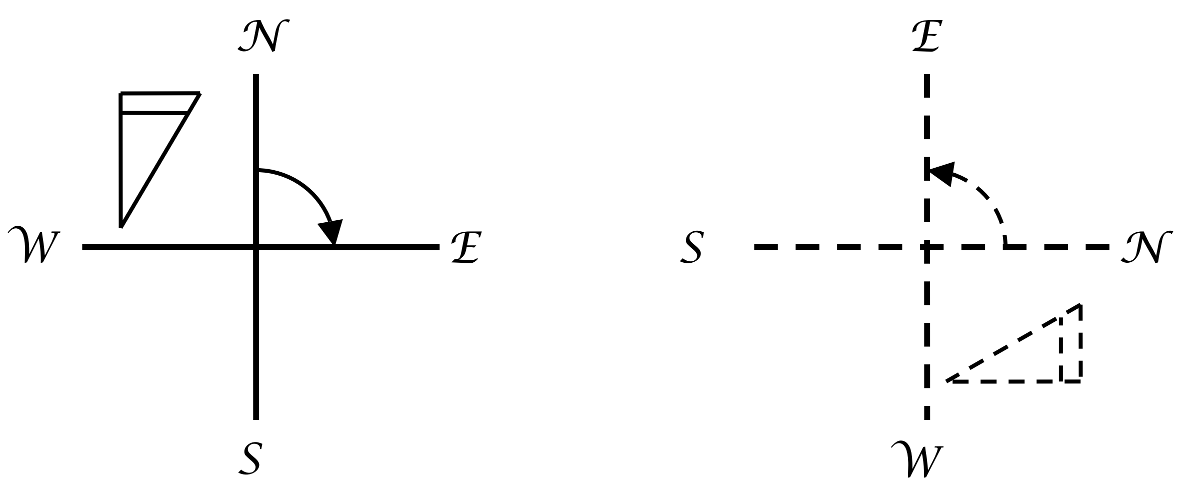
The basic frame of reference for search planners is the nautical chart or other map on a flat surface. When the Earth's surface or portions thereof are depicted on a flat surface such as a paper chart or a computer screen, it is necessary to "project" or transform points from three-dimensional Earth coordinates to two-dimensional Cartesian coordinates. It is impossible to represent a portion of a sphere's surface exactly on a flat surface. Therefore, all chart and map "projections" involve distortions of one kind or another. Some "projections" are not projections in the strict geometric sense but all have well defined mathematical transformations for mapping the area of interest from a spherical surface to a flat one. Different projections have different desirable characteristics. For example, on a Mercator chart, meridians appear as straight parallel lines perpendicular to the parallels of latitude and rhumb lines appear as straight lines regardless of direction. However, areas become quite enlarged and distorted as the poles are approached and the poles themselves cannot be represented. Mercator charts are not suitable for navigation in polar regions but they are the most commonly used navigational charts elsewhere. A variation on the Lambert conformal projection is often used for polar navigation. Another commonly used projection for polar regions is the polar stereographic projection. On both of these projections, plots of great circles are essentially straight lines.

Navigational computations may be, and usually are, done in the two-dimensional frames of reference represented by charts. However, the elements of these frames do not correspond to a normal Cartesian frame of reference. For example, Mercator projections normally have north at the top of the chart, east to the right, south at the bottom and west to the left. If this were all we had

had to worry about, then it would be natural make north the positive y-axis and east the positive x-axis in a normal Cartesian frame of reference. However, courses (directions, angles) are measured on a chart clockwise from true north (000T). This would correspond to clockwise from the positive y-axis instead of the mathematical convention of counterclockwise from the positive x-axis. This means that angles on a chart are shifted by 90 degrees and measured in the opposite direction as the mathematical convention. In addition, by nautical convention positions are expressed as ordered pairs of (latitude, longitude). In our first attempt at a Cartesian system this translates to (y, x) rather than the mathematical convention of (x, y).

CASP (1.x) addressed these issues in the following fashion. The normal chart orientation was rotated 90 degrees clockwise so that north was to the right and west was at the top. Then, the positive x-axis in the Cartesian equivalent was in the same direction as north (to the right) and the positive y-axis was in the same direction as west (toward the top). Angles were measured counterclockwise from the positive x-axis (north). In other words, both took "north" to be zero degrees but they measured angles in opposite directions. This meant that true compass angles had to be converted to Cartesian angles by subtracting them from 360 degrees, and vice versa. This was the only coordinate system conversion that was necessary and once it was made, then all values and mathematical computations could be treated just as they would be in a normal Cartesian frame of reference with all the normal mathematical conventions. This greatly reduced the opportunity for programming errors.

The CASP (1.x) frame of reference can be improved upon to remove even the conversion from true compass to Cartesian angle. If the chart is turned over so the reverse (presumably blank) side of the chart is facing up with north still to the right but east at the top, then the Cartesian angles and compass angles match up and no conversions at all are necessary, as shown in Figure 9-1 below.



a: Normal Mercator Chart Configuration

b: Chart Viewed From the Reverse Side And Rotated to Align with Cartesian Coordinates

Figure 9-1: Normal and "Cartesian" Views of a Mercator Chart

In this case, the positive x-axis and north remain to the right but now the positive y-axis is in an easterly direction and east is at the top at an angle of 90 degrees for both true compass and Cartesian. The nautical convention of (latitude, longitude) pairs now corresponds to (x, y) Cartesian pairs when east is regarded as the positive y-direction and north is regarded as the positive x-direction.

Although it would be possible to use a real chart in the configuration shown in Figure 9-1b by placing it on a surface illuminated from below so the image on the obverse side could be seen, it would be impractical and unnecessary. However, making the transformation from the coordinates on the left to the ones on the right would be convenient, practical, efficient, and less error-prone for the purposes of programming a computer to do the various navigational computations common to the use of Mercator charts. All the software developers would need to remember about the two-dimensional coordinate system used for computations are the following:

- The prime meridian corresponds to the x-axis,
- The equator corresponds to the y-axis,
- North and east are positive, south and west are negative,
- Distances along meridians from the equator (latitudes) correspond to x-coordinates,
- Distances along parallels from the prime meridian (when adjusted for latitude these become longitudes) correspond to y-coordinates, and
- True compass angles and Cartesian angles are the same.

All the usual mathematical conventions will then apply.

There is one other issue that affects “mapping” the globe to a flat surface. In Cartesian coordinates, 360 units separate -180 and 180 on the y-axis. On the globe, they (180W and 180E) are in the same place, also known as the geographic International Date Line. This can lead to significant difficulties near 180E/W unless specific steps are taken to deal with the situation. Probably the easiest way to deal with it is to temporarily shift the coordinate system axes through 180 degrees of longitude, placing the origin at the intersection of the International Date Line and the equator. This shift would be done whenever the center of the replication distribution had a longitude between 90E and 180E or between 90W and 180W. The International Date Line would then be treated as the prime meridian and actual longitudes would be temporarily converted to “relative” longitudes in degrees east or west of the date line. In this situation, the longitude relative to the date line would equal either the actual east (positive) longitude minus 180, converting actual east longitudes to relative west (negative) longitudes, or the actual west (negative) longitude plus 180, converting actual west (negative) longitudes to relative east (positive) longitudes. When the computations were complete, the reverse conversion would be done as needed to restore all positions to the standard coordinate system having its origin at 0N, 0W.

9.3 The Sailings

The most obvious aspect of the new search planning tool that requires a standard coordinate system is the need to simulate movements on and over the earth’s surface.

Simulating movements requires the ability to compute:

- A new position on the earth's surface that is a given direction and distance from another point on the earth's surface along either a rhumb line or great circle,
- The distance and direction from one point on the earth's surface to another along either a rhumb line or a great circle,
- Intermediate points along a great circle or rhumb line route between two points on the earth's surface.

In addition to rhumb lines and great circles between points, there are also several models of the earth's shape to consider. As with most things, there is a tradeoff between accuracy, computing time, and complexity. Since navigational computations may be buried deep in the model and invoked literally millions of times during a single simulation run, computational simplicity and efficiency may well become an issue. This is the primary reason for modeling the earth as a perfect sphere in this application.

Bowditch [2002] lists a number of "sailings" or methods for navigating from one place to another. The most applicable of these for our purposes are the mid-latitude, Mercator (rhumb line), and great circle sailings. These three sailings are described below in terms of how to compute a new position for each of them when given a starting position, a course (or initial course in the case of a great circle sailing), and a distance. Algorithms for other computations of interest, such as computing the distance and (initial) course between two known points, and computing intermediate positions along a rhumb line or great circle connecting two points are available from standard navigational references.

These sailings have two distinct uses. One is to compute a craft's intended pre-distress motion along its intended track to create a corresponding "initial" distribution of distress locations, times and objects. The other use is for post-distress motion updates. Following their descriptions, the three sailings are compared with a view toward recommending when to use each type.

9.3.1 Mercator (Rhumb Line) Sailing

A Mercator or rhumb line sailing requires one to solve the Mercator triangle, shown in Figure 9-2 below. The Mercator triangle is shown in the normal chart orientation where north is at the top and east is to the right.

- l is the difference in the latitudes of the ends of the rhumb line segment ($l = L_2 - L_1$).
- m is the difference in the meridional parts corresponding to the latitudes ($m = M_2 - M_1$).
- p is the departure or distance of the second point to the east or west of the first point in nautical miles.
- Dlo is the difference in the longitudes of the ends of the rhumb line segment ($DLo = \lambda_2 - \lambda_1$).
- D is the length of the rhumb line segment in nautical miles.
- C is the true compass direction from the first point to the second point.

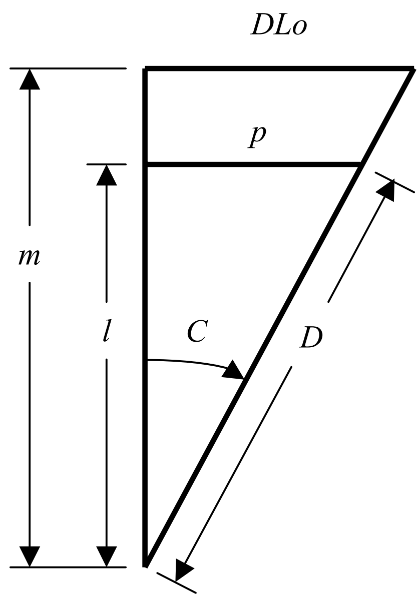


Figure 9-2: Mercator Triangle

Given a direction C and distance D, l and p may be found easily by

$$l = D\cos(C),$$

and

$$p = D\sin(C).$$

The latitude L_2 of the second point is found by simply adding l to the latitude L_1 of the first point as follows:

$$L_2 = L_1 + l\,.$$

The next problem is to find *DLo*. This is done by first finding the meridional difference m and then using the properties of similar triangles. To find m it is necessary to first compute the meridional parts M_1 and M_2 for latitudes L_1 and L_2 respectively. Bowditch [2002] uses the formula given below.

$$M = \frac{21600}{2\pi}\left(\ln\left(\tan\left(45^\circ + \frac{L}{2}\right)\right) - \left(e^2\sin(L) + \frac{e^4}{3}\sin^3(L) + \frac{e^6}{5}\sin^5(L) + \ldots\right)\right)$$

The value e is the eccentricity of the Earth, or about 0.0818188 for the WGS-84 standard. Removing all the correction terms containing e leaves the meridional parts formula for a perfect sphere. Once both M_1 and M_2 have been found, a simple subtraction produces m, to wit

$$m = M_2 - M_1\,.$$

At this point we may observe that the two triangles in Figure 9-2 are similar. This means that *DLo* and *p* are in the same ratio as *m* and *l*. Therefore,

$$DLo = \frac{mp}{l}.$$

Note that for rhumb line courses of 090T or 270T the value of *l* is zero and the above formula becomes undefined. However, these courses may be approached quite closely before shifting to another method becomes necessary.

