Threat and Response • Interoperability • Battle Management • Strategic Considerations • Guidance and Control • Risk Management • Readiness and Training • Simulations

ABOUT THE EDITORS

Ben-Zion Naveh is the Chief Architect and Special Consultant to WALES, Ltd. He holds a Ph.D. in aeronautical engineering. He has also served as President of Rafael, Director of the Infrastructure and R&D Organization at the Ministry of Defense, and Vice President and Chief Operating Officer of the Scitex Corporation Ltd. He was awarded the Israel Defense Award in 1971.

Azriel Lorber obtained his B.S. from the University of Pittsburgh and his M.S. and Ph.D. in aerospace engineering from the Virginia Polytechnic Institute and State University. Currently with WALES, Ltd. as Senior Consultant, he previously worked for the Israel Aircraft Industries, the Israeli Military Industries, and was Principal Assistant to the Director General of the Ministry of Science in Israel. He authored four books on military technology and weapon systems and numerous articles in professional and military publications.

ABOUT THE BOOK

The biggest single increase in the U.S. defense budget request for modernization spending is for ballistic missile defense, including theater and national systems. Engineers, managers, and policymakers will need to stay abreast of the ever-changing state of the art in theater ballistic missile defense.

Theater Ballistic Missile Defense fills that need. The editors pull together 40 years of experience in Israeli theater missile defense activities to provide you with a single volume of work that addresses issues, state of the design considerations, technologies, and procedures related to the theater ballistic missile defense system.

AIAA
American Institute of Aeronautics and Astronautics
1801 Alexander Bell Drive, Suite 500
Reston, VA 20191–4344 USA
Web site: http://www.aiaa.org

ISBN 1-56347-385-2

Progress in Astronautics and Aeronautics Volume 192

Theater Ballistic Missile Defense

Ben-Zion Naveh and Azriel Lorber, Editors

AIAA

Progress in Astronautics and Aeronautics

THEATER BALLISTIC MISSILE DEFENSE

Edited by
Ben-Zion Naveh
Azriel Lorber

Progress in Astronautics and Aeronautics

Paul Zarchan
Editor-in-Chief

Volume 192

Theater Ballistic Missile Defense

Theater Ballistic Missile Defense

Edited by
Ben-Zion Naveh (Editor)
WALES Ltd., Ramat-Gan, Israel
Azriel Lorber (Associate Editor)
WALES Ltd., Ramat-Gan, Israel

Volume 192
PROGRESS IN
ASTRONAUTICS AND AERONAUTICS

Paul Zarchan, Editor-in-Chief
MIT Lincoln Laboratory
Lexington, Massachusetts

Published by the
American Institute of Aeronautics and Astronautics, Inc.
1801 Alexander Bell Drive, Reston, Virginia 20191-4344

ISBN 1-56347-385-2

This book is dedicated to the memory of the late Brig. Gen. (Ret.) Michael Cohen, head of Materiél Directorate at the Israeli Air Force, founder of WALES Ltd., and recipient of the David R. Israel Award for Meritorious Achievement for his work on the Israeli BMD architecture

Progress in Astronautics and Aeronautics

Foreword

While writing this foreword, I was informed of the spectacular success of the Arrow system as it intercepted a Black Sparrow missile simulation of a SCUD ballistic missile.

I was deeply involved with both the Arrow and Black Sparrow projects from their inception and saw them through advanced stages of funding and development. I was at the center of the early crises, including budgetary difficulties and failed prototypes. Because of my close personal involvement with this project, I was deeply satisfied with its latest success. Yet we must remember to constantly examine our assumptions and focus on future challenges and ways to cope with them, while striving to obtain timely answers in advance and not after the fact. After all, we had the technological capabilities to develop the Arrow system even before the Gulf War, yet we did not prepare ourselves sufficiently for the crisis of 1991. Our assessment of the risks involved in the acquisition of ground-to-ground tactical missiles by the countries of the region was inadequate.

The general trend of thought during the 1970s and 1980s downplayed the importance of defense systems against tactical ballistic missiles carrying conventional warheads. Since the accuracy of the tactical missiles left much to be desired, many concluded that they did not constitute a material threat to military forces or strategic assets. In addition, it was argued that deterrence measures, such as the threat of a significant offensive response, were sufficient to prevent attacks against civilian targets and population centers. Furthermore, it was assumed that hostile countries would not dare to fire missiles armed with chemical or biological warheads at civilian targets.

Indeed, a survey of the brief history of the use of ballistic missiles in various conflicts should have contradicted these assumptions and conclusions. However faced with shrinking budgets and immediate requirements, there was an understandable tendency to postpone long-term needs. Thus, immediate security threats received full attention and support, while uncertain threats, such as the deployment of ballistic missiles, became a secondary budgetary priority.

Ballistic missiles were first used in the Second World War when the Germans developed the extraordinary V-2, a tactical ballistic missile. The V-2 was further developed by the Soviet Union and eventually turned into the ubiquitous SCUD. The V-2 was initially intended for use against large population centers. The British failed in attempts to develop a defense against it. They tried to locate the launchers on the French coast and destroy them by aerial attack but these were well dispersed and, by their very nature, difficult to identify.

The Normandy invasion pushed the launching sites further back and out of Britain's range (although the attacks against Antwerp continued, with more missiles hitting that city than London). This prevented the V-2 from becoming a successful and effective weapon against the British population. Otherwise, the Second World War might have proceeded differently.

During the Iran–Iraq War, in the 1980s, both sides fired ballistic missiles against each other's population centers. Tehran suffered great losses as a result, although

the Iraqi warheads were conventional. The city's inhabitants clearly demonstrated concern and fear as they evacuated the city in droves. It is argued that the effect of missile attacks was one of the reasons for the war's conclusion. Moreover, this experience probably contributed to Iran's decision to launch an intensive program of acquisition and development of ballistic missiles.

On a different note, the Syrian military leadership claimed that the uncontested superiority of the Israeli Air Force was the decisive factor in the 1982 Lebanon war. The Syrians' declared conclusion was that while the Lebanon War was fought using ground-to-air missiles, the next war would be characterized by ground-to-ground missiles. (In 1982, the Syrian air defense, based on ground-to-air missiles, suffered a defeat in the Bekka Valley.) As such, the Syrians intensified their efforts to aquire additional SS-21 and SCUD-B ballistic missiles. They later developed an indigenous capability to produce ballistic missiles while also acquiring longer range missiles, such as the SCUD-C and the SCUD-D.

These facts notwithstanding, the assessment of Israel's defense establishment until the 1991 Gulf War held that there was no justification to develop an anti-ballistic missile defense system. It is important to note that this assessment was not the result of faulty intelligence. There was no lack of data regarding the armaments of the region.

Undoubtedly, the Gulf War significantly shifted the appraisal of the situation. On the one hand, it encouraged the countries of the region to intensify their efforts to acquire ballistic missiles. After all, even a superpower such as the United States, armed with commanding satellite systems and the most advanced reconnaissance capabilities and attack aircraft, failed to stop the launches until the end of the war. At the same time, countries faced with the threat of a ballistic missile attack were encouraged to develop antiballistic missile defense systems as a result.

The ground-to-ground ballistic missile has introduced a new dimension to conventional warfare. It is no longer necessary to share a border in order to mount an attack against an enemy country. Consequently, military doctrines are undergoing change since up until now ground forces constituted the decisive force in warfare. Absent a common border, decisions on the ground are rendered obsolete, making it very difficult to achieve a decisive victory in a conventional war. As a result, several countries have developed chemical warheads to obtain a military advantage by means of non-conventional weapons. This development heightened the need for antiballistic missile defense systems to enable political decision makers the needed time to develop a measured response in the event of an attack. After all, the flight time of a ballistic missile ranges from three to seven minutes, depending on the range.

In order to rehabilitate conventional deterrence, it will become imperative to develop a defensive system capable of destroying offensive weapons while still over the aggressor's territory. A country launching missiles with chemical or other warheads will thereby need to carefully weight the risk of several missiles striking its own territory. Consequently, systems for the interception and destruction of ballistic missiles at the launch and boost phases will be critical to the reestablishment of international stability in several regions of the world.

Even if we succeed in developing suitable counter-measures, we are already facing new concepts of warfare developed as a result of the proliferation of ballistic missile technology. The need for early detection and warning systems, passive and

active defenses, as well as long-range offensive systems will bring about many new developments in various fields, including hardware. Those who reach the correct assessments early on and develop appropriate responses will no doubt contribute to regional stability and prevent future conflicts.

His Excellency
Major General (Ret.) David Ivry
Ambassador of Israel to the United States
February 2001

Table of Contents

II. Threat and Response

III. Interoperability, Battle Management, and Strategic Considerations

Preface

In May 1985, U.S. President Ronald Reagan invited Israel to participate in the Strategic Defense Initiative (SDI) program, popularly known as "Star Wars." This program reflected President Reagan's ambitious decision to develop technologies and weapon systems that could provide defense against strategic ballistic missiles, and the Strategic Defense Initiative Organization (SDIO) was established within the U.S. Department of Defense (DoD) to manage this program.

The invitation to join the program was sent to a small number of countries, all U.S. allies, and all possessing highly advanced levels of defense technologies. The inclusion of Israel amongst those highly developed countries was in itself a compliment and a recognition by the United States of Israel's advanced defense technologies and R&D capabilities. Nevertheless, the decision to join this U.S. initiative was not easy. Scientists in many countries opposed the program for various technological, economic, and strategic reasons. Israel had another consideration—people were concerned that Jewish immigration from the Soviet Union might be compromised by Israel joining a U.S.-led program that was directly challenging the Soviet Union.

Israel also evaluated other aspects of joining the program and assessed the advantages to Israel and the potential contribution Israel could make to the program. The technological community at the Israel Ministry of Defense (IMOD) identified the unique opportunity to become a partner in a very advanced technology program, which would boost and enhance Israel's indigenous technological capabilities, and assessed that Israeli scientists and engineers could make a significant contribution to the United States, despite Israel's relatively small size and population.

It is interesting to note that in those days very few people at the IMOD or the Israel Defense Forces were concerned about the ballistic missile threat as long as the missiles were conventional. IMOD, however, did recognize that joining the program would provide an opportunity to develop technologies that could be applied to ballistic missile defense if and when the need emerged, or if countries hostile to Israel acquired unconventional capabilities. Thus the Israeli government decided to join the SDI program, and in May 1986 a Memorandum of Understanding (MOU) was signed.

Following this decision, a first meeting took place between IMOD and DoD representatives, the U.S. team being headed by Lt. Gen. James Abrahamson, then Director of SDIO. Gen. Abrahamson's message was clear: The technological sky is the limit. Israel is invited to submit research proposals as advanced and as futuristic as it wishes. Proposals accepted by the United States would be funded by SDIO. The only condition was that those research proposals must be relevant to ballistic missile defense.

The IMOD established a new, dedicated, SDI Cooperation Program Office, and that office got to work, soliciting and eventually receiving about 70 research proposals from Israeli universities, research centers, and industries. These proposals were submitted to SDIO; then nothing happened.

It took this SDI Cooperation Office several months to realize that the approach was all wrong. SDIO did not expect Israel to submit proposals in scientific fields

Chapter 8

Interoperability Between Defense Systems

Dror Cohen* and Eitan Yariv†
WALES Ltd., Ramat-Gan, Israel

I. Introduction

THE *Dictionary of Military Terms* defines interoperability as "the ability of systems, units or forces to provide services to and accept services from other systems, units or forces and to use the services so exchanged to enable them to operate effectively together".[1] This ability is fundamental to theater missile defense (TMD) systems in particular. The reason for this is twofold.

First, future wars (as already evidenced in current world affairs) are likely to be fought in coalition settings. This means that the forces of two or more nations are likely to participate in countering a common aggressor. A number of nations are developing defense systems and sensors for countering the tactical ballistic missile (TBM) threat, thus the opportunity for interoperability among systems belonging to different coalition partners clearly presents itself.

