



**NAVAL
POSTGRADUATE
SCHOOL**

MONTEREY, CALIFORNIA

A Review of the ASW Model in ITEM

by

Alan Washburn

February 2005

Approved for public release; distribution is unlimited.

Prepared for: Chief of Naval Operations (N81)
2000 Navy Pentagon
Washington, DC 20350-2000

**NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93943-5001**

RDML Patrick W. Dunne, USN
President

Richard Elster
Provost

This report was prepared for and funded by Chief of Naval Operations (N81), 2000 Navy Pentagon, Washington, DC 20350-2000.

Reproduction of all or part of this report is authorized.

This report was prepared by:

ALAN WASHBURN
Professor of Operations Research

Reviewed by:

LYN R. WHITAKER
Associate Chairman for Research
Department of Operations Research

Released by:

JAMES N. EAGLE
Chairman
Department of Operations Research

LEONARD A. FERRARI, Ph.D.
Associate Provost and Dean of Research

A Review of the ASW Model in ITEM

by

Alan Washburn

Abstract

This is a brief, technical review of the Anti-Submarine Warfare (ASW) part of ITEM, a widely used theater combat model. Several apparent deficiencies are pointed out. The recommendation is that ITEM's documentation be improved, and that a more extensive verification be carried out.

Introduction

This review is part of OPNAV (N81)'s Methodology Improvement Working Integrated Process Team (MI-WIPT) review of the ASW portion of the ITEM joint theater campaign model. Specifically, my intention is to contribute to the Process Design Plan for ITEM. My review is based entirely on the technical manual and selected source code provided by SAIC. My primary qualification for reviewing the ASW portion is knowledge of stochastic processes and search theory (Washburn (2002)). I also have some experience in constructing combat models, both Monte Carlo and deterministic.

This review should be regarded as incomplete. It was initiated due to the author's interest in the subject, and then terminated due to lack of funding. A more in-depth review would involve installing and repeatedly running ITEM, which I have not done. Section 5 includes recommendations for continuation.

SAIC has been most cooperative in helping me to review the ASW portion ITEM. I have revised an earlier version of this document based on input from Steve Vedder at SAIC. SAIC has also been helpful in providing documentation and answering my questions on how ITEM works. The comments that follow, however, are entirely my own.

1. The Aggregation Issue

ITEM is a deterministic, time-stepped, campaign level model. Although it is deterministic, ITEM deals with probabilities and expected values, particularly in the ASW part. This is a potential source of inaccuracy in a deterministic model, more so than in a Monte Carlo model. The level of aggregation is crucial.

Let us compare two models of a ship passing through a minefield under the assumption that there are ten mines in the minefield, each of which will independently damage the ship with probability 0.07. The issue is whether the ship is damaged by at least one mine. Model 1 is a Monte Carlo simulation where each mine is tested against a

random number to determine whether the ship is damaged. If the ship survives every mine, then it survives the minefield. Now, the probability that the ship is damaged by some mine in this situation is exactly $1 - (1 - 0.07)^{10} = 0.516$, an aggregated probability for the whole minefield. Model 1 might therefore test a single random number against 0.516, thus saving itself the trouble of generating nine random numbers. The results are equivalent; it does not matter whether one random number is tested against 0.516 or ten random numbers are tested against 0.07.

Model 2, like ITEM, does not employ random numbers. The approach is to first compute the probability of the event under consideration, and then declare that the event has happened if the probability exceeds some input threshold T , typically in the vicinity of 0.5. Model 2 is not ITEM, unless by accident. This is a review of the ASW part of ITEM, not the mine warfare part, and I do not know how ITEM models mine warfare. I am using a mine warfare example only because of my own familiarity with the type. In the aggregated minefield example, the ship would be damaged because 0.516 exceeds the 0.5 threshold. However, the ship would *not* be damaged if the mines were examined one at a time. Even if the inequality is checked ten times, 0.07 is not larger than 0.5. The results of the deterministic model thus depend on the level of aggregation, with high levels of aggregation being best.