Finally, the longitude λ_2 of the second point is found by adding *DLo* to the longitude λ_1 of the first point.

$$\lambda_2 = \lambda_1 + DLo$$

It is important to note that the equation given above for computing the meridional parts corresponding to a given latitude assumes the unit of measure for *l* and *DLo* is minutes of arc and that *p* and *D* (and also *l*) are measured in nautical miles. This means that conversions from minutes to degrees may be required to produce the final result. The meridional parts equation may be modified to work directly in decimal degrees when *p* and *D* are also measured in decimal "degrees" where distance in "degrees" equals the distance in nautical miles divided by 60.

$$M = \frac{180}{\pi}\left(\ln\left(\tan\left(45° + \frac{L}{2}\right)\right) - \left(e^2\sin(L) + \frac{e^4}{3}\sin^3(L) + \frac{e^6}{5}\sin^5(L) + \ldots\right)\right)$$

The formula can also be converted to work directly in radians when *p* and *D* are also expressed in radians where distance in radians equals distance in "degrees" times $\pi/180$, as shown below.

$$M = \ln\left(\tan\left(\frac{\pi}{4} + \frac{L}{2}\right)\right) - \left(e^2\sin(L) + \frac{e^4}{3}\sin^3(L) + \frac{e^6}{5}\sin^5(L) + \ldots\right)$$

9.3.2 Mid-Latitude Sailing

The simplest way to account for the Earth's spherical shape is with a mid-latitude sailing. The mid-latitude method very closely approximates a rhumb line, which plots as a straight line on a Mercator chart. As with the Mercator sailing described above, the latitude difference *l* and departure *p* are computed using simple plane trigonometry. The latitude L_2 of the second point is then computed as the sum of L_1 and *l*, just as with the Mercator sailing. The distance between meridians varies continuously with the secant (reciprocal cosine) of the latitude. The mid-latitude method uses the cosine of the average of the two latitudes to estimate the value of *DLo* from *p* as follows.

$$DLo = \frac{p}{\cos\left(\frac{L_1 + L_2}{2}\right)}$$

Note that this formula for *DLo* works for all rhumb line courses, even 090T and 270T, without risking a division by zero condition.

If p is in nautical miles, then *DLo* will be in minutes of longitude, if p is in decimal "degrees" defined as 1/60th of its length in nautical miles, then *DLo* will be in degrees of longitude, and if p is in radians, defined as a fraction or multiple of the Earth's radius, then *DLo* will also be in radians, defined as a fraction or multiple of the Earth's radius in the plane of the latitude L_2. Multiplying *DLo* in radians by $180/\pi$ will convert *DLo* from radians to decimal degrees of longitude.

9.3.3 Great Circle Sailing

The great circle formulas for computing a new position (L_2, λ_2) from a starting position (L_1, λ_1), initial course C, and distance D are:

$$L_2 = \sin^{-1}(\sin(L_1)\cos(D) + \cos(L_1)\sin(D)\cos(C))$$

$$\lambda_2 = \lambda_1 + \tan^{-1}\left(\frac{\sin(C)\sin(D)\cos(L_1)}{\cos(D) - \sin(L_1)\sin(L_2)}\right)$$

The functions $\sin^{-1}$ and $\tan^{-1}$ denote the inverse sine (arcsine) and inverse tangent (arctangent) functions respectively. These equations assume all quantities have been converted to radian measure.

9.3.4 Comparing the Sailings

Modeling the Earth as a perfect sphere having a circumference of exactly 21,600 nautical miles produces sufficiently accurate results for this application regardless of the sailing method used.

The great circle sailing is the most general method as it is based on the Earth's shape to a very close approximation. Great circle computations do not have difficulties at or near the poles as Mercator and mid-latitude sailings necessarily do. Over long distances between points away from the equator, great circle tracks and rhumb line tracks are considerably different but for short distances they are very nearly identical. However, the great circle computations are clearly the most computationally expensive.

Mercator sailings are less computationally expensive, especially if a perfectly spherical model of the Earth is used, removing all the correction terms for ellipticity. However, the Mercator sailing provides no advantage over the mid-latitude sailing in this case. Since a rhumb line along any course other than 090T or 270T is a loxodrome spiraling into a pole, since Mercator projections cannot represent the poles, and since they become increasingly distorted at high latitudes, Mercator sailings are unsuitable for use near the poles except over very short distances.

Mid-latitude sailings are the most computationally efficient, but suffer from all the same shortcomings near the poles as Mercator sailings. However, mid-latitude sailings may be used for courses at or near 090T or 270T. The differences between positions computed with the mid-latitude and Mercator sailings is less than 0.5% of the distance sailed, even over distances on the order of 1,000 nautical miles. This is true even when corrections for the Earth's eccentricity are included in the Mercator method. For short distances, on the order of 10 nautical miles, the differences between positions computed with great circle and mid-latitude sailings remain well under 1% of the distance moved for all latitudes less than 80 degrees.

9.3.5 Using the Sailings

The following paragraphs describe recommended uses for the sailings.

9.3.5.1 Pre-Distress Motion and Initial Distributions

Whenever the nature of the missing craft's intended route is known and it is necessary to simulate its pre-distress motion, the sailing used in the simulation should emulate the craft's intentions as closely as possible within the limits of uncertainty about those intentions and the craft's ability to carry them out accurately. In other words, the choice of sailing to use for pre-distress motion will normally be determined externally to the model based on the facts or assumptions of the case. The only real choice will be which model of the Earth to use and which "sailing" to use when choosing sample distress positions or way points relative to a given position. While the sample tracks connecting any two sample way points should comply with the type of sailing the craft intended to use on the leg connecting them, computing the sample replication positions themselves could be another matter.

It is recommended that sample positions relative to a mean position or track be computed using a great circle sailing in all cases. This will allow both large and small position uncertainties to be handled correctly. Since these "initial" positions are computed only once, computational efficiency should not be a serious issue.

9.3.5.2 Post-Distress Motion and Search Updates

While pre-distress motion has the potential for long unbroken legs, post-distress motion does not. This is especially true if a drift update cycle with a period on the order of only one hour is used. In the past, the drift update of a distribution has been the most computationally intensive part of the simulation. In the new search planning tool, the combined drift and search updating process needed to properly account for the relative motion between SRUs and search objects will clearly be the most computationally intensive portion of the model. Drift updates for periods when SRUs are not on scene will remain a close second. This means that the computational efficiency of navigational computations for drift updates, SRU movements along search legs, and CPAs between SRUs and replications in the simulation must be as efficient as possible.

It is recommended that the mid-latitude sailing method be used for all post-distress search object motion updates, even those that may be close to the poles. It is recommended that the mid-latitude sailing method be used for all simulated SRU movements when search legs are known or assumed to be along rhumb lines. If search legs in polar regions are along great circles, then points along them and CPAs with replications will have to be computed accordingly.

9.3.6 Computational Considerations

It is recommended that the coordinate system shown in Figure 9-1b be adopted for use in all computational modules whenever planar (e.g., mid-latitude) solutions to navigation problems are used that deal with computing movements, distances, bearings, directions, courses, etc. over the Earth's surface. When spherical coordinates are used directly, as in great circle sailings, it is recommended that the convention of north and east being the positive directions be retained with respect to latitudes and longitudes.

It is recommended that all units used in navigational computations be converted to radian measure prior to actual use in the computational modules of the simulation "engine." This would mean, for example, converting nautical miles to radians, knots to radians per hour, degrees of latitude and longitude to radians, courses/directions to radians, etc. These conversions should be done as either the last step before the data is handed off from an external (e.g., user input) module or as the first step when the simulation "engine" receives input from an external source. Whenever possible, data should be stored in the same coordinates and units as those used by the simulation engine, so as to reduce the number of data conversions required during a simulation run. This includes environmental data to the extent feasible.

Vector data should also be stored in rectangular component form to eliminate the need for polar-to-rectangular conversions prior to adding vectors. Leeway formulas should be expressed as a down wind component and cross wind component for similar reasons, and also because such representation has been recommended by the most recent leeway studies from the USCG Research and Development Center.

Data outputs from the simulation engine that are to be displayed to the user should, of course, be converted to the coordinates and units of measure familiar to the user. It is recommended that such conversions be either the last step before the simulation engine hands the data off to an external module, such as a GIS, or the first step when an external module receives data from the simulation engine.

Whenever positions are to be plotted on a geographic chart or map display, the plotting software should use the same chart or map datum as the underlying chart or map. Normally the GIS in use will address this issue. However, when positions based on GPS devices are received, the chart or map datum to which the GPS device is set needs to be obtained, especially if the search planner is trying to correlate the reported position with a feature on his chart display. Although not a significant problem for open waters, vessels have been known to run aground because the positions provided by the GPS device they were using and the chart on which they were plotting their positions were not both based on the same chart datum.

Conversions from true compass directions to magnetic compass directions should be available. However, there will be a continuing need to keep such conversions current with the Earth's changing magnetic variation. The Earth's magnetic poles are not fixed but actually move over time, and the rate of movement for the north magnetic pole seems to have accelerated in recent years. Aircraft SRUs normally use a magnetic compass, not a gyrocompass, to maintain their headings. A useful source for current information about geomagnetic variation is the web site http://www.ngdc.noaa.gov/seg/potfld/geomag.shtml.

Unfortunately, the computational methods provided by Bowditch [2002] are not always the most suitable for implementation on computers. For example, some formulas cannot be used for directions near 090T or 270T because they use the tangent function. The tangent function tends to infinity as these angles are approached and has discontinuities at those values. The tangent function on a computer also becomes less and less accurate as those angles are approached. There are other ways to solve the same problem that allow 090T and 270T to be approached much more closely with no appreciable loss in accuracy. The mid-latitude method, for example, is very accurate at and near 090T and 270T with no computational vices. Rather than following the procedures given in Bowditch [2002] exactly as written, advantages can often be gained by solving the same navigational problem in a different way. Such improved solutions may be reached by mathematical analysis or by consulting the source code for navigational software developed by reliable sources, such as the National Imagery and Mapping Agency (NIMA).

10 Optimal Resource Allocation

The ultimate goal of search planning is to allocate the available resources in a way that maximizes the chances of finding survivors as early in the search as possible. Since the early 1970s when CASP was first developed, considerable advances in optimal effort allocation algorithms has been made.

10.1 Resource Allocation versus Effort Allocation

At the outset, we should distinguish between optimal effort allocation and optimal resource allocation. Optimal effort allocation, also called "unconstrained" effort allocation, acknowledges only one limitation—the total amount of effort available. There are no restrictions on how that effort may be applied nor is there any limitation on how finely the effort may be divided. An optimal effort allocation might require any levels of effort, large or infinitesimally small, to be applied in several cells of the probability map simultaneously, regardless of whether those cells were contiguous. In SAR terminology, an optimal allocation of a single SRU's available effort might require it to search several widely separated cells simultaneously, using a different coverage in each. Clearly this is not operationally feasible. However, algorithms for optimal effort allocation have been found and shown to work well.

Resource allocation, on the other hand, requires the available effort to be allocated in discrete amounts subject to the constraints of operational feasibility. This generally means allocating effort over a few relatively large contiguous regions, usually rectangular in shape, within which the coverage is approximately uniform. Computing a "constrained" optimization that meets these practical requirements is far more difficult than computing an "unconstrained" optimization. However, the "unconstrained" optimization generally provides a good starting point for developing a nearly optimal search plan. At the very worst, the search planner can inspect the unconstrained coverages and approximate them with an operationally feasible plan. Search plans developed in this fashion are usually quite good and almost always better than those developed by just inspecting the probability map. However, there are ways the computer can aid the search planner to develop more nearly optimal operationally feasible search plans.