Second, TBMs are difficult weapons to counter. Systems and sensors currently in operation or under development have partial, often complementary, abilities in countering them. The TBM travels large distances in a relatively short time, and in each phase of its flight presents defense systems with different problems and intercept conditions. A single system cannot hope to cope with the entire range of these conditions, whereas joint operation of a number of systems with different capabilities offers the possibility of improving overall effectiveness.

Being able to operate different systems and sensors jointly, in a smooth manner, is therefore one of the major considerations guiding defense systems planners. Indeed, the U.S. Ballistic Missile Defense Organization (BMDO) has coined the term "family of systems" to describe its commitment to develop systems that will be able to complement each other and work together as a seamless whole. In this chapter, we will take the view of the planner of defense architecture rather than that of the builder of a weapon system. From this point of view, we shall attempt to define and quantify the possible advantages to be gained by interoperation of defense systems and sensors, as well as the pitfalls that should be avoided. We

*President. Member AIAA.
†Team Leader, Interoperability, External Cueing, and Concept of Operations. Member AIAA.

shall attempt to link between the level of complexity of the chosen implementation of interoperability and the extent of advantages gained. We will also try to highlight some of the problems that may be encountered in implementing various interoperability features in weapon systems, without stating specific solutions or preferences. As a general guideline, we chose to treat this chapter as a survey of the tradeoffs between the complexity of the adopted implementation, the advantages to be gained, and the potential problems. The reader may than choose an adequate working point, using the guidelines and thought processes outlined in the following sections, bearing in mind the qualities specific to the systems comprising a particular architecture and the anticipated threat.

Finally, because interoperability may be implemented between task forces of different nations, in a coalition setup, a few remarks regarding the operational implementation will be provided. These remarks are intended to highlight the problems involved in the operational application of interoperability and not to offer specific schemes or doctrines. Clearly, the writers' position in a small country, which may become the "host" of interoperating parties in an attempt to defend its own territory (as happened in the 1991 Persian Gulf War), influences our views in this respect, and the reader should take this into consideration and make the necessary adjustments to other possible situations. We chose to limit ourselves in discussing the operational issues relevant to ballistic missile defense (BMD) units and avoid dealing with the more general problem of the integration of BMD units in the overall war effort. Such a discussion would entail treatment of the connection between BMD units and air traffic control, passive defense assets, forward deployed fighting units, etc., which is out of the scope of the present volume.

II. Benefits and Drawbacks of Interoperability

In this section, we will try to define in general terms the major benefits and drawbacks associated with interoperability. Some of these benefits and drawbacks lend themselves easily to quantification and will form the backbone of the discussion throughout the rest of this chapter. Others are more qualitative in nature and are mentioned here for the sake of completeness. Theater- and situation-specific considerations should be applied, in order to properly take these factors in account.

Let us begin with the definition of benefits that may be expected from the application of interoperability. These are as follows (in random order).

Sizing: The most obvious benefit, and probably the easiest to make use of, lies simply in the added firepower that additional weapon systems offer the theater defense planner. However, as will be discussed at length later on, simplistic application of the added firepower may be insufficient for the requirements of the combined defense architecture, and may even lead to the emergence of problematic situations.

Footprint: This is a related benefit to that of added firepower. The addition of defense assets to an existing architecture may either be applied to thicken the defense at a given sector or to widen the borders of the defended region.

Better probability of kill (P_k): The existence of varying interception solutions vs a given TBM allows the selection of an optimal plan, optimizing the defense against the specific threat.

Robustness and mutual backup: Having overlapping assets defending a given region allows better robustness in case of system failure resulting from malfunction, resource overload, or enemy counter measures. This quality is intensified if the interoperating systems have different features (e.g., different radar operating frequencies or different interception altitudes allowing multitiered defense), making it much harder for the enemy to neutralize the combined architecture.

Early warning and external cueing: These benefits result from interoperability with longer range sensors than those of the original defense architecture. They are the subject of a separate chapter, and therefore will not be analyzed at length here.

Dealing with a broader range of threats: Added defense systems may incorporate qualities allowing the combined architecture to deal effectively with threats that could not otherwise be countered.

Support in defense processes: Data from additional sources may assist in the application of defense system algorithmic processes such as discrimination (for example between TBM warhead and separated motor), kill assessment, etc.

Cost effectiveness: Ultimately, the aforementioned advantages of interoperability may be utilized to achieve a more cost-effective defense in a specific theater than would have been achieved through a "monochromatic" architecture, based on only one type of defense system.

The benefits of interoperability do not, however, come without an associated cost. The main pitfalls to be considered are the following.

Complexity: Various applications of interoperability require connecting defense systems through elaborate communication protocols and also defining algorithms and interfaces. All of these factors complicate the structure and operation of the combined architecture as well as that of each of the participating systems.

Errors: This problem is related to complexity. Interoperability processes may induce errors that could otherwise be avoided. Examples include wrong correlation of tracks and wrong interception solutions. The effect of such errors, as well as their causes, will be analyzed in depth in the following sections of this chapter.

Load: Interoperability processes generate large amounts of data to be processed in the participating systems. If the added data are not treated wisely, the data may overflow critical components at certain points in time. An easy example is overflowing defense system radar with external cues because of the inability to correctly correlate tracks from several independent sensors. Another, less obvious example, is when a high-tier interception by one system clutters the sky-picture of a lower tier system with a large amount of debris, at the critical point in time when defense planning (possibly for a different threat) is taking place.

Dependency: Relying on systems that are not under immediate control of the ATBM defense planner to achieve a specific goal is risky because it may be that at the crucial moment these systems will not be made available to perform their assigned role. This may stem from political reasons, in cases where more than one nation is involved, but also from practical considerations regarding theater-specific situations. Examples include dual-use defense systems (e.g., air defense and missile defense, or ballistic missile and cruise missile defense) that may be diverted from their ballistic missile defense role according to the overall war situation.

Cost: Obviously, implementing interoperability between defense systems requires a certain amount of investment, which grows with the complexity of the chosen application. This added cost should be weighed against the expected benefits in overall cost-effectiveness assessments.

III. Levels of Interoperability

Interoperability between defense systems may be described as a continuum of possible connections, on one side of which lies the case of no connection. Each system operates on its own without regard to the existence of the other. On the other side is the case of full integration, where defense planning is done jointly, and each system is capable of using the resources of the others. To effectively analyze this spectrum within reasonable time and space, it is necessary to select a number of representative points along it and analyze each point in more detail. The points we selected are intended to highlight the growing capabilities of the resulting architecture. In each point (termed "level of interoperability") the BMD architecture gains the ability to perform an additional part of the defense process in a joint manner. Obviously, the level of complexity of the implementation increases as well.

The levels of interoperability are defined as follows: *level I*—no computerized connection (coordination may be attempted via voice channels); *level II*—single integrated air-picture achieved via computerized connections, external cueing mechanisms, and threat information sharing algorithms; *level III*—defense plan sharing by computerized means; *level IV*—defense planning/execution by one weapon system using data supplied by the other weapon system; and *level V*—full integration (centralized decision making, control, and guidance of the other system's interceptors).

This broad categorization leaves, of course, some options that do not exactly match any single category. An example is receiving external cueing without attempting to also integrate the sky-picture between the systems. In the sections to follow, we will address each level as containing the breadth of possibilities up to it. In the rest of this chapter, the various levels of interoperability will be analyzed through their influence on the defense process occurring in the combined architecture. The defense process is described in detail in a special chapter. A simplified definition is given below to form the background for the described analysis.

IV. Architecture Defense Process

The defense process starts offline, with the setting of defense goals (e.g., the assets to be defended, allowable leakage threshold), selection of deployment sites for the various elements, and coordination of search sectors and areas of responsibility between the systems comprising the architecture.

The online defense process may be described, using a rough approximation, as consisting of four major stages:

1) The air situation picture (ASP) describes the detection of objects and the classification, association, and maintenance of tracks.

2) Defense planning entails arriving at a suitable interception plan for each TBM deemed threatening to the architecture's area of responsibility. This stage includes coordination of the interception between different systems, if such coordination is required.

3) Interception is the execution of the defense plans arrived at in stage 2.

4) Debriefing concerns analyzing the results of past engagements and drawing conclusions for improving future conduct.

We will now proceed to discuss the effectiveness and implementation techniques of interoperability, in each one of the previously defined levels, at each of these stages.

V. Planning and Deployment

Planning and deployment are the first steps in any BMD operation. The deployment of defense assets is based on consideration of the assets to be defended, the allowable leakage rates, the characteristics of the expected threat, etc. The final outcome of this process is a set of decisions regarding the deployment of each system and sensor, primary target lines (PTLs) of the various sensors, and rules of engagement (ROE).

Interoperability with other systems not originally belonging to the architecture should also be considered in the planning process. The architecture's defense goals, deployment parameters and ROE may be changed according to considerations stemming from the possibility to operate in conjunction with other systems. The impact of interoperability depends on the type of connection (interoperability level) and the characteristics of the systems involved.

Some characteristic considerations may include the addition of a sensor to the architecture, whether as an external cueing source or as a sensor for a particular weapon system, which may influence the deployment of all other sensors in the architecture. A particularly easy-to-implement effect is a change in other sensors' PTLs to affect an optimal use of all sensor resources available.

Another characteristic is that defense goals may be changed because of added firepower (e.g., allowing the defense of otherwise nondefended assets).

If the implementation of the interoperability level selected calls for offline division of engagements (for example, the added system is to be given a separate defense sector), this may affect the deployment of other systems as well.

The level of expertise of the units that will be required to perform the real-time coordination and deconfliction tasks should be taken into consideration, and may cause a change in some units' deployment.

Because these considerations are mostly operational, rather than technical, and depend heavily on the specifics of the theater of operations and the particular defense architecture, they will not be developed here in greater detail. We turn our attention to more general considerations governing the real time operation of the architecture and the implementation of interoperability.

VI. Air Situation Picture

The first stage in the creation of an ASP is detection of objects. The contribution of peer systems to this process, through external cueing, is dealt with in a separate chapter. We shall, therefore, limit ourselves here to the discussion of the stages following detection, namely, classification and association. A special case, presenting more problems to the defense architecture, is that in which a TBM does not appear as a single entity, but is accompanied by related objects (a separating motor, decoys, debris from an earlier interception attempt, etc.). In this case, discrimination of the true target among the variety of objects in the air is also required.

Let us start our discussion in the case where all the objects in the air are legitimate targets for the BMD architecture (i.e. TBMs). The first problem facing the architecture is how to form a single integrated air picture (SIAP) in which each defense system is aware of all the tracks, or at least all the tracks relevant to it, and each system associates a common identifier to each track.

To analyze this issue in more depth, let us define two generic defense systems. The first, designated "system A" is an area defense (higher tier) system. It has a

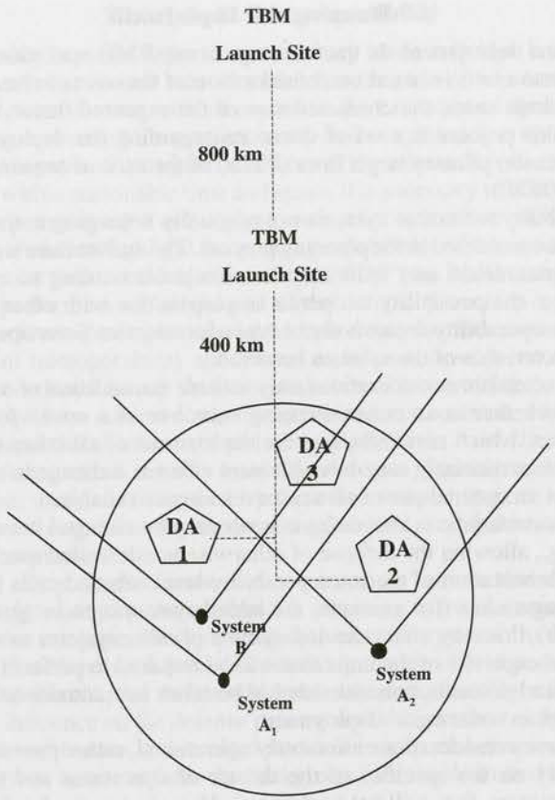

Fig. 1 Defense architecture layout.

radar that is capable of detection of TBMs at 700-km range (for simplicity, we assume a constant detection range). The second system, "system B," is a point defense (lower tier) system, with a detection range of 350 km. Both systems have a detection sector of ±45 deg in azimuth, and 0–85 deg in elevation (for the purpose of SIAP analysis, we need not bother with the definition of the matching interceptors). Our defense architecture is composed of two systems of type A and one system of type B, deployed to defend three defended assets (DAs), as depicted in Fig. 1.