Ideally, a deterministic model would compute only some ultimate probability such as the probability of sanitizing a region within 48 hours, and then employ the threshold exactly once to decide whether the event happens. The trouble is that the analog of raising 0.93 to the tenth power is usually not available for complicated events such as sanitizing an area of the ocean; indeed, the whole motivation for constructing a model in the first place may be the clear impossibility of computing the probability of such highly aggregated events. Monte Carlo simulation responds well to building up an aggregated picture out of details, but deterministic models do not. The specific danger in time-stepped deterministic models is that the probabilities will all be so small that nothing will ever happen, as in the minefield example.

As the above example illustrates, deterministic models of random phenomena are harder to write accurately than are Monte Carlo simulations. The Monte Carlo analyst, if unaware of the right method for aggregating mines, can simply go over the mines one at a time. If he happens to know the right aggregation method, the effect is to slightly increase the efficiency of the simulation, but the accuracy is unaffected. The deterministic analyst does not have the option of simply writing a loop around a simple operation, and must continually search for opportunities to aggregate accurately. This may be difficult. If there were mines of a second type present, for example, each with a different probability of damaging the ship, then they should all be aggregated together. That is easily said, but there are situations where “aggregating everything together” is not easy. What if a second ship goes through the minefield? The number of mines remaining may have been decremented by the first ship, since a mine can detonate only once. Should we subtract one from the number of mines remaining? If so, from which mine type? Should we instead subtract the probability that the ship is damaged from the number of mines remaining (subtract the expected number of mines lost, in other words)? If so, we must be prepared to continually cope with the possibility that the number of mines remaining is not an integer. What if a second ship deliberately follows the first ship? If the first ship has survived, then her luck should transfer to the second, since the

first has more or less proved that there are no mines in the transit channel. For this question, the expected value idea does not provide even a partial remedy. These questions can all be easily addressed in a Monte Carlo model, but a deterministic model will have to either grow complicated or introduce inaccuracies. The mine warfare community has been preoccupied with the aggregation issue for decades, seeking a deterministic model that is simple, while retaining what T. J. Horrigan calls “configuration”.

I do not mean to argue against deterministic models. They are simple, fast, and reproducible—all powerful virtues. It is also true that the simplicity of the Monte Carlo paradigm can lead to the incorporation of detail to the point where Monte Carlo models cease to be simple or even useful, witness the recent demise of JWARS. However, deterministic models are difficult to write accurately, much more so than is a Monte Carlo simulation of the same situation.

The creators of ITEM are clearly aware of the aggregation issue, since probabilities are aggregated over time before being subjected to a threshold. There are still some aggregation issues, however, as will be pointed out in the appropriate place below. In the meantime, bear in mind that deterministic models of random phenomena are most reliable when the number of threshold tests is as small as possible.

2. General Approach to Search

ITEM’s general approach to search is to aggregate detection probability over time by assuming that detections in distinct time intervals are independent. If $P(t)$ represents the cumulative detection probability at time t and $p(t)$ represents the (non-cumulative) detection probability in a time interval of length Δ surrounding t , then the operative update formula is

$$P(t + \Delta) = 1 - (1 - P(t))(1 - p(t)).$$

In other words, there will be a detection before $t+\Delta$ unless there is no detection before t and no detection in the interval $[t, t+\Delta]$. In the limit where Δ approaches 0 and $p(t)$ is proportional to Δ , this leads to $P(t) = 1 - \exp(-Kt)$, where K is the proportionality constant. With $K = VW/A$, this is the random search formula developed by Koopman and his colleagues in World War II. The same assumption can also be used to aggregate over platforms, as well as time. The independence assumption is generally thought to be a realistic, perhaps slightly pessimistic, assumption about searchers who attempt to act coherently. It is certainly appropriate for an aggregated, theater-level model like ITEM to make such an assumption.