10.2 Searching for Fixed Objects

It has been shown that for fixed objects, a uniformly optimal search plan exists and such plans minimize the mean time to find the object. A plan that is uniformly optimal is one where at any point in time as the effort is expended, the partial plan up to that point is also optimal. It has been shown that uniformly optimal plans also minimize the mean time required to find the object. Koopman [1946,1980] discussed some simple optimal allocations as early as 1946. Koopman went so far as to develop optimal allocations for bivariate normal distributions of search object locations based on the inverse cube model of detection and the constraint of uniform coverage in rectangular areas. Charnes and Cooper [1958] developed an algorithm that produces uniformly optimal search plans for exponential detection, also known as "random" search. This algorithm works for any probability distribution that can be represented as a set of regions (or cells) with probabilities of containment and known or computable areas. As Stone [1989] has discussed, the Charnes-Cooper algorithm has been extended to include other detection functions such as Koopman's inverse cube model.

CASP (1.x) currently uses a variation on Koopman's original work. A probability map for a specific datum time is constructed for the primary search object. This map is then approximated using some number of bivariate normal distributions. Resources (SRUs) are then optimally allocated to rectangular areas centered on these distributions in all possible combinations. These allocations are based on the inverse cube model of visual detection. The set of SRU/rectangle pairs that produces the highest probability of success (POS) is returned as the optimal search plan. This algorithm seems to work reasonably well and is much less prone to recommending operationally unacceptable search plans than its predecessor. Although there has been no opportunity to thoroughly test it as part of this project, it appears that the algorithm is suitable for continued use, although some improvements may suggest themselves as work progresses.

Optimal search plans based on a single probability map are said to be "myopic" or near/short-sighted. Such plans maximize the POS for a single search without regard to the impact this allocation may have on future searches. For fixed objects it has been shown that myopic optimization over a series of searches produces the same POS in the end as optimally allocating all of the available effort over all searches in a single massive search, assuming that much effort could be made available all at once. This is known as the additive principle of optimal search. The additive principle of optimal search tells us that, for fixed objects, there is no POS advantage to be gained or lost by choosing between a single massive optimal search or a series of smaller optimal searches that expend the same total amount of effort in several installments. The only advantage to the single massive search option, and it is an important one, is the reduction in the time needed to locate and assist survivors such an operation would bring.

10.3 Searching for Moving Objects—Brown's Algorithm

Unfortunately, the additive principle of optimal search does not apply to moving objects. However, the myopic algorithms are still important as they are part of the solution to this difficult problem. Although the *National SAR School's* advice to "hit it hard, hit it early" by expending considerable resources on the first search is certainly sound and prudent, it does not address how to allocate that effort either myopically or in a manner that will provide maximum benefit over two or more searches should the first search fail.

Brown [1980] developed an algorithm that addresses the problem of optimally allocating effort over several searches for a moving object. The algorithm requires the ability to predict probability distributions on search object location for several, say n, searches into the future and also predict how much effort is available for each of those searches. The algorithm must also be given an initial set of search plans as a starting point. If a starting point of zero (no effort allocated to any of the n searches) is used, the first approximation to an optimal n-search plan will be a series of myopic allocations. This is because the algorithm works as follows:

For each search i, $i = 1$ to n, the cumulative effect of all other searches except search i is computed. Then the available effort for search i is reallocated to maximize the cumulative POS of all n searches. In the first iteration the only POS contributions from other searches are from prior searches. Searches subsequent to the one for which effort is being reallocated contribute nothing. Hence we obtain the result that the first allocation is myopic.

After an initial set of *n* (myopic) search plans has been established, things get more interesting. Brown's algorithm removes the effects of the first search and evaluates the remaining *n*-1 searches (2 through *n*). In a CASP context, the final Pfail value for each replication would then be copied back to its "original" location as of the datum time for the first search (the one that was eliminated). This would be done using a copy of the "original" replication file so the true original remains untouched by this process. Now the effort available for the first search is re-allocated based on the "final" Pfail values that were copied back to the original rep locations. In a very loose and simplistic sense, this is like saying, "See which replications are likely to be found and which are likely to be missed by the other *n*-1 searches, then allocate the effort in this search to maximize the chances for finding those that the other searches are likely to miss."

The next step is to eliminate the effects of the second search and evaluate the remaining *n*-1 searches (search 1, with its new allocation, and searches 3 through *n* with the original myopic allocations). The new "final" replication Pfail values after search *n* are then copied back to their locations as of the datum time for the second search and the effort available for that search is optimally re-allocated accordingly.

This process is repeated until all *n* searches have had their respective relative efforts reallocated. Note that the allocation of the n^{th} search will always be myopic since there is no information about future searches and it can only be based on the posterior distribution produced by the n-1^{st} search. Also note that each optimal reallocation for the i^{th} search proceeds as if the search object were stationary during that search.

The cumulative POS from the second set of *n* searches is compared with that of the first (myopic) set. If the difference is smaller than some small preset value, the algorithm halts and the second plan is recommended. Otherwise, the algorithm repeats itself again, starting with reallocating the effort available for the first search so as to maximize the cumulative POS over all *n* searches.

Brown has proven that the algorithm produces monotonically increasing cumulative POS values for each set of *n* searches where the effort is allocated as just described, and that the algorithm converges, usually quite quickly. The only real problem with Brown's algorithm is that the first approximation—the myopic plan—is usually quite good and not seriously sub-optimal. This is certainly true for distributions that change slowly. If motion were the only issue, then the present myopic effort allocation algorithms would be good enough except in cases where the distribution is initially constrained by a bay, estuary or strait but may reach more open waters where it can diffuse rapidly, greatly complicating the search problem if the search object is allowed to "escape" the more restricted waters undetected. However, when multiple states with state changes are added to the problem, the differences between the current algorithms and the extensions of Brown's algorithm to such cases become more pronounced.

10.4 Optimal Survivor Search with Multiple States

The optimal survivor search problem in the marine environment is to maximize the probability of finding survivors adrift while they are still alive. In this type of problem, the search object is moving and changing states stochastically. Each state may imply different motion and detectability characteristics. For example, persons in a vessel underway (State 0) might become survivors adrift in the disabled vessel (State 1) following a distress incident. They might then be forced to abandon the vessel into a life raft (State 2). The life raft might fail at some subsequent time, leaving them as persons in the water (State 3). Finally, if not found and assisted, persons in any of the above states may expire (State 4). Each of these states would have a different set of drift motion and detectability characteristics. A successful survivor search is defined as one that finds the missing person(s) alive such that a rescue may be effected. Discenza and Stone [1980] developed the necessary and sufficient conditions for optimal search plans for multistate search objects. They then extended the algorithms of Brown and Stone from finding optimal search plans for moving objects to finding optimal search plans for multistate problems where all search objects were in motion and where the set of all possible search objects in different states may occupy the same space.

The basic algorithm presented by Discenza and Stone [1980] is essentially the same as Brown's algorithm. The differences lie in how the distributions are updated over time and how the single stage effort allocations are done at each time t during the process.

Brown's algorithm recognizes only motion and search updates. The probabilities that are computed are the probabilities that the object is in cell j at time t, and has not been detected by all searching done prior to time t and will not be detected by all searching subsequent to time t with the allocations being contemplated for those (other) times. The Discenza/Stone method also recognizes state changes and any consequent changes in the drift and detectability characteristics of the simulated search objects (replications). For this method, it is necessary to compute the probability that the search object is in cell j ***and*** state k at time t, and has not been detected by all searching prior to time t and will not be detected by all searching subsequent to time t with the allocations being contemplated for those (other) times.

The other difference between Brown and Discenza/Stone is in the method for allocating the available effort at each time t. Since the actual state of the search object may be unknown, there may be several choices for effective sweep width. In other words, there is uncertainty about the sweep width of the actual search object since there is uncertainty about what the actual search object really is. Richardson and Belkin [1972] have shown how optimal search plans may be developed in the face of uncertainty about the effective sweep width, regardless of the cause of that uncertainty, although the topic was first discussed by Koopman [1956c]. Stone [1989] provides further discussion on this topic. The method works for either discrete distributions of possible sweep width values, as in the case of the search object's actual state being one of a few choices, or with continuous distributions that might be associated with continuous distributions of uncertainty about on scene environmental conditions. Stone's algorithm for optimal effort allocation in the face of uncertain sweep width is used at each stage in the basic Brown's algorithm process. In a CASP-like simulation, the distribution of sweep widths in any cell at time t may be obtained from the distributions of the different search object states in the cell at that time, together with any distributions related to other issues such as environmental parameters. An interesting technique for complying with the definition of a successful survivor search and ensuring that the effort is optimally allocated on the condition of finding the survivor(s) alive is to assign an effective sweep width of zero to search objects that have expired.

Discenza and Stone go on to describe two common practices used operationally to simplify the problem in one way or another along with a third possibility not currently in use but likely to be thought of at some point. They then compare the results of allocations based on such simplifications with the more complete solution they developed.

Each of the alternative techniques replaces the multistate problem with a proxy single state problem and finds the myopic allocation for the proxy problem. Although the search plans may differ considerably, all techniques use the same method for computing the probability that the object is in cell *j* in state *k* (a proxy state for the simplifications) at time *t*, and were not detected by all searching prior to time *t* and will not be detected by all searching subsequent to time *t*. Consequently, as Discenza and Stone point out, it is necessary to discuss only the stationary search optimization that is done at each time *t*.

The first approximation they discuss is the *selective* method where the optimal allocation for one of the possible search object states is computed and the other states are ignored. This is the method used by CASP (1.x) when the user designates a particular search object type as the "primary" object of the search.

The second approximation they call the *additive* method where a single probability distribution is computed over all states *k* for which the effective sweep width is greater than zero, ignoring the differences in sweep width by choosing a single average value instead. This method also appears to be available in CASP (1.x) when no "primary" search object is designated.

The third method (not in use) they call *weighting* computes a "weighted" version of the probability map where cell probabilities are weighted by the product of the effective sweep width for each search object state and the probability that the cell contains an object in that state.

The multistate myopic search plan was also included for purposes of comparison. This led to a total of six types of plans for the two search object states used in the comparisons. The first was a myopic selective search optimized for the first search object state. The second was a myopic selective search optimized for the second search object state. The third was a myopic additive search, the fourth was a myopic weighted search, the fifth was a myopic multistate search and the last was a *T-optimal* multistate search. A *T-optimal* search is one that maximizes the POS attained by time *T* when all effort available prior to time *T* has been expended. The Discenza/Stone method will come as close to a *T-optimal* search as desired given sufficient iterations.

The comparisons showed substantial improvements of several percentage points in POS values. These were on the order of 20% (e.g., from a myopic POS of 39% to a *T-optimal* POS of 47%) for the same levels of effort available at the same times. Increasing the chances for finding survivors alive by one-fifth just through a better allocation of the available effort is not trivial. It is quite a substantial improvement even though the *T-optimal* POS is only 8 or 9 percentage points higher than the myopic ones.

10.5 Additional Considerations

In all of the above cases except the current CASP (1.x) technique, the optimizations have been done over cellular probability maps under the assumption that the available effort was infinitely divisible into arbitrarily small pieces that could nevertheless be uniformly applied over any number of cells. Such plans are rarely operationally feasible. However, the current CASP (1.x) technique assumes that there is only one possible search object state—either a "primary" state or some fictitious "average" state. Since the distribution of search object states is likely to be non-homogeneous where one cell may have a high probability of containing objects in State 1 and a low probability of containing objects in State 2 while another cell has the opposite bias and a third cell has about an equal probability of containing objects in either state, it is difficult to see how an approximation based on a small number of bivariate normal distributions is feasible.

Abi-Zeid [2002] has made some progress in doing near-optimal allocations of effort over cellular probability maps subject to the constraints imposed on real-world facilities. The primary constraints are that each resource has an associated level of effort it can provide, only a whole number of cells forming a contiguous rectangular area may be used as a search area assignable to a resource, and the resource can only cover the assigned area uniformly. Furthermore, overlapping of search areas in the same search epoch t is prohibited (primarily for safety of flight reasons since most searching is done from aircraft). At present, the algorithms in use are very costly in computational terms, but nevertheless there may be fertile ground here for developing techniques useful to the new search planning tool. This work was done in connection with developing an automated search planning tool called *SARPlan* for planning optimal searches over land in the Canadian wilderness.