One TBM launch site is located 800 km from the center of DA3. Let us assume a salvo of 30 TBMs, 10 aimed at each DA, according to the time line depicted in Table 1. TBM launches start at the designated time and a 2-s interval is assumed between consecutive launches. A second TBM launch site, capable of firing shorter range TBMs, is located 400 km from the center of DA3. This site will remain dormant in the present example.

Let us now look at the emerging ASP, when no attempt is made to correlate it between the participating defense systems. Figure 2 shows the number of tracks in each system as a function of time. Figure 3 shows the same data, for DA1 only. Looking at these graphs the problem of correct association presents itself clearly.

Table 1 Attack scenario

Time, s	TBM launch
0	Six at DA3 Three at DA2 One at DA1
60	One at DA3 Three at DA2 Six at DA1
120	Three at DA3 Four at DA2 Three at DA1

Each one of the defense systems is "seeing" a different sky-picture. In the case presented, we did not clutter the picture with interceptors, debris, and other objects. However, in real life the picture will be cluttered, and in particular when it comes to the picture observed by system B.

The first task of systems wishing to operate together is to unify the sky-picture. This could be attempted via voice channels (level I interoperability), but this would require extremely adept and well-trained personnel in both systems, and will be limited to relatively sparse attack scenarios.

The other alternative is to create an algorithm dedicated to associating tracks reported through interoperability. This process is called threat information sharing (TIS). A few remarks as to the implications of this process are in order, even before we turn to a discussion of the algorithm itself and its effectiveness.

First, each one of the defense systems involved in the process is routinely performing association tests to compare new tracks (initiated by its own sensors) to

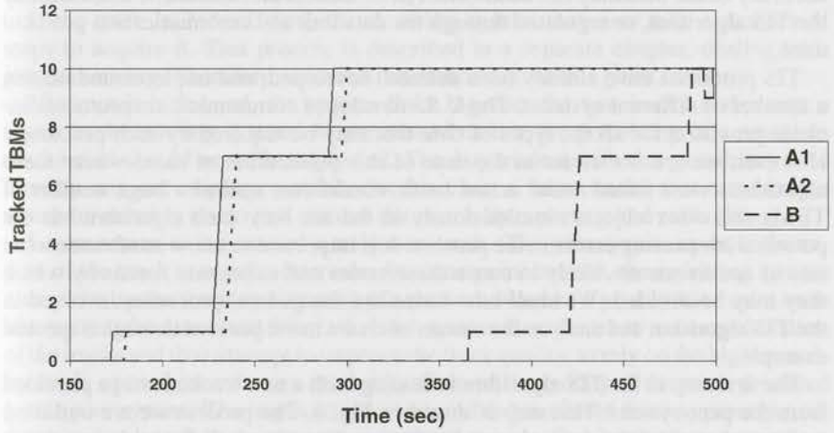

Fig. 2 Number of tracked TBMs vs Time: DA 1, 2, 3.

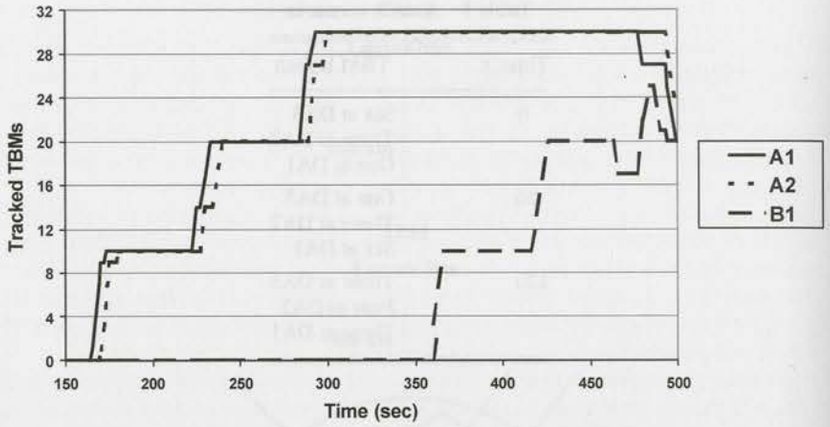

Fig. 3 Number of tracked TBMs vs Time: DA 1.

known system tracks. Problems may arise if the test used in a "joint" TIS process is different from that employed by the system in its regular operation (two tracks may be associated by one of the tests and not associated by the other).

On the other hand, because in the general case one cannot assume that the two systems in question employ identical association techniques (e.g., because they were produced by different countries), a common method—either one of the methods employed by the systems or a third algorithm—should be agreed on. Alternatively, the issue of how to deal with errors stemming from the use of different tests and algorithms, should be dealt with by the joint process.

A third point that should be taken into account is that data supplied by one system will not necessarily be in the same format used by the other. Not all of the data items required may be supplied, update frequencies may vary, and the accuracy of the data may not be known. All of these issues should be dealt with by the TIS algorithm, or regulated through the data link and communication protocol used.

TIS processes have already been defined, developed, and implemented among a number of different systems. The U.S.–developed communication protocols include provision for all the types of data that may be required by such processes. However, we are not aware at the time of this publication of cases where these algorithms were tested under actual battle conditions, against a large number of TBMs and other objects simultaneously in the air. Any such algorithm has the potential for causing errors in the process. It is important to know in advance what sets of conditions are likely to cause these errors and to prepare means by which they may be avoided. We shall now formulate the general processes involved in the TIS algorithm and analyze the causes of errors in the process through a specific example.

The first step in the TIS algorithm is dealing with a new track message provided from the peer system. This step is shown in Fig. 4. The process we are outlining assumes that all of the data relevant for the creation of a track file in the receiving system has been supplied. The receiving system first uses an extrapolation algorithm for each of its own tracks to bring it to a common time with the peer system

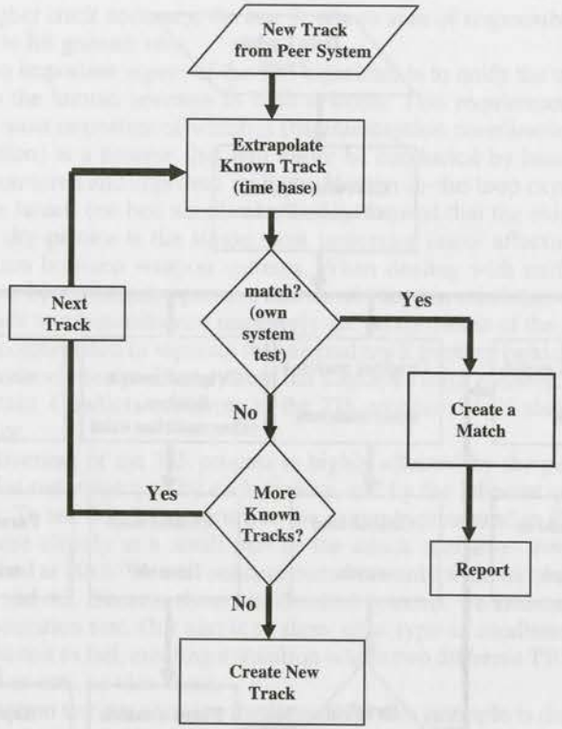

Fig. 4 TIS: dealing with a new track.

report, and then uses its own association test to find whether or not the two tracks are matched. If no match is found, the system creates a new track file.

If the track is, indeed, new, the receiving system may now take a number of steps to acquire it. This process is described in a separate chapter, dealing with external cueing. Therefore, we need not elaborate it at this point in time.

If a match is formed, the system links together the two tracks and issues a report to the sending system. As mentioned earlier, the possibility of a mistaken association should never be neglected. The way to deal with that possibility is to identify any out-of-place matches (in this case, a "new track" that turns out to match an existing track) and resolve the match by reporting and rechecking.

Let us return to the case of a truly new track. The peer system updates the report of the track at regular intervals. At some point in time, the receiving system also picks up the track and a match is achieved. Now, two possibilities exist as to how to treat the data about this track. Each system may choose to use a combination of the tracks and thus attempt to improve the track quality, to rely on the higher quality track, or to choose its own track regardless of the quality of the track provided by the peer system. The implications and possible benefits of these approaches are discussed at length in the chapter on external cueing. One important observation, though, is that to avoid further complication of an already complex procedure, both systems should agree on the same procedure for choosing the data to be used.

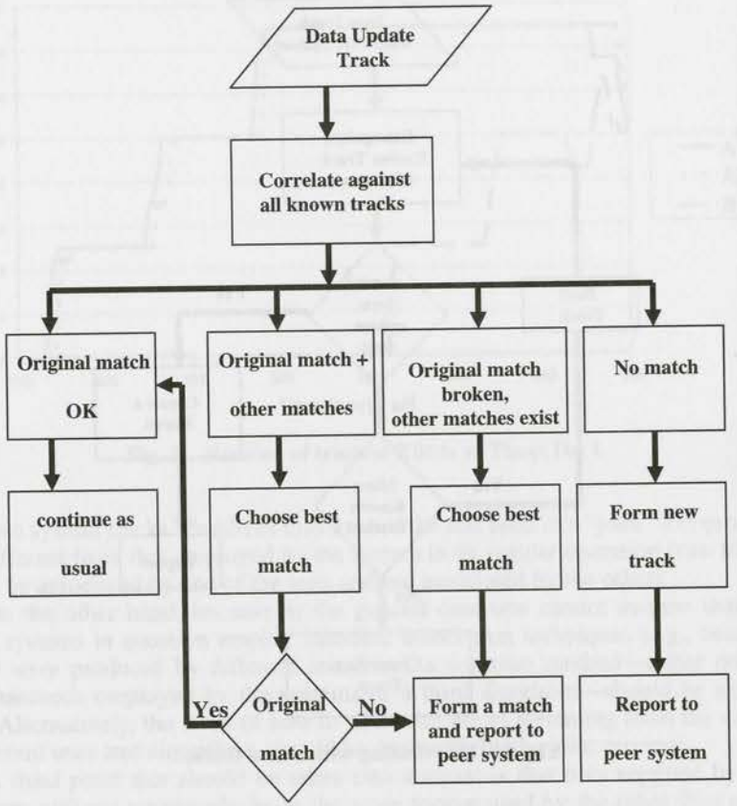

Fig. 5 TIS: update message.

Upon receiving an update to an existing track, no matter if it is provided by the peer system or by own system sensors, a correlation check should be performed to determine if the original match was broken or if another one is formed. Several outcomes are possible from this process, as depicted in Fig. 5: 1) the original match is kept, 2) the original match is kept but another track also matches the updated data, 3) the original match is broken and no existing track matches the updated data, and 4) the original match is broken and one or more new matches are formed. All of these cases except the first, represent anomalies that should be treated in coordination with the other system. In any such case, appropriate precautions should be taken to avoid a process of constantly breaking and reestablishing matches (e.g., not declaring a break of a match before several consecutive updates indicate it, and setting appropriate thresholds for association tests).

Two further comments need to be made about the nature of the TIS process, before we turn to analyzing its effectiveness. First, the process of resolving conflicts between the ASP observed by the interoperating systems has to include a method for taking preference in case an agreed upon match is not achieved. Such a mechanism may choose to prefer the system that first reported the TBM, the one

reporting higher track accuracy, the one in whose area of responsibility the TBM is supposed to hit ground, etc.

Second, an important aspect of the TIS algorithm is to unify the track identifier presented to the human operator in both systems. This requirement has several reasons, the most important of which is that interception coordination (see also in the next section) is a process that will either be conducted by human operators, or at least monitored and approved by them. Human-in-the-loop experiments conducted in the Israeli test bed simulation facility showed that the existence or lack of a unified sky-picture is the single most important factor affecting the quality of coordination between weapon systems. When dealing with unifying the sky-picture as seen by the human operator, one should keep in mind that changes in the displayed track number influence negatively the performance of the operator. It is, therefore, recommended to separate the internal track number (which may change as a result of association problems) from the displayed track number, which should remain constant. Conflicts occurring in the TIS process should also be presented to the operator.