When targets move even while they are being sought, there is both bad news and good news for the searcher. The bad news is that target movement tends to spoil coherent search plans. The good news is that the target may find the searcher, rather than vice versa, since it is relative speed that matters. The good news is handled in ITEM by the dynamic enhancement theory that was also developed in World War II, and the bad news is ignored. Given the commitment to random search, this is reasonable. It is hard for movement to spoil the coherence of a plan that is not attempting to cohere.

3. Specific Issues and Errors

This section is keyed to various sections a.b.c of the ITEM Technical Manual (SAIC, 2003).

13.3.7 Compute Probability of Detection of Submarines versus ASW Platforms

Suppose m searchers look for n submarines in some fixed area A , with searcher i having speed V_i and sweepwidth W_{ij} against submarine j . The rate at which i detects j is $r_{ij} \equiv V_i W_{ij} / A$, and the mean time for i to detect j is the reciprocal of this quantity. ITEM does not go over the submarines one at a time, calculating a detection probability for each one. Instead, ITEM sums all of the detection rates to obtain one superdetection rate

$r = \sum_{i=1}^m \sum_{j=1}^n r_{ij}$, the rate at which some searcher detects some submarine, and the time at

which a report gets generated is based on this sum over both targets and searchers. The high rate r will lead to short times; the more vehicles are present in the area, the faster something will happen. The problem in ITEM occurs not with finding the first submarine, but with finding the last.

To be specific, suppose that $m=1$, $n=5$, and $r_{ij}=1$ for all searchers and submarines. Suppose further that submarines don't shoot back, and that every submarine is sunk as soon as it is found. The initial rate of detection is 5, so the time until the first submarine is detected/sunk is $1/5=.2$. Once the first submarine is sunk, there are only four left, so the additional time until the second is detected/sunk is $.25$. The additional time for the last submarine will be 1, on the average, in agreement with the intuitive notion that finding the last object always takes longer than finding the first. The exact total time to find and sink all 5 submarines is $0.2 + 0.25 + 0.33 + 0.5 + 1 = 2.28$, on the average. In ITEM, the comparable computation would be to solve $1 - \exp(-rt) = T1$ for t , where $T1$ is the detection threshold and $r=5$, the initial detection rate. The solution is 0.14 if $T1=0.5$; smaller than the correct value by a factor of about 16. The comparison would not be so bad in an example where ITEM could exercise its capability of gradually eliminating submarines as they are killed, particularly if the submarines all differed a lot from each other, but the magnitude of the error in this simple example should nonetheless give one pause. If the searchers were trying to "sanitize" an area that includes several similar submarines, there could be a significant exaggeration of the speed with which they could do it.

It may not be important to fix detection problems, since detections are essentially independent of engagements as ITEM is currently organized. However, engagements are handled similarly, and there the issue is important. I don't know how to fix it. The present model sacrifices the average to get an extreme (the first detection time) right. I would be more inclined to get the average right by treating the submarines independently, one at a time, thus sacrificing extremes for the average. In my simple example, this would result in killing all five submarines at time 0.693, a time five times as large as the current value. This would still significantly understate the sanitization time, while also overstating the time of first detection.

In general, search in ITEM is conducted by units whose locations are not specified, except that they lie within a known area A , for targets whose locations are not specified,

except that they lie within a different area B. Let AB be the intersection of the two areas. Having computed a target's detection probability, ITEM corrects it by multiplying it by the ratio AB/B. The implied assumption is that a sub is stationary over a time period of length Δ , and is equally likely to be anywhere within B. Another implied assumption is that all searchers in A manage to be in the overlap area AB, since there is no similar searcher correction. One might argue that, in any scenario where all of the searchers in A are actively searching in AB, their targets would tend to concentrate in the part of B that is not in A, rather than simply distributing themselves at random. If one were to so argue, then the implied assumptions would be optimistic. These implied assumptions are my invention. Although the technical manual makes clear which formulas are in use, it does not actually make clear what assumptions lie behind them.