Another idea for aiding the development of optimal search plans includes computing probability maps on grids that are parallel to the preferred orientation of search areas. Usually this will be the mean direction of drift during the planned search.

Computing an optimal plan on a cellular basis and then creating an "effort map" or "effort grid" could also prove useful. This would be especially true if efficient methods could be developed for approximating that grid with uniform coverage rectangles tailored to resource capabilities. This would be somewhat like the current CASP (1.x) technique of approximating generalized probability maps with a few bivariate normal distributions.

It might also be appropriate to consider computing average cell probabilities over the duration of a candidate search plan during search epoch t, rather than taking the "snapshot" approach. That is, if SRUs were to be on scene searching for eight hours, then perhaps the optimal single stage plan for that period should be based on the average probability density in each cell over that eight-hour period, rather than an instantaneous snapshot at some arbitrary time during the eight-hour interval.

11 Search Evaluation and Relative Motion

One of the most important improvements the new search planning tool can bring to search planners in the field is a more detailed evaluation of search results. The present CASP (1.x) method assumes that either the entire search is carried out in a single instant in time and therefore applies only to the state of the probability density distribution at that instant, or that all events and motion come to a stop for the period when SRUs are on scene searching. While such techniques are suitable for solving the optimal effort allocation problem to a reasonable approximation, useful search evaluation requires a more detailed approach. The effects of motion during the search are now almost completely ignored, even though they are often quite substantial. In addition, only average POD values over areas are used. For parallel track search patterns, replications that fall directly on an SRU's track get the same POD as those halfway between tracks. The computed POD is too low for the former and is too high for the latter. CASP (1.x) cannot evaluate a track line search. All of these shortcomings could be largely removed if the new search planning tool adjusted Pfail values based on distances at the closest point of approach (CPA).

11.1 Estimating Lateral Range Curves

In 1978 the USCG Research and Development Center began performing extensive scientific detection experiments to gather data on the detection performance of Coast Guard crews manning Coast Guard platforms while searching for typical search objects under typical operational conditions. Both visual search and search with a variety of aids and electronic sensors were tested. The data from these experiments were analyzed using a LogOdds (also known as LogIt) technique. This is a standard statistical analysis technique found in various texts and statistical software packages, such as the Statistical Package for the Social Sciences (SPSS). The term "LogOdds" refers to the natural logarithm of the odds ratio. Odds ratios were used to measure the relationships between the various variables on which data were collected and the effective sweep width. This provided two results. First, it allowed the researchers to identify which variables had a significant impact on the sweep width and which did not. Second, it actually produced lateral range curves describing detection performance. In fact, the significance of a variable was determined from its impact on the lateral range curve. More specifically, it was the impact on the area under the lateral range curve that was examined since this area is, by definition the effective sweep width.

The LogOdds model is a binary, multivariate regression technique to find the best quantitative relationship among independent variables x_i and a probability of interest, such as the probability of detecting an object during one transit of a long track as a function of the object's off-track displacement (the lateral range) and the various search conditions (e.g., meteorological visibility, search object type, sea state, type of search platform, etc.). The technique is quite flexible and robust. The independent variables (x_i) may be either continuous (e.g., search speed, search altitude, time on task, etc.) or they may be discrete (e.g., day vs. night, fixed wing vs. helicopter SRU, etc.). The LogOdds technique uses the following equation:

$$POD = \frac{1}{1+e^{-\lambda}} \qquad [11.1]$$

where λ is a function of the independent variables. If λ is a linear function of the independent variables, the basic LogOdds model can be used with the following expression for λ:

$$\lambda = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 + ... + a_n x_n \qquad [11.2]$$

where the a_i are constants determined by software according to the LogOdds statistical analysis technique and the x_i are the independent variable values.

Assume for the moment that x_1 is the lateral range. Also assume that all the independent variables x_i (e.g., meteorological visibility, sea state, SRU type, search object type, etc.) except the lateral range x_1 are known and that all the coefficients a_i for that same search situation are known as well. By substituting these known values into equation [11.2] above, one could obtain an expression for λ as a function of lateral range x_1 alone. One could then substitute this expression for λ in the basic LogOdds equation [11.1] to obtain a lateral range curve for the specific search situation represented by the given set of independent variables. In other words, a specific lateral range curve could, in theory, be constructed for each specific search situation and used to evaluate that particular search.

The ability to "tailor" a lateral range curve to fit the specific search conditions actually encountered would be a powerful addition to the search planner's arsenal. Unfortunately this cannot be done using the actual coefficients from the experiments for several reasons.

- The focus of the detection experiments was to obtain sweep width values. As a result, the lateral range curves were primarily regarded as a means to an end rather than as an end in themselves.
- Although the raw data from the experiments were preserved in appendices to the R&DC reports and the sweep width results appeared in tables constructed for inclusion in the *National Search and Rescue Manual* (1991), the LogOdds coefficients were not preserved.
- The sweep width tables in the *National SAR Manual* (1991) contain over 3,000 entries for "uncorrected" visual sweep width values. In addition, there are a table of four weather correction factors, a fatigue correction factor, and a table of search speed correction factors for aircraft that contain 46 entries with values other than 1.0 (no correction). This represents over a million possible combinations. By comparison, only a few experiments were done for which the a_i coefficients were actually estimated by the LogOdds model. The values in the sweep width tables were then interpolated or extrapolated from a relative handful of experimentally determined sweep widths with the aid of a computer program called *VSW* (Visual Sweep Width). For the vast majority of entries in the sweep width tables, no LogOdds coefficients ever existed.

However, there may be a way to work the problem "backwards" heuristically to obtain approximate "tailor made" lateral range curves from the tabulated sweep width values. For a specific search situation where all the values except the lateral range are known, the LogOdds equation of the lateral range curve can be simplified to a version that involves only two coefficients. The simplified version is

$$POD(x_1) = \frac{1}{1 + e^{-(a_0' + a_1|x_1|)}} \qquad [11.3]$$

where $a'_0 = a_0 + a_2x_2 + a_3x_3 + \ldots + a_nx_n$. The absolute value bars on x_1 produce symmetry to the right and left of the sensor's track.

Figure 11-1 shows two lateral range curves produced by the LogOdds model, along with that produced by Koopman's [1946, 1980] hypothetical inverse cube model of visual detection for reference. The two LogOdds curves shown below are similar to those shown in Figure 3-8 of Edwards, et al [1978].

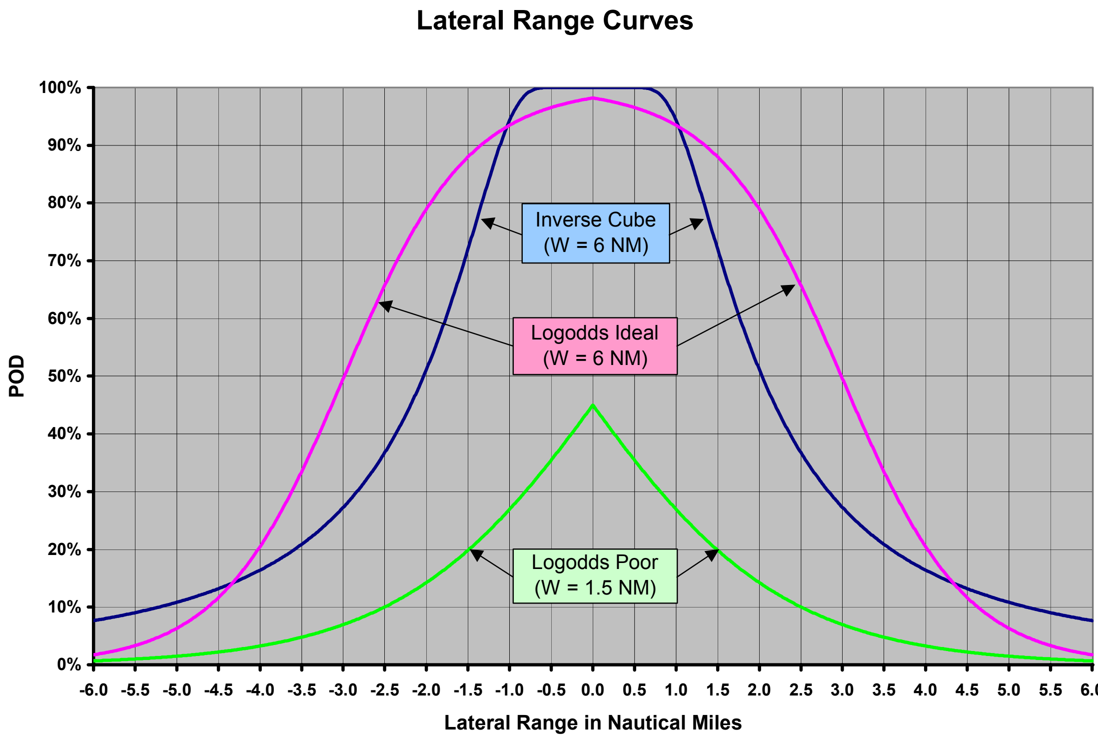


Figure 11-1: Three Lateral Range Curves

11.1.1 Comparing LogOdds to Inverse Cube

Although the inverse cube lateral range curve and that of LogOdds for ideal conditions are clearly different, they are both highly peaked with maximum on-track POD values at or near 100%. In a parallel path search pattern, both curves would produce similar POD vs. Coverage curves. The equation for the inverse cube lateral range curve is given by

$$POD(x) = 1 - e^{-\frac{W^2}{4\pi x^2}}. \qquad [11.4]$$

Comparing equations [11.3] and [11.4] we see that whereas the LogOdds model allows a family of lateral range curves that can vary in both maximum height and shape, the inverse cube model always has the same maximum height (100%) and same basic shape. The inverse cube lateral range curve can only grow broader as the sweep width W increases or more narrow and more sharply spiked as the sweep width W decreases. This would also be true of the LogOdds model if a'_0 were fixed and only a_1 could be changed to reflect changes in sweep width. However, both coefficients of the simplified LogOdds model may vary, producing an infinitely large family of lateral range curves with different shapes and maximum heights. The allowed variation extends from highly peaked inverse-cube-like lateral range curves to curves that are quite low and relatively flat in the sense that the central peak is only a small fraction of the maximum possible height of 100%.

One characteristic of the inverse cube model is that its family of "broad" to "narrow" lateral range curves all produce exactly the same POD vs. Coverage curve for estimating the average POD over an area covered by a pattern of perfectly straight equally spaced parallel search legs. This is not true of the LogOdds family of lateral range curves. The "LogOdds Poor" curve in Figure 11-1 will produce a substantially different, and lower, POD vs. Coverage curve than those produced by either the inverse cube model or the "LogOdds Ideal" lateral range curve. In terms of our simulation, two lateral range curves with shapes as different as the two LogOdds curves shown in Figure 11-1 would produce substantially different Pfail adjustments resulting in quite different post-search probability maps, even if the sweep widths were identical in both cases.

11.1.2 Analyzing the Simplified LogOdds Model

In order to "tailor" the simplified LogOdds lateral range curve to fit the search conditions, we need a way to estimate a'_0 and a_1. It is clear from equation [11.3] that if the POD at zero lateral range (POD_0) were known, then a'_0 would be given by

$$a'_0 = -\ln\left(\frac{1}{POD_0} - 1\right). \qquad [11.5]$$

Then, if W were known, we could use the fact that

$$W = \int_{-\infty}^{\infty} \frac{1}{1+e^{-\left(a'_0 + a_1|x_1|\right)}}\, dx_1 = \frac{2}{a_1}\left[\ln\left(\frac{e^{-a'_0}}{1+e^{-a'_0}}\right)\right] \qquad [11.6]$$

to solve for a_1 in terms of a'_0 as follows:

$$a_1 = \frac{2}{W}\left[\ln\left(\frac{e^{-a'_0}}{1+e^{-a'_0}}\right)\right]. \qquad [11.7]$$

Unfortunately only values for W have been tabulated leaving one equation [11.6] in the two unknowns a'_0 and a_1. There are no corresponding tables of maximum (on-track) POD values where we could look up the POD_0 value for each tabulated W value. If we are to simulate the range of lateral range curve shapes that the USCG Research and Development Center reports indicate are possible, then we must develop some reasonable heuristic for estimating POD_0.