The effectiveness of the TIS process is highly affected by the performance of the association test employed by each system, and by the inherent qualities of the sensors used. To see this, let us return to the example presented in Fig. 1. We will now look more closely at a small part of the attack scenario—two consecutive TBMs launched at DA3. We will concern ourselves only with the two area defense systems, A1 and A2. Because these are identical systems, we assume that they use the same association test. Our aim is to show what type of conditions cause such an association test to fail, creating a situation where two different TBM tracks may be processed as one, or vice versa.

The association test we chose to implement for this example is the χ^2 test. The χ^2 statistic is applied to the difference between the two state vectors in position and velocity. If we designate ΔX as the difference along the X axis between the two vectors, ΔV_X as the difference in the velocity along the X axis, and likewise σX as the standard deviation in X, etc., then χ^2 is calculated by

$$\chi^2 = \frac{(\Delta X)^2}{(\sigma_x)^2} + \frac{(\Delta Y)^2}{(\sigma_Y)^2} + \cdots + \frac{(\Delta V_x)^2}{(\sigma V_x)^2} + \cdots + \frac{\Delta X \Delta V_x}{\sigma (X V_x)^2} + \cdots \qquad (1)$$

The χ^2 statistic is applied on the Cartesian state vectors (x,y,z,v_x,v_y,v_z) containing each radar's measurements. A value of $\chi^2 > 12.6$ indicates a 95% confidence that the two objects tracked are different.[2]

Let us look now at two TBMs as already described, launched at exactly the same time, and let us assume they are separated by a certain distance X (one case) or 5.5 X (second case) along the X axis. Figure 6 shows the χ^2 in both cases. As can be seen from the graphs, the test achieves clear discrimination when the two TBMs are widely spaced, but fails to separate them when the distance between the launch points is only X meters. Figure 7 shows the same analysis for two TBMs launched from exactly the same location, but with a time difference. The graph shows that the test fails only in cases of a very small separation dt.

These two generic examples point to the fact that closely spaced objects in time and/or space may mislead the TIS algorithm. The actual conditions under which a separation test is misled will depend on the nature of the test itself and the qualities, in particular in terms of the signal to noise ratio S/N, of the participating radars. Analysis of the type shown in Figs. 6 and 7 should be conducted in advance so

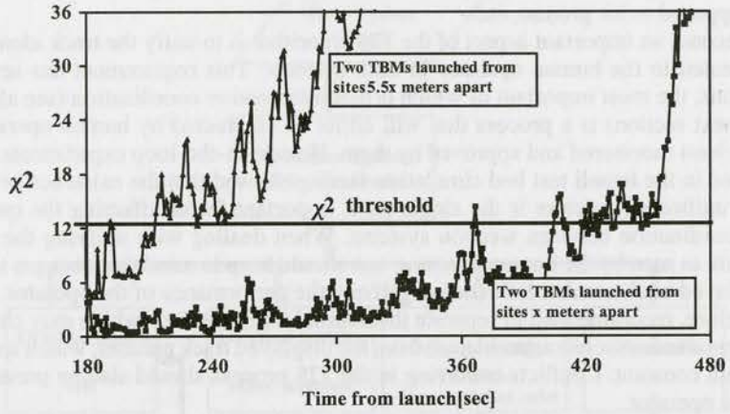

Fig. 6 χ^2 **Analysis launch location difference.**

that the limitations of the particular implementation chosen are well understood. Before leaving the χ^2 example, let us explore one more point, which was only briefly mentioned earlier. The issue is that of not knowing exactly what is the quality of the track provided by the peer system. In particular, the biases involved in the tracking of the other system may not be known. A quasi-linear model was used to simulate biases. The model is summarized by the following formula[3]:

$$\delta R = c_R \tag{2}$$

$$\delta A = c_1 \tan(E)\sin(A - c_2) + c_3 \tan(E) + \frac{c_4}{\cos(E)} + c_5 \sin(A - c_6) + c_7 \tag{3}$$

$$\delta E = c_1 \cos(A - c_2) + c_8 \cos(E) + c_9 \sin(E - c_{10}) + c_{11} \tag{4}$$

where A and E are the azimuth and elevation as measured by the radar; δR, δA, δE

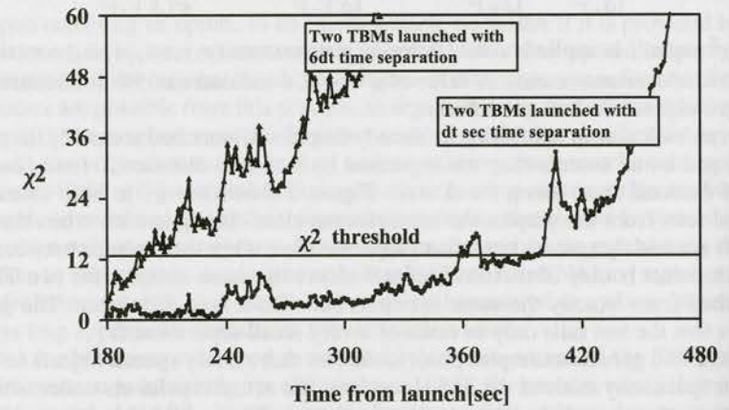

Fig. 7 χ^2 **Analysis: launch time difference.**

Table 2 Bias model coefficients

Notation	Error
c_1	Leveling error amplitude
c_2	Leveling error phase
c_3	Orthogonality error
c_4	Colimation error
c_5	Azimuth eccentricity error amplitude
c_6	Azimuth eccentricity error phase
c_7	Azimuth offset
c_8	Droop
c_9	Elevation eccentricity error amplitude
c_{10}	Elevation eccentricity error phase
c_{11}	Elevation offset
c_R	Range measurement error

are the biases in range, azimuth, and elevation; and the coefficients represent different errors according to Table 2.

Figure 8 presents a case where a single TBM is viewed by the radar of systems $A1$ and $A2$. $A1$ has no bias, but $A2$ reports are corrupted by severe biases. The result is that up to 240 s before impact, the TBM is regarded as two separate objects. This phenomenon has severe implications, which may include waste of resources (cueing A1's radar, for example), and may even lead to unnecessary launch of interceptors. To solve this problem, we use the fact that the development of the difference between the vectors in time is much less sensitive to biased measurements (because they are nullified by the subtraction) than the original state vectors.

We turn, therefore, to test the time derivative of $\chi^2 - \mathrm{d}\chi^2/\mathrm{d}t$. In Fig. 9 the results of $\mathrm{d}\chi^2/\mathrm{d}t$ are shown for the same measurements that were used for Fig. 8. The result is a clear indication that there is only one object in the sky.

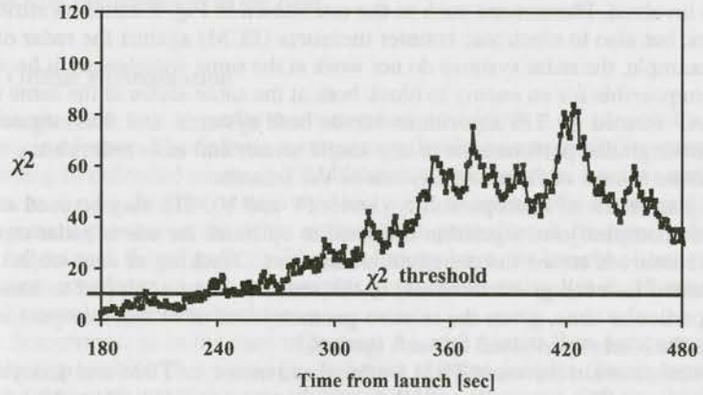

Fig. 8 χ^2 **Analysis: one TBM in presence of severe bias.**

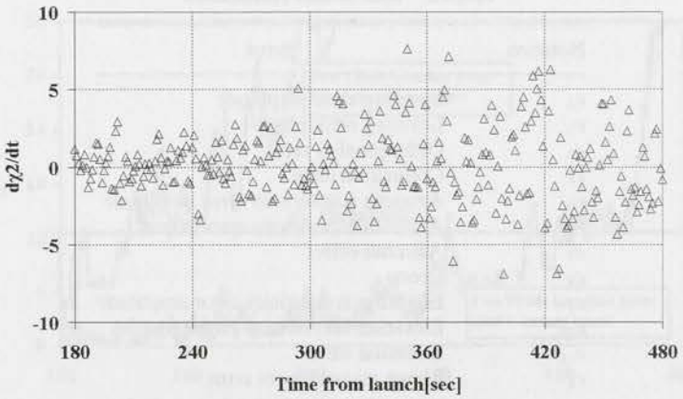

Fig. 9 Time derivative of χ^2: severe bias.

Because the nature of the peer system with which interoperability will be practiced, and the conditions under which it will have to function, are not known in advance, the planners of TIS algorithms should apply considerations akin to those outlined previously to arrive at association tests that are as robust as possible to biases, and to other system errors (e.g., false alarms).

Up to now, we explored the need for SIAP via automatic TIS algorithms, as a cornerstone for interoperability of level II and above. However, it should also be noted that SIAP contains, by itself, several possible advantages, even when it is not considered as part of the coordination of interceptions. The possible uses and advantages of these procedures include the following.

1) External cueing. A separate chapter in this book is dedicated to cueing and its implications.

2) Robustness. Correlating and integrating the sky-picture between two or more systems carries the advantage of being less sensitive to problems in any one of the sensors involved. Phenomena such as the one shown in Fig. 8 could be attributed to biases, but also to electronic counter measures (ECM) against the radar of A2. If, for example, the radar systems do not work at the same wavelength, it becomes almost impossible for an enemy to block both at the same sector at the same time. The SIAP formed by TIS algorithms serves both systems, and thus negates the effects of degraded performance in any single sensor and adds redundancy to the architecture in case of failure of any one of the sensors.

3) Higher levels of interoperability (levels IV and V). TIS may be used as part of a more complex joint algorithm designed to optimize the use of radar or other sensors resources across the complete architecture. Tracking of each object may be entrusted (at each given moment) to the sensor(s) best equipped to handle it at that particular time, given the relative geometry and other relevant parameters (such as the load experienced by each system).

4) Discrimination (between TBM warhead and motor, or TBM and decoys) and kill assessment. This use requires high levels of interoperability (IV or V) because of the strict accuracy and time line requirements involved in these processes.

VII. Interception Coordination

Following the creation and understanding of the sky-picture, the next stage in the defense process is defense planning. In this stage a specific plan of interception, for each TBM threatening the designated defended assets (DAs) is to be reached. The defense plan (DP) consists of a combination of tracking and firing entities (radar and fire units) and their activation time. If more than one planning entity (e.g. battle management center) is included in the architecture, then deconfliction, or coordination, of the interception has to take place. This is also the case where interoperability with systems outside the architecture is concerned.

Let us return now to the attack scenario example defined in Table 1. Assuming that no effort was made to coordinate the interceptions between the defense systems, and that each system launches one interceptor vs any TBM threatening a DA in its footprint, this 30 TBM scenario would result in a launch of 50 (assuming system B does not defend DA3) to 60 (in case it does) interceptors. This, when in theory 30 interceptors would suffice to provide an adequate level of defense (assuming that one interceptor per TBM is enough to provide the required level of defense). This high ratio of interceptors/TBMs is clearly unacceptable. At the very least, it is unacceptable not to be able to influence it when the need arises. Obviously, in almost any situation involving the deployment of several weapon systems, coordination of the interception tasks will offer considerable benefits.

Coordination of interceptions can take place at any of the previously-defined levels of interoperability. It can take the form of drawing a clear dividing line between the areas of responsibility of any two systems and then act according to that division at each system (level I interoperability), or of actively performing coordination of each TBM answering to specific characteristics either by voice (level II interoperability) or via computerized algorithms (level III). It may even take the form of a dual-system defense plan in which a lower tier system serves as a second layer to an upper tier one (levels III–IV). The most extreme version may take the form of central planning of interceptions, drawing upon a "bank" of available resources from the involved systems (level V). We will discuss each of these options in turn, focusing most of the analysis on verbal and computerized coordination, as these are, from our point of view, the more probable versions to be actually implemented in the field.