What would ITEM do if searcher area A overlapped two separate sub areas, B1 and B2? Would the searchers simultaneously appear in both overlap areas?

There is another overlap issue here. Detections in separate time intervals are assumed to be independent, so the events that a target is in the joint area in separate time intervals are assumed to be independent. In effect, each target is assumed to be moving around rapidly enough in B that its position Δ from now is independent from its current position. If the dimensions of B are 30×30 nm, and if $\Delta=10$ min, then the implied target speed is on the order of 180 kt. While this may be acceptable, I think it is clear that the time interval should not be made even smaller by some user who is under the impression that smaller is better. If $\Delta=1$ min, for example, then, in a 10 min period, there will be 10 opportunities for detection, and the cumulative effect may be such that the detection probability exceeds the probability that the target is in the joint area. For example, if $VW/A=1$ per min and if the inclusion probability is .1, the detection probability in 1 min is $.1(1 - \exp(-1))=0.0632$. This is, as expected, smaller than the inclusion probability. However, the detection probability in 10 min is $1 - (1 - .0632)^{10} = 0.48$, which is much greater than the inclusion probability. How can one detect a target that isn't even there?

There is also a conceptual error in this section; namely, the statement that the areas swept by various platforms do not overlap. Given the random search formula that is in use, the assumption is actually that the areas *do* overlap to the extent implied by having the platforms patrol in an unorganized fashion. If they did not overlap, we would not be using the random search formula. Adding up the areas is nonetheless the right thing to do.

13.3.9 Compute Probability of Detection of ASW Platforms versus Submarines

There are three sources of ASW detection in ITEM: platforms, SOSUS, and aircraft. Each source has an independent detection probability: p_1 , p_2 , or p_3 . ITEM wisely aggregates the three numbers into one before subjecting the result to a threshold, but unfortunately aggregates them by simply taking the largest of the three. If the sources are independent, the probability that at least one of them detects the target is $1 - (1 - p_1)(1 - p_2)(1 - p_3)$, which is always larger than the maximum. Using the maximum is a pessimistic error. The fix is to use the formula given above. In other words, the same idea being used to aggregate over time (multiply independent failure probabilities together) should also be used to aggregate over detection sources.

13.3.10 Compute Cumulative Sonar Probability of Detection

I do not understand the motive for calculating only one sonar probability of detection (PD), instead of one for each target, given that each target is already a separate object with its own survival probability. That would avoid the necessity of using the average value of the evasion probability (p_{evade}) to correct for evasion. The implied assumption about evasion is that, every time a detection occurs, the submarine immediately makes the detection inoperative with probability p_{evade} . The experience of the searcher would simply be one of fewer observed detections for evasive submarines, so p_{evade} might be measured as the ratio of the detection rate for evasive submarines to the detection rate for unevasive submarines. I cannot otherwise imagine how to determine p_{evade} from actual experiments, and suspect that it is simply an outright guess on the part of the user.

13.3.11 Compute Cumulative SOSUS Probability of Detection

This section of the manual does not correspond to the way SOSUS detections are currently handled in ITEM. See Section 4 for comments based on source code.

13.3.12 Compute Cumulative Radar Probability of Detection

Detection of periscopes and snorkels by radar are handled similarly in ITEM, so I will only comment on periscopes.

The radar detection range is handled by including a detection range for a standard periscope in the data, and then adjusting it based on the assumption that the signal-to-noise ratio is proportional to the radar cross section of the actual periscope and the inverse fourth power of range (hence the $\frac{1}{4}$ power of rel_det_periscope in the equation for ac_search_area). This seems reasonable to me, but technical experts might be able to supply a slightly different power, or find a way of incorporating a radar horizon. Making the corresponding change in ITEM would be trivial, but perhaps significant.