Perhaps the best way to judge the relative "level" of the search conditions in a given situation is to note where the corrected sweep width falls in relation to the maximum possible sweep width. It seems reasonable to assume that the maximum sweep width in any given group of sweep widths representing a given a given search object type and a given SRU type, search altitude, and search speed, would have a lateral range curve similar to that of the "LogOdds Ideal" curve shown in Figure 11-1. By assuming a high POD_0 value for use under "ideal" environmental conditions when W is at its maximum, we can compute both a'_0 and a_1 from the assumed POD_0 value and the tabulated W_{max} value. Then, if we can develop a plausible heuristic describing how the value of POD_0 falls as environmental conditions deteriorate and the value of W also falls, we will have a method for generating a plausible family of lateral range curves.

To give an example for purely illustrative purposes: Consider a fixed wing aircraft searching for a twenty-foot power boat from an altitude of 1,000 feet while flying at a search speed of 150 knots. The maximum sweep width in this case is 5.5 nautical miles (5.0 uncorrected W for 30 NM visibility × 1.1 correction factor for 150 knot search speed). Suppose the actual visibility is 15 NM, the winds are between 15 and 25 knots, and the seas are 2-3 feet. The corrected sweep width is computed to be 2.42 NM (4.4 × 0.5 × 1.1).

For the "ideal" curve, let us assume a maximum POD at zero lateral range of about 98.2% for ideal environmental conditions. (This would give a'_0 a value of about 4.0, the value used for the "LogOdds Ideal" curve in Figure 11-1. The value of a_1 for a 5.5 NM sweep width would be about –1.46, making the resulting curve slightly narrower than the one in Figure 11-1, which has a slightly larger sweep width.) Now let us assume that $POD_0(W)$ is related to $POD_0(W_{max})$ according to the following equation:

$$POD_0(W) = POD_0(W_{\max})\sqrt{\frac{W}{W_{\max}}} \qquad [11.8]$$

Substituting our values ($POD_0(W_{max})$ = 98.2%, W_{max} = 5.5 NM, W = 2.42 NM) we get a value of about 65.1% for a maximum POD at a lateral range of zero. We now use equation [11.5] to get a value of about 0.625for a'_0. We then compute a value of about –0.871 for a_1 using equation [11.7]. Hence the desired equation for the lateral range curve is given by

$$POD(x_1) = \frac{1}{1 + e^{-(0.625 - 0.871|x_1|)}} \qquad [11.9]$$

for x_1 in nautical miles. This curve is shown in Figure 11-2 below while Figure 11-3 illustrates the family of curves for which this particular solution is one member.

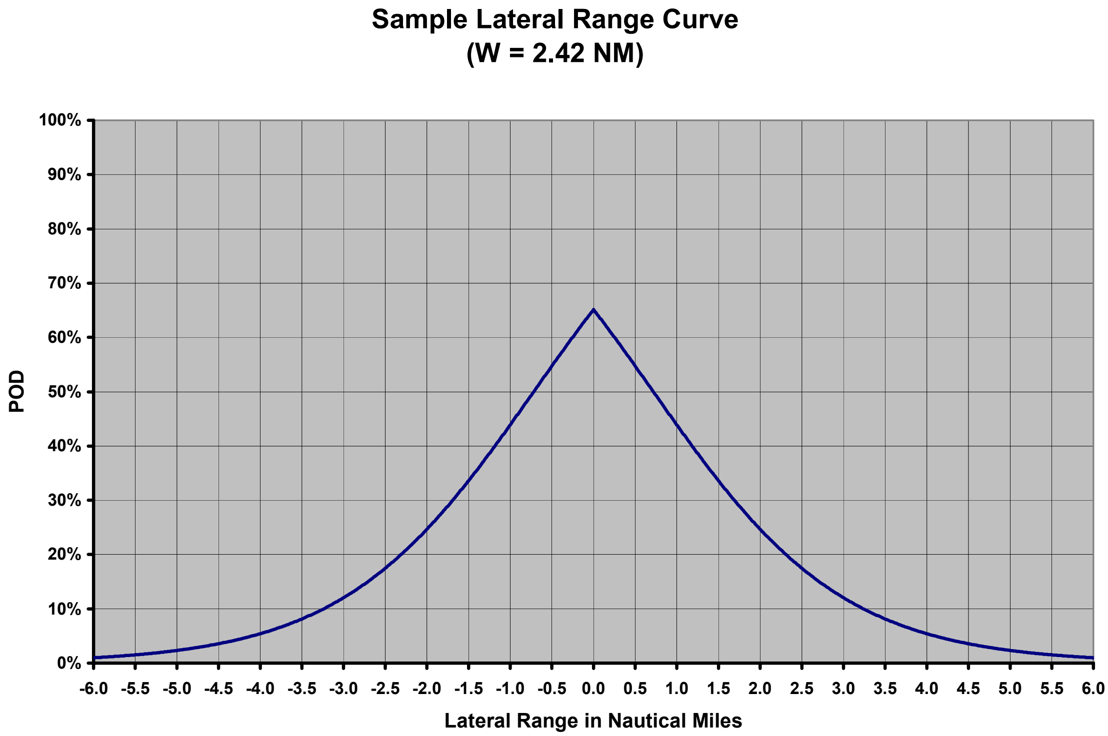


Figure 11-2: A Sample LogOdds Lateral Range Curve

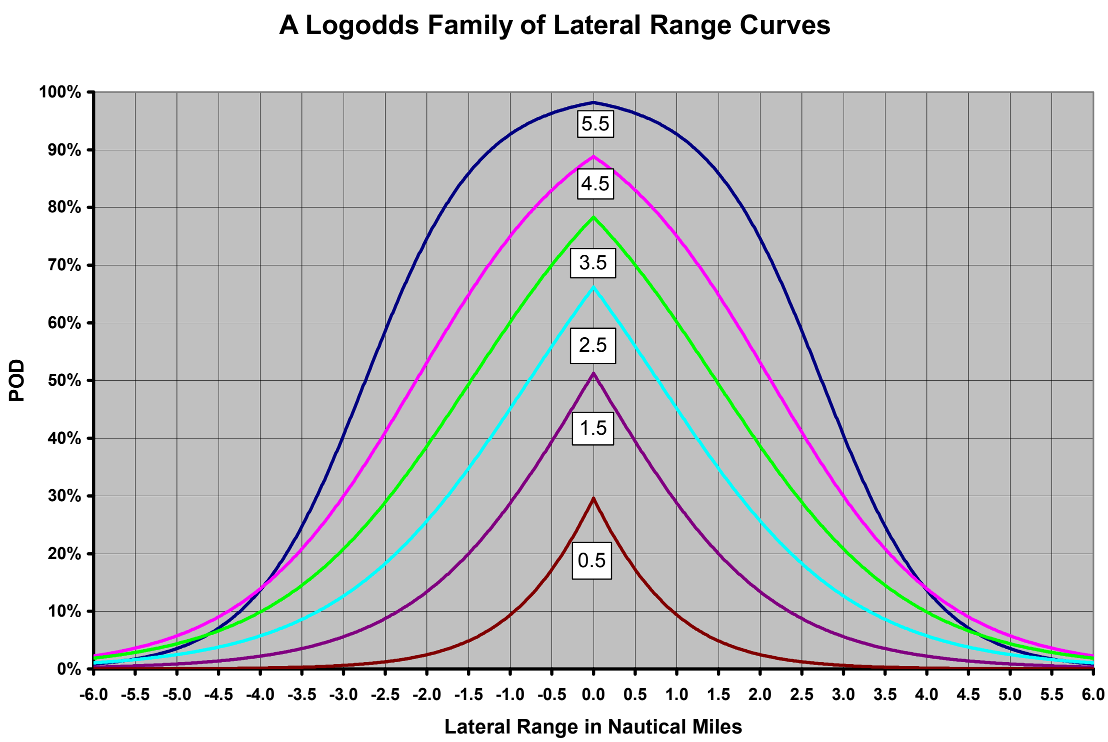


Figure 11-3: A Family of LogOdds Lateral Range Curves

Figure 11-3 above illustrates the progression of lateral range curves that would result from using equation [11.8] to estimate POD_0 for a fixed wing aircraft searching for a 20-foot boat from an altitude of 1,000 feet while flying at 150 knots as the environmental conditions deteriorated with consequent reductions in sweep width. The values shown below the peak of each curve represent the sweep width of that curve. Other search situations would produce similar families of curves. These curves would in turn produce families of POD versus Coverage curves ranging from curves similar to the traditional "single search" curve based on Koopman's inverse cube model down to the so-called "random search" curve, also developed by Koopman. The shaded region in Figure 11-4 below shows the range of possible POD versus Coverage curves, assuming all detection functions are regular and that coverage of the search area is uniform.

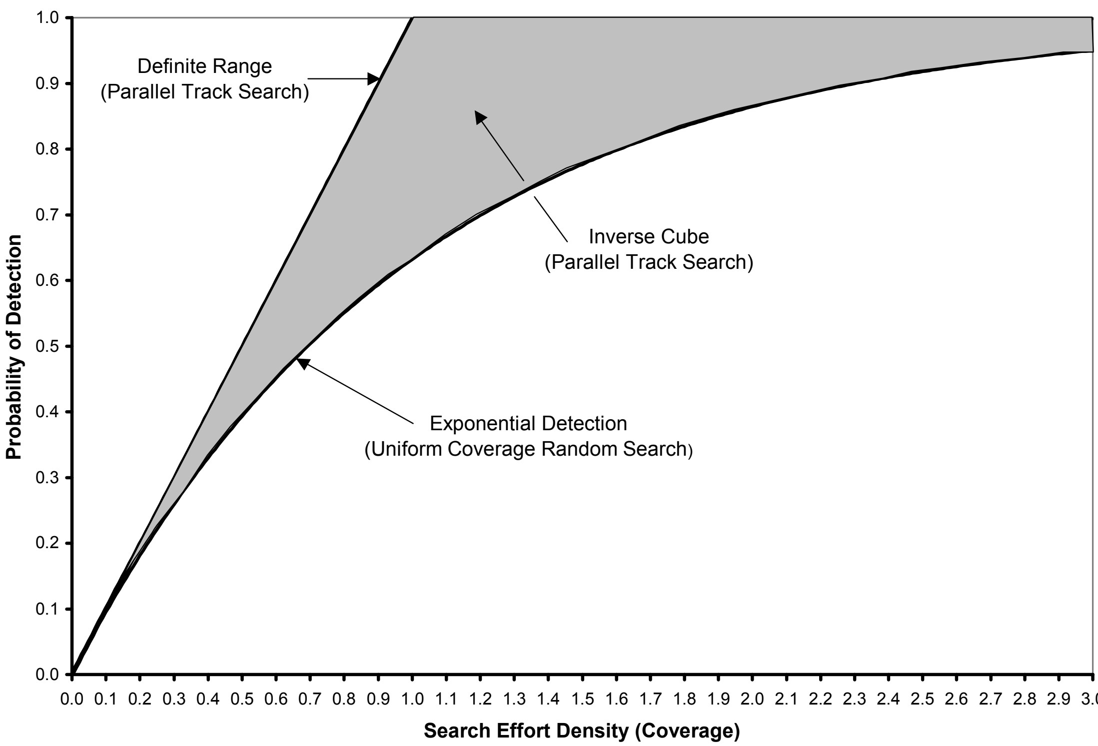


Figure 11-4: POD versus Coverage Curves

The heuristic given in equation [11.8] is not intended as a recommendation but is used here only for purposes of illustration. The square root of the ratio of W to W_{max} was chosen only because it equals 1.0 when $W = W_{max}$ and decreases more slowly as W starts moving away from W_{max} then more rapidly later as environmental conditions become worse and worse for searching. There are infinitely many possible heuristic functions although all useful ones should probably have similar general characteristics.