A. Offline Coordination

The simplest way to perform interception coordination is, obviously, to do it offline, in advance. The defense systems may divide the responsibility among them according to defended assets or TBM launch sites/countries. Another option is to allow, as a rule, the first system that has a DP to engage the target and halt the engagement process in the other system (this procedure is frequently described as a "shoot and shout" policy). These solutions have the benefit of not requiring real-time coordination and the complex processes involved in it. However, they entail several major drawbacks:

1) Sometimes, as in the case of system A1 and system B in our example, it is simply impossible. This situation, in which a point defense (lower tier) system will be deployed inside the area of responsibility of an area defense (higher tier) system, is a very probable one in real deployment.

2) Defense systems vary in their capabilities. System B for example, which is a less capable, smaller footprint system than system A, may not be able to defend against longer range, faster TBMs, although it may perform as well as, or even better than, system A against shorter range TBMs. Dividing responsibility according to simplistic area division may thus lead to a situation in which the provided defense is not adequate.

3) Also, one should keep in mind that the information about a track is different in different systems. Some factors, such as the nature of the (suspected) TBM warhead, or even the type of TBM, may be calculated (or derived from intelligence information) by one system and not known in the other. These factors, although they may be important for differentiating between the two systems' ability to intercept a specific threat, cannot be used for a priori differentiation, but only in real time, during an actual engagement.

4) Finally, this type of "offline coordination" fails to provide a lot of the benefits already described, which an architecture planner would like to achieve through interoperability. Having two systems capable of defending a common region allows for mutual backup, alleviating load problems from any single system and optimizing the defense on a track to track basis or for a whole TBM salvo. All of these benefits cannot be achieved using offline, strict division of labor.

For the reasons just outlined, we consider coordination by offline decisions as an inferior option to real-time coordination.

B. Real-Time Coordination

Real-time coordination may be achieved at various levels of interoperability. It may be applied by voice channels (level II), or by automatic defense plan sharing (DPS) algorithms (levels III and above). In the following discussion, we will attempt to identify the major issues affecting defense coordination, and also demonstrate its effectiveness in various situations. Finally, we will discuss the causes and effects of coordination errors, and possible ways to avoid them.

C. Time Line for Coordination

The first issue to be considered regarding defense plan coordination at all levels is that of the available time. To understand this problem more clearly, and to get a hint of its magnitude, let us return to our ongoing example from Fig. 1.

Because we are now dealing with the interception of TBMs, we have to introduce into the example the qualities of the defense planning process. As we are only interested in this chapter in the effects of the coordination between the systems, we will allow ourselves to use a very simplistic model for these qualities.

In our model the interceptors of both systems are flying a straight line to a chosen interception point, at a constant velocity. The boundaries of the available battlespace for each system are fixed in range and altitude. For the moment, we will not bother with the interceptor's P_k. The only defense planning parameter that we introduce is a requirement that the TBM be tracked for a certain period of time before a defense plan can be produced. The parameters of the two types of defense systems are presented in Table 3. These numbers do not represent any known defense system. However, they are representative numbers that may be used to draw initial conclusions, relevant to the actual situations an architecture planner might encounter.

Table 3 Defense systems characteristics (example)

System type	Interceptor velocity, m/s	Interception altitude, km		Interception range, km		Required tracking time, s
		Min	Max	Min	Max	
A	1500	20	100	5	300	45
B	1200	5	30	5	50	15

Let us now define what we mean by "the time for coordination." To do this, let us look at Fig. 10. In this figure, a TBM is launched at time t_0, and is supposed to hit ground at the attacked target at time t_{end}. The first system to detect the TBM, system 1, detects it at t_1; this system takes n_1 s to plan a DP, which is ready at time t_3. The DP of system 1 has to be executed at t_{11}. Meanwhile, a second system (system 2) detects the TBM at time t_2. This system now takes n_2 s to plan a DP of its own, at time t_4. The time available for coordination between the two systems is the time between t_4 (system 2 knows whether it can intercept this TBM and with what parameters) and t_{11} (system 1 has to activate its DP). In other words,

$$t_{\text{co}} = t_{11} - t_4 \tag{5}$$

where t_{co} is the time for coordination.

Clearly, the more time given for coordination, the better. As can easily be seen from Eq. (5), the only way that the systems involved in the process can influence the time available (other than using external cueing to achieve earlier detection), is by changing t_{11} (t_2 is determined by the radar qualities and the TBM trajectory,

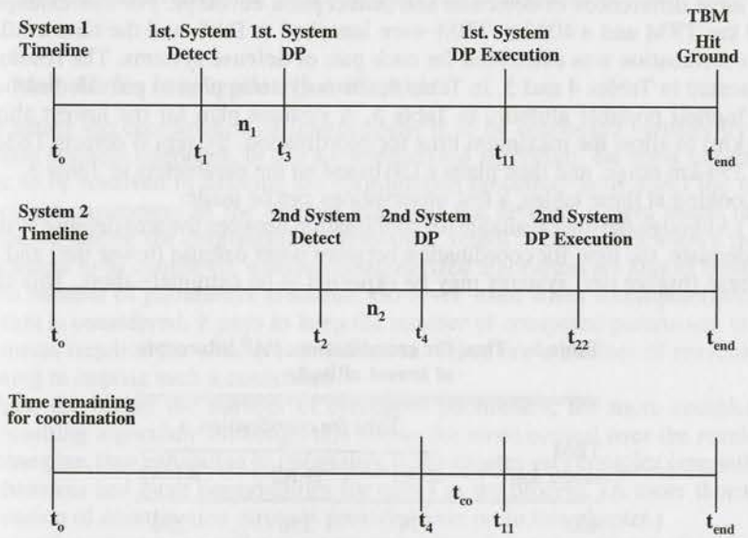

Fig. 10 Time for coordination.

Table 4 Time for coordination: "A" intercepts at highest altitude

TBM range, km	Time for coordination, s		
	B-A1	B-A2	A1-A2
400	87	72	155
800	16	−1	155

and n_2 is a system parameter). Time t_{11} is affected, of course, by t_1 and n_2, which cannot be changed. However, it is also affected by the choice of interception point. In our simple example, t_{11} is directly proportional to the altitude of interception because if the interception is to take place at altitude $\boldsymbol{a}$ and range $\boldsymbol{r}$ (from the launcher), then

$$t_{11} = \frac{\sqrt{a^2 + r^2}}{V_{\text{interceptor}}} \tag{6}$$

where $\boldsymbol{a}$ and $\boldsymbol{r}$ are related through the TBM trajectory. This relation between the interceptor launch time and the chosen interception altitude may be used to maximize the available time for coordination, by choosing the lowest possible interception altitude.

Let us look at the concrete example of our three systems: A1, A2, and B. Clearly, since A1 and A2 are identical systems, they detect the TBMs at nearly the same time, and the time to coordinate is almost identical to the time from end of defense planning to interceptor launch in any one of the systems. The situation is drastically different between any one of the A-type systems and system B, because of the large differences in detection and interception envelope. For this example, a 800 km TBM and a 400 km TBM were launched to DA1, and the time available for coordination was calculated for each pair of defense systems. The results are presented in Tables 4 and 5. In Table 4, the A systems plan to provide defense at the highest possible altitude. In Table 5, A systems plan for the lowest altitude (20 km) to allow the maximum time for coordination. System B detects TBMs at the 350-km range, and then plans a DP based on the parameters in Table 3.

Looking at these tables, a few observations can be made:

1) Although the time available for coordination between the area defense systems is adequate, the time for coordination between point defense (lower tier) and area defense (higher tier) systems may be expected to be extremely short. This effect

Table 5 Time for coordination: "A" intercepts at lowest altitude

TBM range, km	Time for coordination, s		
	B-A1	B-A2	A1-A2
400	179	156	230
800	88	66	221

grows in magnitude for longer range, fast TBMs. In some deployments, there may be negative time for coordination (one system has to launch an interceptor before the other picked up the target).

2) The relative deployment geometry between the coordinating systems and the attacked asset affects the interception solution (DP) adopted by each system, and thus the resulting time for coordination. In our example, we chose a relatively small deployment area. However, in real life the deployment area may be much larger. An example is the case where one of the systems is a naval system, which has to deploy at a safe distance from the shore.

3) Lowering the interception altitude increases the time available for coordination. However, one must keep in mind that this practice is detrimental from the point of view of defense quality. For most systems, going to lower altitude means a loss of P_k, and the ability to perform shoot-look-shoot processes is sometimes lost, as well as the ability to compensate for any unexpected problems during the interception process.

The question of how to conduct the coordination of interceptions—by voice or by computerized process—now becomes a question of the nature of the defense systems performing the coordination, the chosen deployment, and the expected threat. For humans to conduct verbal coordination, tens of seconds are required. This means that if only voice coordination is available, the defense planner must use a mix of predefined rules of engagement, and various means to increase the available time such as lowering interception altitude, cueing the lower tier system, or attempting to predict its ability to intercept a given TBM before it has actually detected it. One should also keep in mind that verbal coordination entails other problems, mostly concerning the operation of the human-operator in the coordination loop. Problems relating to language barriers, different conceptions of defense plan quality, different command structure, and even the simple fact that the two systems do not display the same set of data items to their operators, have to be discussed and resolved before such a solution can be considered.

D. Methods for Interception Coordination

Whether the coordination of interceptions is to be conducted verbally by the defense systems operators, or via a computerized algorithm, the most important issue to be resolved in defining the coordination procedure is to select the most important parameters of the defense plan and the defense system status to be compared during the coordination. Obviously, verbal coordination allows fewer parameters to be compared, whereas computerized algorithms are almost limitless in the number of parameters available. However, even when a computerized algorithm is considered, it pays to keep the number of compared parameters to the minimum required for effective coordination. There are a number of reasons for wanting to impose such a constraint:

First, the higher the number of compared parameters, the more complex is the resulting algorithm. Although this allows for more control over the resulting defense plan, thus enhancing its optimality, it also creates very complex comparison mechanisms and more opportunities for errors in the process. (A more thorough discussion of coordination errors is provided later on in this chapter.)

Second, current communication protocols allow for automatic transfer of only a few of the parameters associated with a defense plan. It is possible to design and

implement special links and protocols, but the outcome has to be weighed in terms of improvement over what can be achieved using a smaller number of parameters.

Third, in planning coordination algorithms to be implemented in any defense system, one should keep in mind that the partner system for interoperability is not necessarily decided on in advance. This is especially true for systems that are planned to be deployed for defense of forward-deployed troops or as part of a coalition in different theaters. Not all of the defense systems with which interoperability may be enacted use the same parameters for building and evaluating defense plans. Thus, using many parameters as part of the coordination algorithm may increase its theoretical effectiveness, while at the same time harming its real world usefulness.

In this section we will try to quantify the contribution of various methods of coordination to the results achieved by the antitactical ballistic missile architecture. We will focus the discussion on coordination using three parameters, which we consider to be the most important for defense planning.

1) *Probability of kill (P_k):* This parameter (together with the number of interceptors to be used) indicates the quality of the defense plan as a singe entity (without taking into account cumulative effects of any kind). Other parameters could also be taken as representing DP quality (e.g., interception altitude, maneuverability ratio) but we believe this is the most representative.

2) *Inventory:* The number of remaining interceptors at the time of planning is reflected here. This number gives no indication of DP quality, but is indicative of the possibility of future problems the defense system may face through interceptor shortage.

3) *Load on defense system assets (mainly the radar):* This number gives an indication of possible problems that the defense plan may face when the time to execute it arrives. The more loaded the system, the more chances that the interceptor launch will be delayed, or even cancelled, because of the need to perform higher priority tasks (e.g., interception of TBMs aimed at higher priority DAs). In this chapter we will be using the immediate load at planning time as an indicator. It is possible to perform a prediction of expected load at the time designated for interceptor launch, either using the data available in the system itself, or using external cue data, information regarding future events. It remains to be seen whether these more complex considerations are justified in the context of coordination between weapon systems. (For more information on these topics, the reader is referred to Chapter 9.)