The data also include an item called pd_scope that is subsequently used to modify the area swept. This is an error. Once one has determined the detection range against periscopes, it does not make sense to later say that maybe those weren't really detections after all. I cannot imagine how such a number could be realistically measured or estimated. It should be eliminated (taken to be 1.0).

The overlap question is handled differently in this section than in 13.3.7. If A and B overlap, then the joint probability is $p_{\text{overlap}} = (AB/A)(AB/B)$, rather than just AB/B as before. Now the aircraft, in addition to the subs, are assumed to be randomly distributed over their patrol area. Furthermore, p_{overlap} is incorporated into the coverage ratio, rather than applied as a correction to $p_{\text{scope_detect}}$. The implied assumption is that both aircraft and subs are moving about rapidly within their patrol areas, even within an interval of length Δ . The same assumption is applied to the periscope duty cycle; that is, the submarine is in effect assumed to raise and lower its periscope several times within each period of length Δ , even while exposing it only the given fraction of the time (duty cycle). The same high frequency assumption is applied to aircraft motion, sub motion, and periscope exposure. Note that $p_{\text{scope_detect}}$ can approach 1.0, thus exceeding both $\text{periscope_duty_cycle}$ and p_{overlap} .

There is no easy fix for this. One could, of course, first calculate `p_scope_detect` and then correct it by multiplying by `periscope_duty_cycle` and `p_overlap`. This would keep `p_scope_detect` below each factor and make the treatment parallel 13.3.7, but then I would complain as I did in that section. The basic problem is that `periscope_duty_cycle` does not sufficiently describe periscope emergences because nothing having the units of time is in the database. Given that a periscope is up, say, $1/60^{\text{th}}$ of the time, it matters whether it stays up for one hour out of sixty or one second per minute. The latter corresponds to what is currently done, the former corresponds to the correction factor approach, and the truth is probably somewhere in between. The subject is really just too difficult for a simple deterministic model to get right.

Historically, I believe that the most common method of detecting periscopes has been by eyeball. Perhaps that will change with better radar systems, but it is odd that eyeballs are missing from the list of periscope detection mechanisms.

13.3.13 Generate ASW Sensor Contact Report

A report is issued whenever any of three cumulative detection probabilities exceeds a threshold, one for each platform type. Perhaps only one threshold should be applied to the three jointly, in accordance with the idea that the fewer such tests, the better. Cumulative detection probabilities can only go up, so, once a report is issued, there will be a report issued in every interval thereafter.

13.3.14 Compute the Probability that ASW Platforms Engage Submarines

Detections are not currently coupled to engagements in ITEM, so all of the detection parameters can be changed without affecting the course of the war, at least as far as attrition is concerned. Attrition is handled through a completely separate engagement process.

The detection and engagement processes are mathematically similar, differing primarily in that engagement ranges are smaller than detection ranges. The procedure is clear enough, but I don't see the logic behind it. Actual engagements are triggered by detections, with the detecting platform either cueing some other platform or closing to within engagement range. The fact that engagement ranges are smaller than detection ranges is barely relevant as far as predicting the number of engagements is concerned. Nonetheless, ITEM proceeds by replacing the detection range with an engagement range, calculating an engagement coverage ratio, and ultimately updating a cumulative engagement probability in each time period, the same basic process that is used for detections. The engagement range is simply the minimum of the weapon range and the detection range, except that the detection range is first subjected to reduction by being multiplied by an `engagement_factor`. Earlier comments about detection apply equally to engagements, but some new issues arise.