It is important to understand that there will be no "perfect" solution to the problem of how to estimate the most appropriate lateral range curve for each search situation, even if all the LogOdds coefficients from the actual experiments were available. Although the progression shown in Figure 11-3 seems plausible, it is possible to find some potential inconsistencies between those curves and the real-world circumstances they represent. For example, a decrease in sweep width from 5.5 NM under ideal conditions to 4.5 NM could be due solely to a reduction in meteorological visibility from 30 NM to 12 NM. This probably would not significantly affect the maximum on-track POD in the real world, but using equation [11.8] would cause a significant reduction of POD_0 in our model. In a similar vein, the assumption of a high on-track POD near 100% under ideal conditions may not be realistic when the search object is one of the smallest types, such as a PIW or small raft, and the SRU is an aircraft. Finally, note in Figure 11-3 that the tails of the "ideal" conditions curve (W = 5.5 NM) fall below the other curves starting just a little less than 4.0 NM from the SRU's track. This does not seem very reasonable. However, in all fairness, the POD values at such ranges in this case are so low that no significant error in the computations would result from this apparent anomaly.

The goal of the discussion above was to lay the groundwork for developing a more realistic detection model than the present one where LogOdds-derived sweep widths are used with the inverse-cube-derived POD versus Coverage curve. The Research and Development Center's detection experiments showed that Koopman's model was not always realistic and that lateral range curves do change as conditions deteriorate in ways that Koopman's model cannot accommodate. The new search planning tool needs to reflect this knowledge to the extent feasible.

11.2 Adjusting Pfail Values from Lateral Range Curves

Once the coefficients for an equation of the lateral range curve for a particular SRU on a particular search have been estimated, Pfail adjustment based on distance at CPA is a relatively simple matter away from the ends of search legs. Each leg is treated as a statistically independent detection opportunity. The first step in evaluating a search leg's impact on a replication's Pfail is to compute the distance between the SRU and the replication at CPA. This distance would then be substituted for x_1 in the appropriate lateral range equation, such as equation [11.9], to compute the POD. The replication's Pfail value would then be adjusted according to

$$Pfail_{new} = Pfail_{old}\left(1 - POD\left(x_1\right)\right) \qquad [11.10]$$

If deemed to be significant, SRU navigational error can also be accommodated by assuming a normal distribution of search leg placement errors perpendicular to the leg's direction. The result would be a distribution of possible search tracks that were all straight and parallel, but not equally spaced, the distribution of spacings being normal as well. One way to handle this approximation of the effects of SRU navigational error is to convolute the lateral range function with the normal function representing the SRU's distribution of search track placement errors. The result will be an expected or average lateral range curve that is lower and flatter than the basic lateral range curve. This adjusted lateral range curve could then be used for Pfail updates.

11.3 Computing Closest Point of Approach (CPA) Distances

The basic method for computing the distance between two objects each moving with a constant course and speed is to first solve for the time of CPA, then compute the position of each object at that time, and then compute the distance between the two positions. To solve the CPA problem, it is first necessary to establish the positions of the two objects at some initial time $t = t_0$. Let the initial position vector of the replication be given by $\boldsymbol{R_0}$ and the initial position vector of the SRU be given by $\boldsymbol{S_0}$. Let the velocity vector of the replication be given by $\boldsymbol{U}$ and velocity vector of the SRU be given by $\boldsymbol{V}$. Then the time of CPA, t_{CPA}, is given by

$$t_{CPA} = \frac{-\left(\vec{R}_0 - \vec{S}_0\right) \bullet \left(\vec{U} - \vec{V}\right)}{\left|\vec{U} - \vec{V}\right|^2}$$

where the symbol • in the numerator denotes the dot, or scalar, product of the two vector quantities on either side and the vertical bars in the denominator denote the magnitude of the vector quantity they contain. Note that when dealing with a search leg of finite length, the computed CPA time may be prior to the leg's start time or later than the leg's end time. The distance between the replication and the SRU at CPA, d_{CPA}, is then given by

$$d_{CPA} = \left|\vec{R}_{CPA} - \vec{S}_{CPA}\right|$$

where $\boldsymbol{R_{\mathrm{CPA}}}$ is the position vector of the replication at the time of CPA and $\boldsymbol{S_{\mathrm{CPA}}}$ is the position vector of the SRU at the time of CPA. When vector quantities are expressed and stored in terms of their orthogonal (x, y) components, the above computations are quite straightforward and efficient. However, there are some important details to consider with respect to waypoints in the search pattern used by the SRU.

The SRU may be thought of as having a "detection envelope" of some description surrounding it as shown in Figure 11-5 below for an assumed omni-directional and perfectly symmetric sensor.

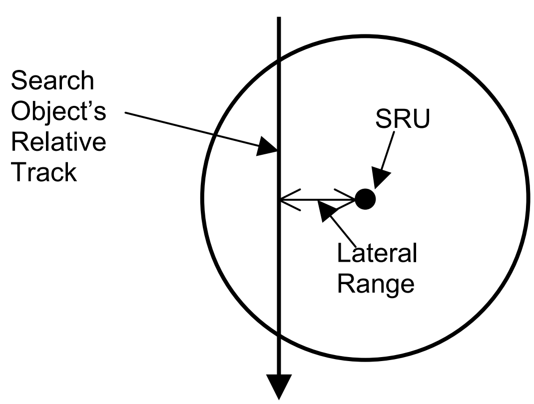


Figure 11-5: Complete Penetration of Detection Envelope
POD = POD(Lateral Range)

The lateral range curve value at a given lateral range represents the integral of the instantaneous detection function along a search object's path relative to the SRU that traverses the entire detection envelope and passes the SRU at the given lateral range at CPA. Therefore, the lateral range curve should be applied only to those replications that have fully penetrated the SRU's detection envelope from one side to the other (or front to back relative to the SRU) along a straight relative track. If all the search legs were infinitely long, then using the lateral range curve for any segments of them would be entirely correct.

Unfortunately, real-world search patterns have legs of finite length that are connected in a continuous pattern, although the traditional POD vs. Coverage curve does not acknowledge this fact. Figure 11-6 below shows how a search object's relative track might appear to the SRU near the ends of two search legs connected by a cross leg in a parallel track search pattern.

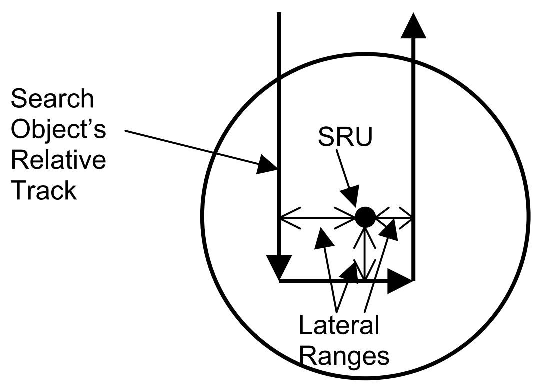


Figure 11-6: Incomplete Penetrations of Detection Envelope

This raises the question of how to deal with replications that do not fully penetrate the SRU's detection envelope as well as how to define the extent of the detection envelope. In this case, a reasonable but slightly optimistic approximation could be made by ignoring the cross leg and assuming the search legs extended from one edge of the search area to the other, rather than beginning or ending one-half track space inside the search area. Then, by assuming that all replications with computed CPA times between the start and end times of each leg completely penetrated the detection envelope regardless of whether they actually did, and assuming that all search legs are statistically independent of one another as far as POD evaluation is concerned, the replication's Pfail could be adjusted based on the CPAs with each SRU on each search leg. Ignoring the cross leg is reasonable in most cases, especially for aircraft SRU's that must bank into turns and quickly set up on the next leg in the opposite direction. Between the aircraft's attitude and the navigational preoccupations of the crew, not much searching is done on cross legs. If the inverse cube lateral range curve were used, the technique just described would result in an average POD over the search area that was very close to the "traditional" value.

Unfortunately, the technique just described would not properly handle expanding square (SS), sector (VS) or track line (TSN, TSR, etc.) search patterns where every leg (except possibly short cross legs at the ends of track line patterns) must be properly evaluated and the assumption of statistical independence of search legs connected at a turn point is not really justified. The problem with evaluating these patterns is how to avoid either "double counting" or not counting detection opportunities for replications near the SRU as it approaches, executes, and departs a turn point in its search pattern. For an expanding square (SS), all turns are 90 degrees (usually to the right) and the search legs get longer and longer. For a sector search (VS), all turns are 120 degrees (usually to the right) and the "cross legs" are one half the length of the "search legs." Track line searches may contain turns varying from slight "dog legs" up to 180 degrees and legs may be of any length.

Another equivalent problem concerns "turn points" in the individual replication drift trajectories. If small time steps, e.g., one hour, are used for drift updates, then each replication's direction and rate of motion will be subject to change every hour. A change in the replication's drift would have an effect on the relative track similar to a change in SRU course and/or speed. Therefore multiple CPA computations may be required to determine when CPAs were reached.

Finally, there is the issue of replication state transitions during a search. Usually such transitions will affect both the replication's motion and the ability of SRUs to detect it. Again, additional study will be needed to determine the best way to handle such situations. In the meanwhile, a reasonable approximation would be to adjust the replication's Pfail based on its state at the time of CPA. The effect would be as if the replication representing the first state simply disappeared at the transition time to be replaced in the same position by a replication in the second state but still carrying the same Pfail value. That is, the Pfail would be updated based on the CPA while the replication was still in the first state, then carried forward and updated again based on the CPA while it was in the second state. Rarely the two CPAs might coincide, but this is unlikely to cause serious problems.

The ideal situation would be to find a single method that could be applied to all types of search patterns and produce sensible results. It is anticipated that some additional study will be required to determine the best general approach to the issues outlined above. One way to address all of them is to develop and use "instantaneous" detection functions in true (as opposed to lateral) range and use a very fine time step, on the order of seconds, for search updates. However, even given the high speeds and large memory capacities of today's computers, this may not be feasible. Besides, there are no instantaneous detection functions for USCG sensors or visual search that are ready for implementation so some adaptation of the lateral range approach will be needed, at least initially.

The best approach to the above issues is to view the simulation of SRU/search object interactions from an event chain perspective, similar to the GPSS event chain approach described earlier. If a replication-centric approach were taken, then the event chains would be of a manageable size as each replication in turn moved through the search areas. Events would include the start and end of each search leg (which would not change), drift updates, entry into the SRU's detection envelope, departure from the SRU's detection envelope, and CPAs. This will also provide a framework for resolving Pfail update issues near SRU turn points.

12 Implementation Approach

This report has shown that developing a search planning tool that meets the Coast Guard's requirements is a complex undertaking. In order to properly manage and minimize the risks involved and assure the resulting software will provide the required operational capabilities in the probable operating environment and be highly reliable, robust, technically correct, user-friendly, portable and easy to maintain, an appropriate project plan should be developed. Although the following paragraphs do not constitute a complete list of issues such a project plan should address, it does include some major points that are too often overlooked in software development projects.

12.1 Develop a Design Framework

Before any attempt at actual software development is attempted, a general design framework should be established for the application that is robust enough to accommodate all the desirable features. This framework would serve as a road map for guiding both the initial development effort and later addition of features and capabilities not initially included as well as future enhancements. One of CASP's greatest shortcomings has been a lack of vision in its design that has discouraged a program of continuous, stepwise refinement and improvement. Such programs have proven their worth time and again across the software industry. Applications that are being continuously improved have longer life cycles, lower life cycle costs, lower corrective maintenance costs and much higher levels of reliability—an important issue for a mission-critical application like the Coast Guard's primary search planning tool.