As can be seen from these definitions, the parameters to be compared describe different aspects of defense system operation. Because of that, they are seen as complementary, allowing for maximal added value of each one to the overall defense effectiveness.

To quantify the contribution of each of these parameters to the effectiveness of the coordination function, we return now to the example presented in Fig. 1. We will deal in the following examples only with coordination between the two area defense systems, A1 and A2. Notice that for A1 and A2, the defended space may be divided into three subareas 1) S_1, which includes DA1, where only A1 can provide defense (disregard, for this example, system B), 2) S_2, which includes DA2, where only A2, can provide defense, and 3) S_{12}, which includes DA3, where both systems can provide defense. This quality will have an important role in our example. (The fact that such a division is possible is trivial. The fact that each subarea contains

Table 6 Reference P_k table (example)

Defense system	DA	TBM range 400 km	TBM range 800 km
A1	1	0.86	0.84
	2	0	0
	3	0.83	0.78
A2	1	0	0
	2	0.86	0.84
	3	0.85	0.8

exactly one defended asset, which is separated from the other DAs, is a simplifying assumption). Furthermore, we will make another simplifying assumption: TBMs are only fired at the designated assets. That is to say, no TBM is fired at a point that lies between the designated DAs, and no TBM misses its intended target (through statistical dispersions and CEP). We will now introduce P_k values for our two defense systems, to be used while coordinating the defense. Table 6 presents P_k for each system against TBMs of the two types presented earlier, in defense of each of the three DAs. Notice that while the two systems are identical, P_k for each asset is different. Also notice that we assign to each system zero P_k in the domain of responsibility of the other. The P_k given by the table is the single shot probability of kill (SSP_k).

The scenario consisted of 30 TBMs of 400 km range, and 30 TBMs of 800 km range. Out of these 60 TBMs, 40 were aimed at DA3 (the asset over which coordination is performed) and the other 20 were evenly divided between DA1 and DA2. Let us look now at the effect of various methods of coordination on the results achieved by the combined architecture vs this attack scenario.

Figure 11 shows the number of leakers suffered by the architecture as a function of the initial interceptor inventory in each system (the number of interceptors is identical in both systems). Two graphs are shown. In the first graph, no coordination was conducted. Each system fired at each TBM, provided it had a nonzero P_k, until no more interceptors remained. In the second graph, coordination by P_k was

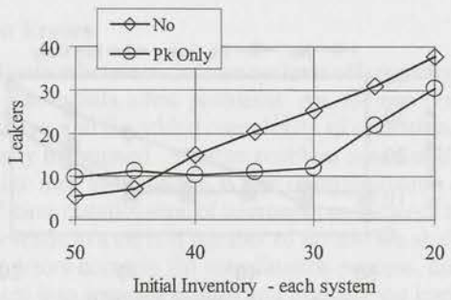

Fig. 11 Coordination by P_k vs "no coordination."

conducted on the TBMs aimed at DA3. The system with higher P_k was chosen, unless it was out of interceptors.

When the architecture has a large number of interceptors (on the left side of the chart) we can see that the results (expressed by the number of leakers) are better if no coordination at all is performed. This is simply because the P_k for each interception over DA3 is improved (two interceptors instead of one). However, this is achieved at a price of launching 100 interceptors (both systems fire at the 40 TBMs aimed at S_{12}, and 20 TBMs are intercepted by one system only) instead of 60 in the case that coordination is performed.

As we move to the right of this figure the results of the "no coordination" case deteriorate quickly, because the interceptors are depleted before the end of the attack. The coordination by P_k scheme, on the other hand, achieves almost constant results until the point where there are fewer interceptors than TBMs, and only then its performance is degraded. These results show clearly the benefits that the architecture may derive from even the crudest form of coordination. The resulting architecture performs in a much more robust fashion, enabling the defended nation to counter different attack plans and scenarios. However, there is also another way of looking at this graph. If we subtract one line from the other we can immediately get the maximal benefit of interoperability, expressed in terms of consumed interceptors. There is a tradeoff between acquiring more interceptors and conducting interoperability, and there is a limit to the investment that should be made in interoperability. Sometimes, the "rich man's solution" of simply adding more firepower, can turn out to be less costly. Our simple example is, of course, too crude to be used as a "meter" for studying the cost-effectiveness of various forms of interoperability. Cost-effectiveness considerations should be applied on the basis of intimate knowledge of the two systems to be connected, as well as a detailed description (as much as possible) of the foreseen threat.

Let us now add to our example another form of coordination. This time the systems coordinate according to the immediate inventory; i.e., the system with more available interceptors is the one to fire. The results of this form of coordination appear in Fig. 12, in addition to the "no coordination" and "coordination by P_k" cases. Looking at this graph, a few observations can be made:

First, the major gain from interoperability between the systems is achieved by the mere fact that coordination is enacted. The actual parameter by which coordination is conducted is less important. Second, the "obvious" parameter for judging the

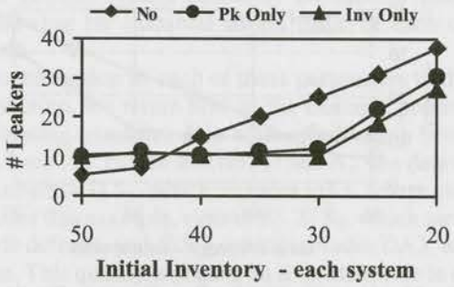

Fig. 12 Coordination by inventory.

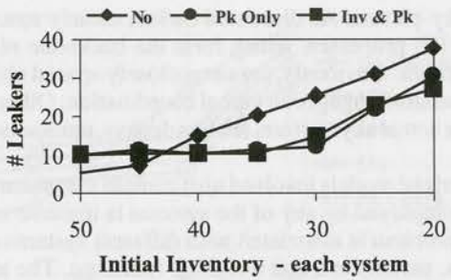

Fig. 13 Coordination by P_k and inventory.

quality of defense plans is not always the best. In this example, where the system's P_k values are close one to the other, coordination by inventory manages to achieve the same level of performance as coordination by P_k, and it is even slightly more robust toward the right side of the chart, where inventory levels are low. This occurs because coordination by P_k only may cause situations in which one system becomes short of interceptors, whereas the other still has a sufficient number. If TBMs are launched to the area where only the first system can engage, the result will be a large number of leakers.

To gain the best of both worlds, a combined coordination rule is introduced and plotted in Fig. 13 alongside the "no coordination" and "coordination by P_k" graphs. In this mode or coordination, the system with more interceptors launches, unless its P_k is lower than a predefined threshold value (in our example -0.79). The results indicate, again, achievement of equivalent level of performance to that of coordination by P_k only, but in a more robust manner.

There are, of course, many other parameters that may be considered as candidates for supplying a basis for coordination between defense systems. The most important one, in our view, is the load experienced by each system. The load on the system could be taken as the instantaneous load at the time of defense planning, or, using external cue data, predicted load at the time designated for interceptor launch. (For further discussion of this topic see Chapter 9.) Load may be factored into the coordination procedure using a threshold mechanism similar in nature to the one already introduced for inventory and P_k coordination.

E. Coordination Errors

Although coordination between defense systems offers many advantages, as was already shown, it also entails a few problems. An obvious problem is that of the cost of implementation and the added complexity of operation, as well as the timing problems already mentioned. Another problem associated with coordination, which we would like now to highlight, is that of coordination errors.

Conducting real-time coordination of interceptions between two or more weapon systems inevitably leads to a certain number of errors. We shall first discuss some of the reasons that errors occur in the coordination process, and then we will proceed to classify them into separate groups and evaluate the impact of various error types on the performance of the combined architecture. Errors in the coordination process stem from several different sources.

1) The actual sky-picture. As discussed earlier, closely spaced objects may induce mistakes in TIS processes, which form the backbone of any computerized coordination algorithm. Obviously, the same closely spaced objects also may confuse the human operators engaged in verbal coordination. Other forms of problems associated with the actual sky-picture, such as decoys, remnants, or ECM, will have a similar effect.

2) The computerized models involved also contain the potential for errors. None of the algorithms employed by any of the systems is immune to problems. Nevertheless, the main problem is associated with different systems employing slightly different measures, parameters, and weighing functions. The interpretation of the various parameters exchanged may be different, and some parameters needed by one system in defense planning may not be considered at all by the other. Errors will generally increase with the complexity of the chosen algorithms.

3) Human operators in the two systems are not necessarily thinking in the same terms. In the general case, they may not even be speaking the same language. The goals of the defense architecture, as well as its ROE and CSOP (combined standing operating procedures), may not be fully agreed upon in advance, especially in cases where defense is conducted by a multinational architecture.

4) Last, but not least, is the time line issue, which we discussed earlier in this chapter. Interoperability errors are one of the manifestations of this problem. In this respect, it is also important to remember the physical links and communication protocols involved, which also contribute to the timing problems and also introduce errors in the data transmitted.

Let us now turn to the classification of coordination errors and assessment of their impact on architecture performance. We find that the most convenient (albeit, not the only) way to classify coordination errors is by their result in terms of a specific interception. According to this approach, coordination errors may be divided into three groups.

1) No interception: Situations where each defense system assumed that the responsibility to intercept a specific TBM rests with the other system. The result is that none of them engages the TBM, which leaks through.

2) Double interception: A case where both systems assume they are responsible for intercepting the TBM, and both fire at it. The result is a waste of interceptors, while at the same time also increasing the P_k for that specific intercept.

3) Wrong interception: A case where a correct interpretation of prevailing concept of operations (CONOPS) and ROE would mean that one system should engage, but the coordination process assigns the interception to the other system. This is the most elusive type of error. The result in terms of architecture performance depends on the set of parameters by which coordination is conducted, as well as on the specific systems used.

We shall now turn to a simple demonstration of the way in which coordination errors affect architecture performance, and use again our previous example, of the coordination between systems A1 and A2. We keep the same attack scenario, and, for the sake of simplicity, conduct coordination by P_k only. Figure 14 shows the original graphs for "no coordination" and coordination by P_k, together with three graphs representing various levels of "no interception" errors.

Looking at this graph, the following observations can be made: First, errors obviously diminish the cost-effectiveness margin of coordination. At some point (which may be even at low error values), they may even lead to a case where it is not worthwhile to conduct coordination between the systems at all. Second, the general effect of "no interception" errors is to "lift" the coordination graph while retaining its

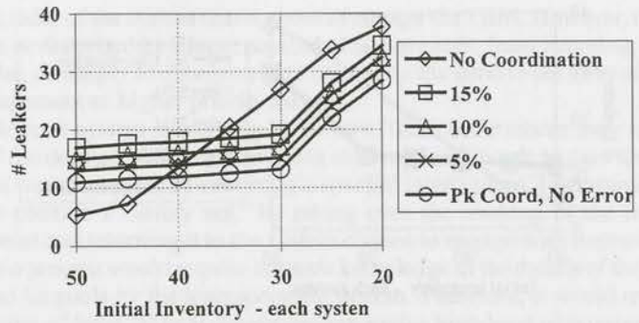

Fig. 14 "No interception" errors.

shape. This is because this type of error results in a direct increase in the number of leakers. Third, this type of error is severe from the point of view of the architecture planner, since low levels of errors result in significant performance degradation.

Figure 15 shows the same graphs for different values of "double interception" errors. Again, some observations are in order. First, at high levels of interceptor stock (the left side of the graph), this type of error actually improves the architecture performance, because it improves the P_k for specific intercepts. Second, as the number of interceptors available decreases relative to the attack size, double interceptions are punished by early depletion of interceptors and, as a result, more leakers. Thus, a rapid decrease in architecture performance is observed. Third, the sensitivity to this type of error (the percent of errors that cause significant decrease in performance) depends on the ratio between the available inventory and the (expected) attack size. Generally, it may be stated that this type of error is less severe from the architecture point of view than the "no coordination" errors.