13.3.15 Process ASW Engagement

When `cumAswPe` exceeds a threshold, all submarines in the associated area are subjected to attack. In fact, unless they are evasive submarines, they will be subjected to attack in every succeeding interval, since `cumAswPe` will never decrease. This is harmless enough in the case of detections, but is likely to result in the quick demise of

nonevasive submarines. I don't think that this makes sense. Results are certainly very sensitive to the length of the time interval Δ , since there is an attack every Δ . If Δ were very small, nonevasive submarines would disappear instantly, as soon as they are engaged. The best case would be if Δ were the length of time required to make the typical extended ASW attack, but that surely exceeds the ten-minute standard of ITEM. Here is one possible fix:

- Every time cumAswPe crosses a threshold, an engagement occurs and the submarine instantaneously suffers a change of state with various given probabilities. These probabilities correspond to probabilities that apply for attacks that continue until contact with the target is lost (not for attacks that continue for Δ).
- Every time cumAswPe crosses a threshold, it is reset to 0. This represents the idea that attacks only cease because contact has been lost.

Resetting cumAswPe to 0 will prevent the current cascade of attacks. The new interpretation of cumAswPe is "probability that there has been an engagement in the time since the most recent attack". The meaning of ASW_salvo_size would have to be changed to "average number of weapons used in prosecuting a submarine until contact is lost", and other changes of interpretation might also be necessary.

Two state probabilities are tracked for each sub: the probability of survival (p_{survival}) and the probability that the combat system is still operative (p_{scop}). When the submarine is attacked, the two are updated using the formulas

$$p_{\text{survival}} \leftarrow p_{\text{survival}}(1 - Pk), \text{ and } p_{\text{scop}} \leftarrow p_{\text{scop}}(1 - Pc),$$

where Pk and Pc are the probabilities of killing the sub and knocking out its combat system, respectively. Both of these numbers are essentially data, albeit data scaled to account for the number of shots. My concern is with the meaning of Pc . If I were asked to estimate that number, I would interpret it as the probability of knocking out the combat system, *given* that the submarine otherwise survives the attack. My reason for the condition is that functioning of the combat system is irrelevant if the submarine is dead. With that interpretation, Pc could exceed Pk . Now that I can see the update equations, I can see that the correct interpretation for Pc is actually the probability that either the submarine is killed or the combat system is knocked out or both. With that interpretation, p_{scop} never exceeds p_{survival} , and p_{scop} retains its desired meaning as the probability that the submarine is still both alive and functioning. That meaning is desired, I think, because p_{scop} is a factor in computing the number of salvos fired by the submarine in any future engagement. My point is that a user asked to input Pc is likely to interpret it incorrectly.

Once cumAswPe exceeds its threshold for nonevasive targets, it will remain above the threshold for all future time intervals, and the target will continue to absorb salvos until its p_{survival} has decreased below the survival threshold. In each time period, cumAswPe is used to initialize salvos_expended. Although the technical manual omits the factor, salvos_expended is one of several factors in salvos_engagement.

Let's make a small comparison. Suppose we have only one searcher, one target, and one weapon with kill probability Pk to launch in each time period. The target is assumed to be nonevasive and to not shoot back, so the shooter's p_{scop} remains 1 throughout. Assuming that we shoot-look-shoot until the target is killed, the true average number of

weapons expended is $1/Pk$. For comparison, in ITEM we will first wait until cumAswPe exceeds its threshold $T1$ before taking the first shot at the target. We will continue to shoot in successive periods until the target's survival probability decreases from 1 to the survival threshold $T2$. The survival probability decreases by a factor of $(1-T1*Pk)$ in each period (the technical manual would have this factor being $(1 - Pk)$, but the manual's equation for salvos_engagement differs from the ITEM code in not including the factor salvos_expended). Therefore, the number of periods required is $\log(T2)/\log(1-T1*Pk)$, after which the target will be deleted and no further action will be taken against it. Ignoring the fact that cumAswPe will grow slightly with each period, the number of shots in each period is $T1$ on account of the aforementioned initialization, so the total number of shots over all periods is $Shots_{ITEM} = T1 \frac{\log(T2)}{\log(1 - T1 * Pk)}$. The default values for $T1$ and $T2$ are 0.1 and 0.5. If $Pk=0.5$, that means $Shots_{ITEM} = 1.35$, to be compared with the reciprocal of Pk , or 2. A fix here would be to take $T2$ to be $\exp(-1)=.368$, rather than the current .5, but there may be other good reasons for leaving $T2$ at .5. If $T1$ were set to .5, we would have $Shots_{ITEM} = 1.20$; that is, for this purpose it is important that $T1$ be kept small compared to .5.