The contributing factors to this desirable situation are many and varied, not the least of which is keeping a cadre of individuals intimately familiar with the software. In the past, CASP has suffered long periods of neglect when no one on the staff responsible for software development and maintenance in the host facility had any significant knowledge of the application. In fact, the situation today is nearly at that point. This should never be allowed to happen with any mission-critical application. The key to proper support of any application is a technical staff that understands both the mission and the people the software is supposed to support, along with any germane technical areas (and a search planning tool will have many technical areas to master). Having programmers competent in the appropriate programming language is a necessary, but relatively small and by no means sufficient condition for proper application support.

12.2 Minimal Useful Subset

Once the general framework has been established it is necessary to choose, with great care, a minimal useful subset of features and capabilities for initial implementation. This is especially true for an application with the high potential for complexity that a simulation-based search planning support tool has. It would be a grave mistake to attempt a "complete" implementation in a single installment.

An example of a minimal useful subset of the features discussed in this report might be the following:

1) A single simple but common scenario type, such as distress position and time approximately known. Multiple such scenarios would be allowed, as would weighting them.
2) Initially tagging all replications as "underway" or "not-in-distress."
3) Transition from a not-in-distress state to some (small) subset of the search object types (states) listed in the latest leeway taxonomy, within the position and time uncertainties given in the scenario definition. The entire taxonomy would be implemented.
4) Post-distress motion (drift) modeling using the existing environmental database(s) with improved algorithms (latest leeway formulas, interpretation of environmental data as "point" data with interpolation within (x, y, t) cells of the environmental data grids,).
5) A simple, possibly arbitrary, method for effecting post-distress state transitions and/or the Canadian Cold Weather Survival Model. Initially, the primary function will be to provide distributions with varying replication states as well as positions so the effort allocation algorithm(s) below may be tested.
6) SRU characteristics for multiple SRUs and types for each of several search epochs.
7) Complete implementation of sweep width tables.
8) *T-optimal* effort allocations using the method outlined by Discenza and Stone [1980] with corresponding near-optimal resource allocations. Existing myopic allocation methods would serve as a backup.
9) Search updates using CPAs, LogOdds lateral range curves, heuristic for a'_0.
10) All necessary supporting data entry and data display support.

The above subset is just an arbitrary example for purposes of illustration. Note that it does not include area or track scenario types, pre-distress motion modeling, or hazards. This does ***not*** mean that the initial implementation should totally ignore these issues and take whatever shortcuts not implementing them right away might allow. Since we have established a framework for the application, we know these things are coming. Therefore, any initial development must acknowledge this fact by putting the necessary "hooks" in place, leaving sufficient room in pertinent data structures, and generally avoiding any constructs that would prevent later implementation of the different scenario types, pre-distress motion modeling, hazards, or other features that may not have even been considered for implementation up to this point.

Also note that while this might be the first "useful" implementation developed, it might yet be premature for release to users. It might be more appropriate to add some of the "missing" features, at least in their most basic forms prior to the first field release.

Bibliography

Appendix A —Search Theory References

Books

Haley, K. D. and L. D. Stone (1980), *Search Theory and Applications*, Plenum, New York.

Koopman, B. O. (1946), *Search and Screening*, OEG Report No. 56, The Summary Reports Group of the Columbia University Division of War Research. (Available from the Center for Naval Analyses.)

Koopman, B. O. (1980), *Search and Screening: General Principles with Historical Applications*, Pergamon, New York.

Sternhell and A. Thorndyke (1946), *Antisubmarine Warfare in World War II*, OEG Report 51, Center for Naval Analyses, Alexandria, VA.

Stone, L. D. (1989), *Theory of Optimal Search*, 2nd ed., Operations Research Society of America, Arlington, VA.

Washburn, A. R. (1996), *Search and Detection*, 3rd ed., Institute for Operations Research and Management Science, Linthicum, MD.

Papers

Abi-Zeid, I. and J. Frost, (2002), *SARPlan*: A Decision Support System for Canadian Search and Rescue Operations, submitted to the European Journal of Operations Research.

Arnold, R. D. and J. Bram (1962), "Estimating the Lateral Range Curve from Observed Detection Ranges," IRM-27, Center for Naval Analyses, Washington, DC.

Bell, David E. (1982), "Regret in Decision Making under Uncertainty," *Operations Research* **30**, 961-981.

Benkoski, S. (1978), "Optimal Use of the Operational ASW Search Information System," Commander Patrol Wings Atlantic Fleet Lessons Learned, 261-279.

Benkoski, S. J., M. G. Monticino, and J. R. Weisinger (1991), "A Survey of the Search Theory Literature," *Naval Research Logistics* **38**, 469-494.

Brown, S. S. (1980), "Optimal Search for a Moving Target in Discrete Time and Space," *Operations Research* **28**, 1275-1289.

Charnes, A. and Cooper, W. W., The Theory of Search: Optimum Distribution of Search Effort. *Management Science* **5**, 44–50 (1958).

De Guenin, J. (1961), "Optimum Distribution of Effort: An Extension of the Koopman Basic Theory," *Operations Research*, **9**, 1-7.

Discenza, J. H. and L. D. Stone (1981), "Optimal Survivor Search with Multiple States," *Operations Research*, **29**, 309-323.

Dobbie, J. M. (1963), "Search Theory: A Sequential Approach," *Naval Research Logistics Quarterly*, **10**, 323-334.

Eagle, J. N. (1984) "The Optimal Search for a Moving Target When the Search Path Is Constrained," *Operations Research* **32**, 1107-1115.

Eagle, J. N., and J. R. Yee (1990), "An Optimal Branch-and-Bound Procedure for the Constrained Path, Moving Target Search Problem," *Operations Research* **38**, 110-114.

Everett, H. (1963), "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," *Operations Research* **11**, 399-417.

Forrest, R. N. (1987, rev. 1990), "Notes on Search, Detection and Localization Modeling," Naval Postgraduate School Report NPS71-87-001.

Hellman, O. (1972), "On the Optimal Search for a Randomly Moving Target," *SIAM Journal of Applied Mathematics* **22**, 545-552.

Iida, K. (1993), "Inverse N^{th} Power Detection Law for Washburn's Lateral Range Curve," *Journal of the Operational Research Society of Japan* **36**, 90-101.

Kalbaugh, D. V. (1992), "Optimal Search Among False Contacts," *SIAM Journal of Applied Mathematics* **52**, 1722-1750.

Kalbaugh, D. (1993), "Optimal Search Density for a Stationary Target Among Stationary False Targets," *Operations Research* **41**, 310-318.

Keeney, Ralph L. (1982), "Decision Analysis: An Overview," *Operations Research* **30**, 803-838.

Koopman, B. O. (1953), "The Optimum Distribution of Effort," *Operations Research; The Journal of the Operations Research Society of America*, **1**, 52-63.

Koopman, B. O. (1956a), "The Theory of Search. Part I. Kinematic Bases," *Operations Research* **4**, 324-346.

Koopman, B. O. (1956b), "Fallacies in Operations Research," *Operations Research* **4**, 422-430.

Koopman, B. O. (1956c), "The Theory of Search. Part II. Target Detection," *Operations Research* **4**, 503-531.

Koopman, B. O. (1957), "The Theory of Search. Part III. The Optimum Distribution of Searching Effort," *Operations Research* **5**, 613-626.

Koopman, B.O. (1967), "Search and Information Theory," Office of Naval Research, Code 462, Contract N00014-66-C0115.

Koopman, B. O. (1979), "An Operational Critique of Detection Laws," *Operations Research* **27**, No. 1.

Koopman, B. O. (1979), "Search and Its Optimization," *American Mathematical Monthly* **86**, 527-540.

Koopman, B. O., (1986), "An Empirical Formula for Visual Search," *Operations Research*, **34**, pp. 377-383.

Lukka, M. (1977), "On the Optimal Searching Track for a Moving Target," *SIAM Journal of Applied Mathematics* **32**, 126-132.

Mangel, M. (1981), "Search for a Randomly Moving Object," *SIAM Journal of Applied Mathematics* **40**, 327-338.

Mangel, M. (1982), "Probability of Success in the Search for a Moving Target," *Operations Research*, **30**, 212-222.

Mangel, M. (1982), "Applied Mathematicians and Naval Operators," *Society for Industrial and Applied Mathematics (SIAM) Review*, **24**, 3, pp. 289-300 (July 1982).

Morse, P.M. (1982), "Bernard Osgood Koopman, 1900-1981," *Operations Research*, **30**, pp. 416-427.

Ohsumi, A. (1986), "Optimal Searching for a Markovian Target and Relation to Optimal Stochastic Control," In *Theory and Applications of Nonlinear Control Systems* (C. I. Byrnes and A. Lindquist, Eds), Elsevier, Amsterdam, pp. 569-582.

Reber, R. K. (1956), "A Theoretical Evaluation of Various Search/Salvage Procedures for Use with Narrow Path Locators, Part I, Area and Channel Searching," Bureau of Ships, Minesweeping Branch Technical Report 117 (AD 881408)

Richardson, H. R. and L. D. Stone (1971), "Operations Analysis During the Underwater Search for *Scorpion*," *Naval Research Logistics Quarterly* **18**, 141-157.

Richardson, H. R., and Barry Belkin (1972), "Optimal Search with Uncertain Sweep Width," *Operations Research* **20**, 764-784.

Richardson, H. R. and J. Discenza (1980), "The United States Coast Guard Computer-Assisted Search Planning System (CASP)," *Naval Research Logistics* **27**, 659-680.

Richardson, W. H. (1968), "Empirical Sweep Width Analysis (Air to Surface)," Scripps Institute of Oceanography, San Diego, CA (SIO Ref 68-30).

Shanthikumar, J. G. and R. G. Sargent (1983), "A Unifying View of Hybrid Simulation/Analytic Models and Modeling," *Operations Research* **31**, 1030-1052.

Stewart, T. J. (1979), "Search for a Moving Target When Searcher Motion is Restricted," *Comput. Operations Research* **6**, 129-140.

Stone, L. D. (1972), "Incremental Approximation of Optimal Allocations," *Naval Research Logistics Quarterly*, **19**, 111-122.

Stone, L. D. (1973), "Total Optimality of Incrementally Optimal Allocations," *Naval Research Logistics Quarterly*, **20**, 419-430.

Stone, L.D. (1977). Search for Targets with Generalized Conditionally Deterministic Motion. *SIAM Journal of Applied Mathematics*, **33**, 456-468.

Stone, L. D. (1979), "Necessary and Sufficient Conditions for Optimal Search Plans for Moving Targets," *Mathematics of Operations Research* **4**, 431-440.

Stone, L. D. and J. B. Kadane (1981), "Optimal Whereabouts Search for a Moving Target," *Operations Research* **29**, 1154-1166.

Stone, L. D. (1983), "The Process of Search Planning: Current Approaches and Continuing Problems," *Operations Research* **31**, 207-233.

Stone, L. D. (1992), "Search for the *SS Central America:* Mathematical Treasure Hunting," *Interfaces* **22**, 32-54.

Stromquist, W. R. and L. D. Stone (1981), "Constrained Optimization of Functionals with Search Theory Applications," *Mathematics of Operations Research* **6**, 518-529.

Thomas, L.C, and P.B. Hulme (1997). "Searching for Targets Who Want to be Found," *Journal of the Operational Research Society*, **48**, 44-50.

Tierney, L. and J. B. Kadane (1983), "Surveillance Search for a Moving Target," *Operations Research* **31**, 720-738.

Trummel, K. E. and J. R. Weisinger (1986), "The Complexity of the Optimal Searcher Path Problem," *Operations Research* **34**, 324-327.

Wagner, D. (1989), "Naval Tactical Decision Aids," NPSOR-90-01, Naval Postgraduate School, Monterey, CA 93943, NTIS # ADA 330401.

Washburn, A. R. (1980), "Expanding Area Search Experiments," Naval Postgraduate School Report NPS55080-017.

Washburn, A.R. (1980), “On a Search for a Moving Target,” *Naval Research Logistics Quarterly* **27**, 315-321.

Washburn, Alan R. (1980), “Search-Evasion Game in a Fixed Region,” *Operations Research* **28**, 1290-1298.