Finally, Fig. 16 illustrates the effect of "wrong interception" errors. This effect is marginal in our example, because of the small difference in the P_k of the two systems. However, this type of coordination error should not be neglected in decision-making processes. It should be evaluated according to the actual parameters of the defense systems involved, the exact coordination scheme contemplated, and the envisioned threat.

Coordination errors will be a part of any process, be it verbal or automated. However, their presence may be controlled and contained by careful consideration at the definition phase, and constant training of the relevant personnel. (The issue

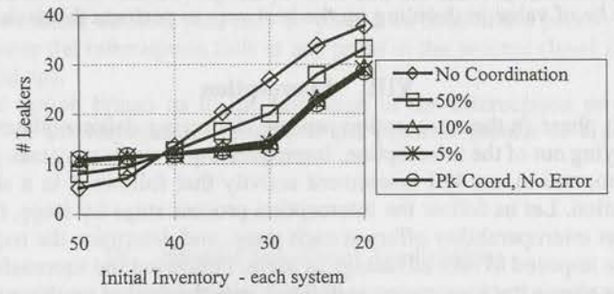

Fig. 15 "Double interception" errors.

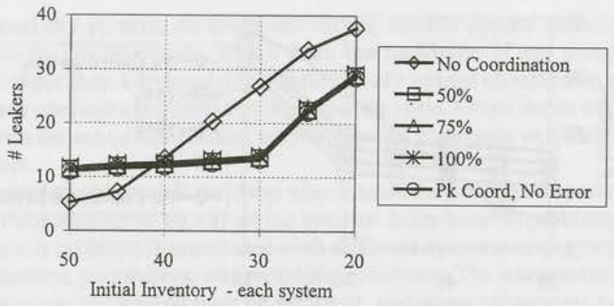

Fig. 16 "Wrong interception" errors.

of training for interoperability will be developed further toward the end of this chapter.) Computerized processes, such as TIS algorithms, may be tuned so that a specific type of error will occur with a lower probability. (An easy case is "preferring" double interception to no interception errors by controlling the threshold distance under which two tracks are considered as one object.) Architecture planners and defense system designers should be aware of this issue and develop the necessary tools to deal with it effectively.

F. Real-Time Interception Coordination—Summary

Real-time coordination of interceptions offers significant benefits to the missile defense architecture. We have shown that many of these benefits may be achieved by relatively simple procedures. The resulting architecture will be both more effective and more robust in terms of being able to counter deviations from the planned-for threat. Mutual backup possibilities are an added benefit, as is the enhanced ability to defend against threats to the defense architecture itself.

However, these benefits do not come without an associated cost. The cost of coordination is multifaced. It consists of the actual cost of linking the systems, the added complexity in the algorithms, and the inherent possibility of errors in the process. Defense architecture designers should consider carefully the cost-effectiveness of each proposed implementation, and strive to adopt the simplest possible approach for the specific architecture at hand. To effectively conduct this analysis, good knowledge of both defense systems concerned is required, as well as a thorough consideration and assessment of the expected threat. Some of the considerations and techniques discussed here, as well as in other chapters of this book, may be of value in deciding on the best way to perform this task.

VIII. Interception

The next phase in the interception process, following defense planning, is the actual carrying out of the interception. Interoperating systems may take part in this process also, and in any kill assessment activity that follows it in a shoot-look-shoot situation. Let us follow the interception process stage by stage, discuss the benefits that interoperability offers at each stage, and determine the requirements that may be imposed to take advantage of them. Following the successful creation of a defense plan, a tracking regime is defined, with the goal of reaching a sufficient accuracy at the time required for interceptor launch. This regime is to be carried

out by the radar of the system that is about to engage the TBM. However, the initial plans may be disturbed by a large number of occurrences, from jamming, to attack on the radar, or simply to excessive load caused by the need to perform other tasks (e.g., engagement of higher priority targets).

If the defense system is working on its own, these occurrences may result in a change of the defense plan (e.g., lowering interception altitude to gain more time) and, in the worst case, even in aborting a specific interception. Interoperating systems may provide a "safety net," by taking over the tracking of the target at a specific point and returning it to the system chosen to engage with improved accuracy. Such a process would require intimate knowledge of the details of the tracking regime and its goals by the interoperating system. Therefore, it would require interoperability of level IV or V. Implementing such a high level of interoperability, particularly among systems belonging to different nations' task forces, entails very complex changes in both systems. The cost-effectiveness of such solutions should be weighed, based on the attributes of the actual systems involved.

Another possible use of interoperability at this stage, as well as in the actual launch and interception stages, is to have the peer system prepare a defense plan of its own to be used as a backup in case the first system fails in any stage of the process. This can be done relatively easily, using only level II or III interoperability (not level I, because some information on the target, and possibly the current plan, has to be shared). However, this is a wasteful process, doubling (or almost doubling) the resources devoted to a single interception over the entire architecture.

The next stage in the engagement process is the actual launch of the interceptor and supplying it with information until some predefined point in its flight via uplink. (For more information on these topics, the reader is referred to Chapter 7.) Again, interoperating systems may be used to assist in this process. Several options, at various levels of complexity, exist.

1) The peer system may actually guide the first system's interceptor (level V interoperability). This is the most complicated approach, bringing to the fore problems of compatibility of wavelengths, uplink/downlink messages, etc. If the two systems will not be jointly planned to include this capability, implementation at a later stage would entail very large investments and considerable development risks.

2) The peer system may supply information to the original system, which would use it to update the interceptor (level IV interoperability). This could be done throughout the interceptor flight, requiring very good knowledge on the part of the peer system of the data type and accuracy required by the interceptor. Alternatively, it may be done up to a predefined point where the engaging system has to acquire the target on its own, simplifying the process and the data requirements involved.

3) An alternative defense plan may be placed on hold in the peer system, as a backup in case the interception fails at any point in the process (level III interoperability and up).

This last option brings us to the last stage of the interception process. The assessment of the engagement result and the possible launch of a second tier interceptor.

IX. Shoot-Look-Shoot Applications

A defense system applying shoot-look-shoot involving a kill assessment mechanism may use information supplied by a peer system to enhance the effectiveness

of the process. (For more information on kill assessment processes, see Chapter 7.) Obviously, this requires that the receiving system be aware of exactly what type of information is being supplied, from which location (since relative geometry is very important), etc. It is questionable, however, how beneficial is this approach, especially in cases were the constant presence of the peer system cannot be relied on (as is the case in most situations involving coalition warfare). Shoot-look-shoot can also be conducted as a combined defense plan, where one system provides the first layer and the other provides the second layer. Indeed, this appears to be a very natural procedure when one of the interoperating systems is an area defense system and the other is a point defense system.

The application of combined defense plans brings about two issues that have to be resolved: 1) How is kill assessment to be conducted? 2) What is the effect of the time intervals involved on the effectiveness of the process?

The first question is not an easy one to resolve. Combined kill assessment (KA) brings about even more complex problems than those already discussed for assistance by one system in KA performance by the other. If only one system is to perform the kill assessment, the question is which one will it be. On the one hand, the system that performed the interception has the most knowledge about the way its interceptor behaved during the interception (including the most up-to-date downlink data prior to interception), and is also the most aware of the type of sky-pictures resulting from its interception. It is therefore the best equipped to decide on the kill result. On the other hand, if the system that has to perform the second layer fails to identify the target, the result of the kill assessment process is meaningless. If KA is performed by both systems, whose judgment should be preferred in case the results do not match? There is no clear-cut answer to these questions, nor is there at the moment sufficient experience in kill assessment and kill pictures to support such an answer.

The second question just posed brings us back to the issue of the time for coordination, which was already discussed earlier in this chapter. In its current incarnation, the problem takes the form of assessing the effect of the time taken by the kill assessment process on the defended footprint of the lower tier system.

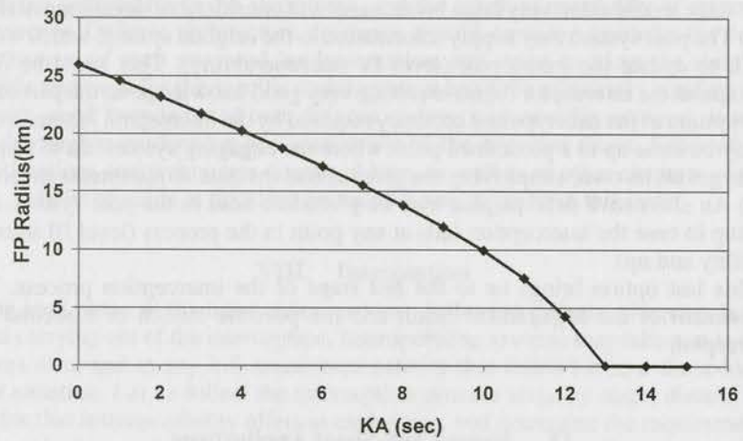

Fig. 17 Footprint radius as function of kill assessment time.

One should bear in mind that the time involved includes the KA time, the communication time, and the reaction time of the activated system.

To get a feeling for the effect of this type of time line problem on defense system performance, we return to our familiar example from Fig. 1. We will now focus on an 800-km TBM, attacking DA 1. System A, the area defense system, attempts to intercept it at 60-km altitude. If it fails, system B, the point defense system, is to perform a second layer interception at 20-km altitude. Figure 17 shows the size (radius) of the defended footprint by the lower tier system, as a function of the time taken for kill assessment and related activities. As can be seen from the graph, the problem is by no means negligible, and should be considered whenever such joint procedures are contemplated.

X. Debriefing in an Interoperability Environment

Debriefing is an integral part of any military operation, and debriefing capabilities should form a part of every defense system. In the field of TMD, the debriefing capabilities are especially important, and should be as extensive as possible, because of the relatively small amount of data currently available and the rarity of actual data (the difficulty in actively training for TMD). The situation is even worse where interoperability is concerned, for a number of reasons:

1) Training and actual operation, in any particular interoperability setting, will be extremely rare. This means that any such occurrence should be used to gather as much data as possible.

2) The data relevant for analysis of the combined operation of interoperating systems are much harder to accumulate than the data relevant for analysis of a single system operation.

3) The ability to implement the lessons learned from debriefing (especially in the short time allowed to respond within the same war), is also constrained by the fact that two nations' task forces may be involved.

Let us take a minute to elaborate on the second issue (the first will be discussed later on). Analyzing interoperability operations requires gathering information on activities (e.g., coordination processes) conducted between the systems, rather than at each system individually. As an example, let us look at the data required to analyze the results of a single interception, which was coordinated between two defense systems. We assume that an automatic TIS algorithm handled track correlation, and the interception coordination was done verbally.

First, we need to establish that the systems were indeed talking about the same target. For this, we need to know the following: 1) What was the information at each system at the time of track correlation and what data were transmitted across the communication network. 2) What was the interpretation given to the data at each system (as we can not assume that both systems perform the same association tests). 3) What data were actually displayed to the human operators performing interception coordination at the time the coordination occurred (which may be different from the final results of discrimination algorithms stored in the system's files). 4) Was the TBM correlated really the one talked about by the operators (they may have talked about one TBM but conducted operations on a different target).

Following this analysis, we should turn to analyzing the coordination process and its results. For this, we need to know the following: 1) What were the prevailing rules of engagement at the time of coordination (because ROE may change

dynamically during the war). Specifically, what were the rules set by higher command concerning the priority assigned to each defense system vs the specific threat? 2) What were the defense plan options available to the operators in both systems. We need to know what options were available inside each system, which of them were actually discussed in the coordination process, what was the basis for the decision-making process (which parameters were discussed), and which was the selected defense plan? 3) What was the timing involved in the process, when was the process started, how long did it take, and what was the time allowed for coordination?

The next step in the process is the debriefing of the actual interception. To properly analyze this phase, we need to know what each operator in each defense system did (e.g., activate DP, place DP on hold, change specific parameters), at what time line, and what was reported to the peer system. Finally, a similar process should occur for the kill assessment and launch of the lower tier interceptor, if this were a part of the defense plan executed.