13.3.17 Decrement Detection and Engagement Probabilities

Evasion probabilities are used for two purposes. One is during the search phase, where the effect of the evasion probability is to decrease the sweepwidth as earlier described in 13.3.10. That same logic is used in computing cumAswPe. So far, there is no effect that would make cumAswPe decrease; it is merely a matter of how fast it increases. However, the second use of evasion is right after an engagement, where cumAswPe is multiplied by $(1 - p_evade)$. The effect is to reduce cumAswPe. It is no longer possible to interpret the quantity as the cumulative probability of engagement since time 0, since that quantity can only increase with time. Nor can we interpret it as the cumulative probability of engagement since the last engagement, since that quantity would have to get reduced to 0 after every engagement. Is there any reasonable interpretation we can place on cumAswPe that will justify the way it is used in ITEM? The technical manual does not address the issue, but here is one possibility. Assume that

- During the search phase, ASW makes occasional contacts that might result in an engagement, but does not act immediately. Some of these contacts are immediately foiled by evasive targets and are therefore inoperative, but the rest are used to initiate or continue target tracking. Except at the moment of engagement, track is never lost, once established.
- In any interval when cumAswPe exceeds a threshold, ASW engages the target.
- After an engagement, a surviving submarine will attempt to evade. The ASW track is lost if and only if the submarine's attempt succeeds.

If the evasion attempt succeeds, cumAswPe is reduced to 0. If it fails, cumAswPe is not affected, since ASW is still tracking. The new cumAswPe is therefore the old value times the nonevasion probability plus 0 times the evasion probability, as ITEM calculates. After being reduced by the possibly successful evasion, cumAswPe resumes the process of increasing. The required interpretation of cumAswPe is "probability that

the submarine is currently being tracked". Note that the evasion probability does not affect the submarine's ultimate fate or ultimate munitions consumption, but only has the effect of delay.

4. Other Observations

SAIC has provided me with a paper copy of the ASW source code, which has on several occasions clarified doubts left by the technical manual. In the process of perusing the source code, I have noticed the following:

- In computing the area covered by SOSUS, the first step is to find the SOSUS detection range R . SOSUS doesn't move, so the relative motion comes from the submarines themselves. The area covered by SOSUS is taken to be πR^2 for stationary subs, or $2VR\Delta$ for subs with positive speed V . I have several objections to this.
 - The area covered is a discontinuous function of V . A sub with speed .00001 kt will cover less area than a sub with speed 0.
 - The sweepwidth should be R , rather than $2R$, since SOSUS only looks in one direction.
 - Even R is too large, since subs don't always patrol normally to the SOSUS beam.
 - What about the angular width of SOSUS coverage? Should it not enter the computation?
 - The area covered for a stationary sub might better be taken to be 0 than πR^2 . For SOSUS, πR^2 might be 10^4 nm^2 . If Δ is the standard ten minutes, this corresponds to sweeping $60,000 \text{ nm}^2/\text{hr}$. That seems large to me. The same issues occur in assessing the effectiveness of sonobuoys. ITEM essentially assumes that sonobuoys have infinite lifetimes, but are only operative if a P-3 is available to monitor them. This seems reasonable, but the assumption that stationary sensors effectively move through the water at the submarine's speed, while in accord with the dynamic enhancement theory, is optimistic. An alternative analytic model of sonobuoys is the one by Cox (1972).
- The dynamic enhancement theory requires an integral whose symbol is ek in the technical manual. The integral is actually a complete elliptic integral of the second kind, approximated in ITEM by a power series. Elliptic integrals are not well approximated by power series, so ITEM's ek can be in error by about 2%, in spite of including terms up to the 18th power. While this is not a large error, there are methods for approximating ek that are simpler and more accurate.