Washburn, Alan R. (1981), “Note on Constrained Maximization of a Sum,” *Operations Research* **29**, 411-414.

Washburn, A. R. (1981), “An Upper Bound Useful in Optimizing Search for a Moving Target,” *Operations Research* **29**, 1226-1230.

Washburn, A. R. (1983), “Search for a Moving Target: The FAB Algorithm,” *Operations Research* **31**, 739-751.

Washburn, A. (1998), “Branch and Bound Methods for a Search Problem,” *Naval Research Logistics* **45**, 243-257.

Appendix B—Simulation Programming References

Birtwistle, G. M. (1979). *Discrete Event Modelling on Simula*, Macmillan Press

Breiman, Leo. (1973), *Statistics: With a View Toward Applicatio*n, Houghton Mifflin Company, Boston, Massachusetts

Brooks, Federick P., Jr. (1975), *The Mythical Man-Month, Essays on Software Engineerin*g, Addison-Wesley Publishing Co., Reading, Massachusetts

CACI, Inc. (1997), *SIMSCRIPT II.5 Reference Handboo*k, CACI, Inc., Los Angeles, California

Dahl, O. J. and K. Nygaard (1966). "SIMULA- An Algol Based Simulation Language," *Comm. ACM* **9**(9), pp. 671-678 [Note: this describes *Simula-1*, a precursor to the OO language Simula_67]

Dahl, O.-J. and K. Nygaard (1967). -*SIMULA- A language for Programming and Description of Discrete Event Systems*, Oslo 3, Norway, Norwegian Computing Center, Forskningveien 1B, 5th edition, September 1967, 124 pages

Dahl, O.-J. (1968). Discrete Event Simulation Languages. *Programming Languages*. F. Genuys. London, Academic Press, pp.349-395

Dahl, O.-J., B. Myhrhaug and K. Nygaard (1970). SIMULA-67 Common Base Language, Norwegian Computer Centre, Oslo, Norway, Pub. **S-22**

Deutsch, C., O. Leuangthong, H. Nguyen, K. Norrena, J. Ortiz, B. Oz, M. Pyrcz, and S. Zanon (2001), "Principles of Monte Carlo Simulation: A One-Day Short Course," Centre for Computational Geostatistics (http://www.ualberta.ca/~cdeutsch/MCS-course.htm)

Fishman, G. S. (1973), *Concepts and Methods in Discrete Event Digital Simulatio*n, John Wiley & Sons, New York

Fishman, G. S. (1979), *Principles of Discrete Event Simulatio*n, John Wiley & Sons, New York

Franta, W. R. (1977). *The Process View of Simulation*, North-Holland

Golden, D. G. and J. D. Schoeffler (1977), "GSL — A Combined Continuous and Discrete Simulation Language," *SIMULATION*, Vol. 20, pp. 1-8, Simulation Councils, Inc., La Jolla, California, (January 1977)

Gordon, Geoffrey (1978), *System Simulatio*n, 2nd ed., Prentice-Hall, Englewood Cliffs, New Jersey

Kirkerud, B. (1989). *Object-Oriented Programming with SIMULA*, Addison-Wesley

Kiviat, P. J., H. M. Markowitz, and R. Villaneva (1997), *SIMSCRIPT II.5 Programming Language*, (edited by E. C. Russell), CACI, Inc., Los Angeles, California

Kiviat, P. J., R. Villaneva, and H. M. Markowitz (1968), *The SIMSCRIPT II Programming Language*, Prentice Hall, Englewood Cliffs, New Jersey

Kleine, Henry (1977), *Software Design and Documentation Language*, Document 77-24, Jet Propulsion Laboratories, Pasadena, California

Law, Averill M., and W. David Kelton (1981), *Simulation Modeling and Analysis*, McGraw-Hill, New York

Markowitz, Harry M., Bernard Hauser, and Herbert W. Karr (1963), *SIMSCRIPT, A Simulation Programming Language*, Prentice-Hall, Englewood Cliffs, New Jersey

Meyer, B. (1988). *Object-oriented Software Construction*, Prentice Hall

Payne, W. H., J. R. Rabung, and T. P. Bogyo (1969), "Coding the Lehmer Pseudorandom Number Generator," *Communications of the ACM*, Vol. 12, No. 2, ACM, New York, (February 1969)

Pooley, R. J. (1987). An Introduction to Programming in SIMULA. Oxford, Blackwell Scientific Publications

Pritsker, A.A.B. (1973), *The GASP IV Simulation Language*, John Wiley & Sons, New York

Schriber, Thomas J. (1974), *Simulation Using GPSS*, John Wiley & Sons, New York

Shannon, Robert E. (1975), *Systems Simulation —The Art and the Science*, Prentice-Hall, Englewood Cliffs, New Jersey

Solomon, Susan L. (1977), "Simulation as an Alternative to Linear Programming," *Simuletter* (a quarterly publication on modeling and simulation) by SIGSM (ACM), Vol. 8, No. 2, pp. 13-15, ACM, New York, (January 1977)

Appendix C—Mathematics, Numerical Algorithms, and Navigation References

Abramowitz, M. (ed.), & I. Stegun (ed.) (1984), *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*, U. S. Government Printing Office

Bowditch, N. (2002), *The American Practical Navigator: An Epitome of Navigation*, National Imagery and Mapping Agency, Pub. No. 9, http://www.irbs.com/bowditch/

Dutton, B. & E. S. Maloney (1985), *Dutton's Navigation and Piloting*, U. S. Naval Institute

Gradshteyn, I. S., I. M. Ryzhik, Alan Jeffrey (ed.), D. Zwillinger (2000), *Table of Integrals, Series, and Products*, Academic Press

Graham, R., Orin Patashnik, D. E. Knuth (1994), *Concrete Mathematics: A Foundation for Computer Science, 2nd ed.*, Addison-Wesley

Knuth, D. E. (1997), *The Art of Computer Programming: Fundamental Algorithms (Vol. 1, 3rd ed)*, Addison-Wesley

Knuth, D. E. (1997), *The Art of Computer Programming: Seminumerical Algorithms (Vol. 2, 3rd ed)*, Addison-Wesley

Knuth, D. E. (1998), *The Art of Computer Programming: Sorting and Searching (Vol. 3, 3rd ed)*, Addison-Wesley

Kokoska, S. & D. Zwillinger (2000), *CRC Standard Probability and Statistics Tables and Formulae, Student Edition*, CRC Press

Ministry of Defence (UK) (1987), *Admiralty Manual of Navigation Volume 1*, Ministry of Defence, UK.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, B. P. Flannery (1992), *Numerical Recipes in Fortran: The Art of Scientific Computing*, Cambridge University Press

Press, W. H., S. A. Teukolsky, W. T. Vetterling, B. P. Flannery (1993), *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press

Press, W. H., S. A. Teukolsky, M. Metcalf (1996), *Numerical Recipes in Fortran 90: The Art of Parallel Scientific Computing (Fortran Numerical Recipes, Vol. 2)*, Cambridge University Press

Press, W. H., S. A. Teukolsky, W. T. Vetterling, B. P. Flannery (2002), *Numerical Recipes in C++: The Art of Scientific Computing*, 2nd ed., Cambridge University Press

Vetterling, W. T., S. A. Teukolsky, W. H. Press (1993), *Numerical Recipes Example Book (C)*, 2nd ed., Cambridge University Press

Yang, D. (2000), *C++ and Object Oriented Numeric Computing for Scientists and Engineers*, Springer Verlag

Zwillinger, D. (ed., Chemical Rubber Company) (1996), *CRC Standard Mathematical Tables and Formulae*, CRC Press

Appendix D—Other Important Reports and Publications

Allen, A. and J. Plourde (1999), *Review of Leeway: Field Experiments and Implementation*, U. S. Coast Guard Research and Development Center, NTIS # ADA 366414.

Blackwell, N. J., R. R. Simmons, J. R. Watson (1982), Preliminary Study on Scanning Techniques Used by U. S. Coast Guard Lookouts During Search and Rescue Missions, U. S. Army Aeromedical Research Laboratory, NTIS # ADA 120597.

Bucklin, B. L. (1971), *Field Dependence & Visual Detection Ability*, U. S. Army Picatinny Arsnal, NTIS # AD 724115.

Edwards, N. C., S. R. Osmer, T. J. Mazour, and D. B. Bouthilette (1978), *Analysis of Visual Detection Performance (Fall 1978 Experiment)*, U. S. Coast Guard Research and Development Center, NTIS # ADA 065118.

Edwards, N. C., T. J. Mazour, G. L. Hover and S. R. Osmer (1980), *Analysis of Visual Detection Performance for 16-Foot Boat and Life Raft Targets*, U. S. Coast Guard Research and Development Center, NTIS # ADA 083565

Edwards, N. C., T. J. Mazour, R. A. Belmont and S. R. Osmer (1980), *Evaluation of National SAR Manual Probability of Detection Curves*, U. S. Coast Guard Research and Development Center, NTIS # ADA 095748

Edwards, N. C., S. R. Osmer, T. J. Mazour and G. L. Hover (1981), Factors Affecting Coast Guard SAR Unit Visual Detection Performance, U. S. Coast Guard Research and Development Center, NTIS # ADA 112995.

Frost, J. R. and L. D. Stone (2001), *Review of Search Theory: Advances and Applications to Search and Rescue Decision Support*, U. S. Coast Guard Operations (G-OPR), NTIS # ADA 397065.

Ketchen, H. G., L. Nash, and G. L. Hover (1983), *Analysis of U. S. Coast Guard HU-25A Visual and Radar Detection Performance*, U. S. Coast Guard Research and Development Center, NTIS # ADA 133380.

Osmer, S. R., L. Nash, G. L. Hover and T. J. Mazour (1982), *Utilization of AN/APS-94 Side-Looking Airborne Radar Systems in Search and Rescue*, U. S. Coast Guard Research and Development Center, NTIS # ADA 114484.

Osmer, S. R., L. Nash, G. L. Hover and T. J. Mazour (1982), *Preliminary Assessment of U. S. Coast Guard Short Range Recovery (SRR) Forward Looking Infrared (FLIR) System Small Target Detection Performance*, U. S. Coast Guard Research and Development Center, NTIS # ADA 117916.

Remondini, D. I., M. Light, M. L. Everson, L. Nash, D. L. Good and S. R. Osmer (1982), *A Pilot Study of Human Factors in SAR*, U. S. Coast Guard Research and Development Center, NTIS # ADA 117917

Robe, R. Q. and G. L. Hover (1986), *Visual Sweep Width Determination for Three Visual Distress Signalling Devices*, U. S. Coast Guard Research and Development Center, NTIS # ADA 176224.

Weisinger, J. R. (1983), *Validation of the Visual Detection Methodology*, U. S. Coast Guard Atlantic Area, Operations Analysis Branch (Memo report from Daniel H. Wagner, Associates under contract DTCG26-83-C-0019).

Weisinger, J. R. (1984), *Analytical Techniques to Estimate Lateral Range Functions for Visual Detection*, U. S. Coast Guard Atlantic Area, Operations Analysis Branch (Final report from Daniel H. Wagner, Associates under contract DTCG26-83-C-0019).

International Aeronautical and Maritime Search and Rescue Manual, International Civil Aviation Organization and International Maritime Organization (1999).

"Evolution of SAR…An Editorial, *Air Sea Rescue Bulletin* (July, 1946a), pp. 2-11, U. S. Coast Guard for the Air Sea Rescue Agency.

"Aerial Search—Fact or Fancy?", *Air Sea Rescue Bulletin* (August-September, 1946b), pp. 1-11, U. S. Coast Guard for the Air Sea Rescue Agency.

Methods for Locating Survivors Adrift at Sea on Rubber Rafts, H. O. No. 235, U. S. Navy Hydrographic Office (1944).

National Search and Rescue Manual, Joint Pub 3-50, COMDTINST M16120.5A, Joint Chiefs of Staff and U. S. Coast Guard (1991).