5. Recommendations

If ITEM is to be further developed, then it is my opinion that development should take the course of debugging and verification, rather than enhancement. Since even my cursory inspection of the technical manual and code has revealed several serious problems, there is reason to suspect others.

The first step should be to revise the technical manual. Errors and omissions such as the ones that I have found should be corrected. Equally important, the assumptions on

which ITEM is built should be made explicit. At the moment, the technical manual has a strong tendency to explain what equations are being used, without explaining how the equations follow from the assumptions being made about combat, or at least how they approximate what should follow from those assumptions. Assumptions that I have earlier claimed are being made implicitly should be made explicitly. Verification is the process of convincing ourselves that the computer code correctly implements the model that we have in mind. Until the model is made clear, verification is impossible.

Some of the processes in ITEM are underspecified. The periscope emergence process is one of them, since it specifies a duty cycle without saying anything about timing. For periscopes I suggest assuming that submerged periscopes emerge in a Poisson process with rate λ , remaining up for a period d before being retracted. The duty cycle in that case would be $\lambda d / (1 + \lambda d)$, which is already part of ITEM's data, but two parameters are needed if the process is to be completely specified. Another underspecified process is how platforms move about within their areas.

The simplest way to verify the various components of ITEM would be to pose a collection of tactical problems such as the ones I have introduced above. The tactical problems should be simple enough to permit solution by means other than ITEM, so that results can be compared. I think it likely that carrying out a sequence of such comparisons would reveal errors in ITEM that I have missed, but which should be corrected. It might also reveal how to set ITEM's thresholds. A program to accomplish this would be appropriate for the Naval Postgraduate School. It would cost on the order of \$100,000, depending on scope, and would be spread over about two years to permit the formulation of theses.

Nunn and Heimerman (1994) reviewed version 5.2.3 of ITEM. They say that

“Almost every time that the review team made a test run involving a previously unused module of code, it encountered some shortcoming in the documentation or ‘bug’ in the code. It is very unlikely that all such surprises were encountered during this review. We recommend that ITEM undergo verification testing to ensure that the design was faithfully implemented into the code.”

The current version of ITEM is no doubt much improved from version 5.2.3. Nonetheless, ten years later, I can only say that I agree.

References

Cox, H. 1972. “Simple Detection Models for Sonobuoy Fields”, Office of the Secretary of Defense OASD(SA)GPP/NFD.

SAIC, ITEM Technical Manual, version 8.6, ITEM-TM-8.6-U-RBC0, 10 October 2003.

Nunn, W. and Heimerman, K. 1994. “Review of the Integrated Theater Engagement Model (ITEM)”, Center for Naval Analyses report CRM 94-19.

Washburn, A. 2002. *Search and Detection*, 4th edition, INFORMS.

Initial Distribution List

1. Research Office (Code 09).....	1
Naval Postgraduate School Monterey, CA 93943-5000	
2. Dudley Knox Library (Code 013).....	2
Naval Postgraduate School Monterey, CA 93943-5002	
3. Defense Technical Information Center.....	2
8725 John J. Kingman Rd., STE 0944 Ft. Belvoir, VA 22060-6218	
4. Richard Mastowski (Editorial Assistant).....	2
Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5000	
5. Professor Alan Washburn.....	4
Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5000	
6. Office of Deputy Chief of Naval Operations.....	1
Director, Assessment Division (N81) 2000 Navy Pentagon Room 4D453 Washington, DC 20350-2000	
7. SAIC ITEM Program Manager.....	1
10260 Campus Point Drive San Diego, CA 92121	
8. The CNA Corporation.....	1
Attn: Dennis Shea 4825 Mark Center Drive Alexandria, VA 22311	