Looking at this one example, it is easy to see some major problems that are involved in debriefing operations involving interoperability: First, for the debriefing to be effective, the data files of both systems should be analyzed together. This will inevitably cause a problem in the general case, where the systems do not come from the same nation's task force. Second, coordination is, by nature, an act of decision making. Even if we can reconstruct the database and time line that was available at the time the decision was made, several interpretations of the data are probably legitimate. A "schoolbook solution" doesn't exist in most situations. Third, coordination is a decision-making process that takes place at more than one point in space (i.e., it occurs *between* the two systems using, ultimately, human conversation across a verbal communication net). This means that complex analysis of verbal communication protocols is required.

The question of how to conduct efficient debriefing in an interoperability setup has to be resolved on a case-by-case basis. It is important that this issue not be overlooked whenever interoperability between defense systems, particularly systems belonging to different nations, is being discussed.

Before we leave the subject of interoperability, we feel that a short discussion of human-in-the-loop issues is in order. These issues are not particular to the case of TMD systems, but some aspects are singular, due to the special nature of the TMD war.

XI. Human-in-the-Loop Considerations

As has been continually stressed throughout this chapter, the major differences between interoperability in TMD and interoperability in any other military operation, are the short time intervals under which decisions have to be made and carried out and the uncertainty as to the actual battle picture, stemming both from its complex nature and from the small amount of data available and real combat experience. These two characteristics lead to two important requirements for TMD systems wishing to operate together. These are 1) the need for constant training/exercising and 2) the need for operator assistance tools.

We will discuss each of these requirements briefly, not attempting to provide general solutions, but rather to provide some useful observations and guidelines.

A. Training for Interoperability

The complex nature of the processes involved in interoperability, together with the lack of real-time operational experience and the timing problems involved, create a need for constant training, both at the single system as well as the multisystem (or nation) level. However, these same factors also impose difficulties on joint training and exercises. The problem is somewhat alleviated when several nations use equipment manufactured by a common source, but even then problems of compatibility between various versions of the same system, not to mention problems arising from different doctrines and modes of operation, and even national culture, will doubtless arise.

Thus, there is a need for constant training and exercising to reach proficiency in operating a joint architecture, but the opportunity to conduct such joint exercises is rare. Exercises involving live firing are even rarer, yet some aspects of interoperability can only be tested and exercised in them.

A partial solution to the problem lies in the use of simulations and simulators. Simulations can be used with relative ease. Using today's distributive interactive simulation (DIS) protocols, exercises can be performed in remote locations. Yet, even so the performance of such exercises is a costly and complex undertaking. Moreover, these exercises are good for testing issues such as combined standing operating procedures (CSOP), language problem, time line, etc., but they are not adequate for testing, for example, the physical link between the systems. (More information on the use of simulations in interoperability exercises may be found in this book, in Chapter 14.)

There is a need, therefore, for establishing a training regime for any two systems wishing to interoperate. This regime should include periodic exercise in different forms, and it should touch on all of the aspects of interoperability. Specifically, one may define the following aspects as describing the elements that should be trained and exercised.

1) Engagement coordination: The ability to perform efficient coordination of engagement, either automatically or through voice channels.

2) CONOPS/CSOP: The existence of commonly accepted rules and organizational structures able to support the required level of joint operation.

3) ASP correlation: The ability of the joint architecture to create (either automatically or by voice) a common "sky-picture."

4) Equipment connectivity: The ability of the two systems to physically connect one to the other and send and receive all the data required for performance at the agreed upon level. This includes both automated and vocal procedures.

5) Time line/response time: The available time for the aforementioned activities, in a real engagement.

6) Mutual debrief: The ability to jointly reach conclusions, within a reasonable time, as to the effectiveness of the joint operation, and ways to rectify the problems identified.

7) Other joint procedures: The ability to jointly conduct activities such as discrimination, kill assessment, etc.

8) Logistics: Readiness of the joint architecture in terms of being able to support actual deployment of forces and supplying their material needs, and the required infrastructure for connecting the systems.

Several types of exercises are available to train the combined architecture. They are 1) simulations (including human-in-the-loop models), 2) war games, 3) joint

Table 7 Interoperability exercises

Attribute	Exercise type					
	Simulation	War game	Plan meeting	CPX	Deployment	Live firing
Logistics		+	+	+ (+)	++	+
Equipment connectivity			+	+	+ (+)	++
CONOPS/CSOP	+ (+)	+	+	++	++	++
ASP correlation	++	+			++	+ (+)
Engagement coordination	++	+			++	+ (+)
Other procedures	+ (+)	+			+ (+)	++
Mutual debrief	+ (+)	+	+	+ (+)	++	++
Time line	++				+ (+)	++

planning meetings, 4) command post exercise (CPX), 5) unit deployment, and 6) live firing.

Obviously these methods may be combined. For example large-scale exercises may involve several units actually deployed in the field whereas other units are participating via simulations. Such an exercise may even culminate with one or two units conducting live firing of interceptors. To fully train and test the combined architecture, a carefully defined mix of the previously defined exercise types is required. To see why this is so, one needs to look at a table of each of the elements of interoperability, such as Table 7, which shows the level of testing achieved in each type of exercise. In the table, one plus sign (+) means that the attribute in question is partly tested by the specific exercise type, a plus followed by a plus in parentheses (+(+)) means that an attribute is almost fully tested, and two plus signs (++) mean that it is fully tested.

A few notes regarding Table 7 are in order. First, as can be clearly seen, only live testing, including firing, can accomplish true testing of all the attributes of interoperability. However, it is possible to test a lot of the attributes to a large degree by simpler means. Second, we assume that even in large-scale exercises live firings will be conducted against single targets and not salvos (this seems to be a natural assumption, at least in the general case). This is why, for example, live firings do not fully test the engagement coordination capability. And third, the contents of the table reflects the authors' experience and beliefs and is, of course, debatable. However, no matter what other views are held by the reader, the main message of the table will remain this: No single type of exercise fully tests the ability of two systems/units to conduct interoperability at any specific level.

We strongly believe that the level of training and testing by the involved units/systems will play a major role in the success of TMD interoperability operations. A training regime should be set, based on the specifics of each case, to include a combination of the different exercise types. Simulations and war games probably will play an important role in any such training regime, but we must not delude ourselves into thinking that they can, by themselves, fully test the combined architecture.

B. Operator Support Tools

Interoperability between defense systems probably will be conducted by a combination of computerized algorithms and communication between human operators. The grave nature of the decisions being made (affecting directly the probability of assets being hit by TBMs) lends itself naturally to human decision making, or at least human monitoring and approval of computerized decisions. On the other hand, the short time available for decision making puts extreme pressure on the human operator and may even, at times, make human intervention impossible. The immediate conclusion from these observations is that designers of TMD systems who wish to operate jointly, should invest a lot of effort in supplying their human operators with adequate support tools to carry out this complex mission.

The goal of support tools should be twofold: To allow the operator to "see" the picture viewed by his peers in the other system, as accurately as possible (situation awareness tools), and to alleviate the timing problems by providing as much advanced information as possible. (Decision support tools).

We conclude this brief discussion with some comments regarding specific operator assistance tools and their utility.

1) TIS algorithms are almost a must when it comes to coordination between weapon systems. The time required for verbal correlation of the sky-picture is prohibitive from the point of view of effective battle management. Care should be exercised in careful design of the algorithms so that changes in the sky-picture representation at the system level are not automatically repeated at the operator level. (The system may take a number of "trial and error" steps to correctly discriminate, for example, a warhead from its associated decoys. Displaying all of the interim stages of this process to the user may overload him or her.)

2) Early knowledge of any event in the sky is important. Displaying external cue information to the operator allows for early correlation and coordination procedures, and lets the operator prepare for the tasks to follow. On the other hand, too much advanced information, or too early a presentation of the data, will clutter the display and make it more difficult for the operator to understand the immediate sky-picture. The best mix of data to be displayed is particular to each system, and is affected by the display capabilities and the role the operator is expected to play in the interception process.

3) For effective coordination of interceptions, or monitoring of a computerized coordination process, the operator needs to be made aware of most, if not all, of the parameters being exchanged and weighed. It is very important for the operator to be able to understand what is to be contemplated, and reach a decision based on known and clear rules.

4) Other, more advanced, methods for saving coordination time may be suggested, based on the specifics of the systems being connected. An example is that an upper tier system may attempt to predict the capability of a lower tier system to intercept a given track, by an algorithm representing the qualities of the lower tier system. Alternatively, a lower tier system might be fitted with the capability to conduct its defense planning process based on external cueing data, without self-detection of the target. Both methods allow longer time for coordination, and each entails its own benefits and drawbacks. (For example, planning according to cue data allows for a full coordination process by the system that is actually going to carry it out. On the other hand, it takes resources off other tasks, and may wind up allocating resources to a target that will not be detected or intercepted.)

The human operator involvement in interoperability processes, both offline and online, is an issue that requires more analysis and thought. It is the view of the authors that this will turn out to be one of the core issues for interoperability studies in the near future.

XII. Summary

Although interoperability has been, in recent years, the subject of many studies and analyses, as well as actual progress in working out the details for connecting specific systems, it is the authors' view that we are still just scratching the surface of this issue. Many of the important questions and parameters discussed in this chapter were completely unknown a few short years ago. Many more, we feel, are still lying in uncharted regions of this vast field.

We have, however, learned a lot in recent years. Treatment of interoperability has advanced from neglect, to fervent study and analysis, and to actual implementation of some algorithms, links and protocols, designed to accomplish at least a part of the tasks discussed in this chapter. As implementation continues to mature, and is tested and trained in various drills and exercises, the knowledge base will continue to grow, and our understanding of the fundamental issues governing interoperability will be heightened.

Just as in many other areas in TMD, we are still trudging the road toward a full understanding of interoperability. We hope the present chapter helps, if not actually to navigate us onward, at least to chart the course we have already covered.

Acknowledgments

The authors would like to acknowledge the work of Karel Pick and Yuval Karakookly in preparing material for this chapter.

References

[1]*Dictionary of Military Terms*, U.S. Dept. of Defense, Greenhill Books, London, 1995.

[2]Spiegel, M.R., *Theory and Practice of Statistics*, Schaum's Outline Series in Mathematics, McGraw–Hill, New York, 1961, App. IV.

[3]Blackman, S.S., and Banh, N.D., Hughes Co., "Track Association Using Correction for Bias and Missing Data," *Signal and Data Processing of Small Targets*, Paper 2235-34, Vol. 2235, edited by Oliver E. Drummond, Society of Photo-Optical Instrumentation Engineers, 1994.

Chapter 9

External Cueing

Dror Cohen,* Eitan Yariv,† and Ayala Gur ‡
WALES Ltd., Ramat-Gan, Israel

I. Introduction

THE process of external cueing, i.e., supplying/receiving target-related data from sources external to the defense architecture, is a special form of interoperability. Interoperability between defense systems is discussed in some detail in Chapter 8. However, we feel that external cueing, because of its relative ease of implementation (compared with other forms of interoperability) and its many uses, merits a chapter in its own right. The basic process of external cueing is depicted graphically in Fig. 1.

The external cue source detects the target before it is detected by the defense system radar. It transmits the target data to the defense system, enabling earlier detection and allowing more time for the defense system to react to the threat. This can either increase effective radar coverage by providing detection at greater ranges and angles than the radar can effectively scan or increase effective radar capacity by freeing radar resources (both in search and track). The main commodity exchanged in this process is time. On the one hand, time is "gained" in terms of earlier detection. On the other hand time is "lost" in terms of the requirement for the defense system to invest extra resources (long-range cued beams) and to track the incoming tactical ballistic missile (TBM) for a longer time. The balance between the time lost and the time gained provides the benefits of external cueing. These benefits are a function of the threat, properties of the cue source, the intended use of the cue data, and the defense system capabilities. The purpose of this chapter is to map the relationships between these elements and their impact on the effectiveness and utility of the cue.

The cue source is any sensor that is not under the control of the defense system. It may be a sensor dedicated for this purpose, such as a surveillance satellite, or a sensor performing external cueing as a secondary activity, such as a forward-deployed radar of another weapon system. It may even be an "opportunity-based

*President. Member AIAA.
†Team Leader, Interoperability, External Cueing, and Concept of Operations. Member AIAA.
‡Senior Systems Analyst. Member AIAA.

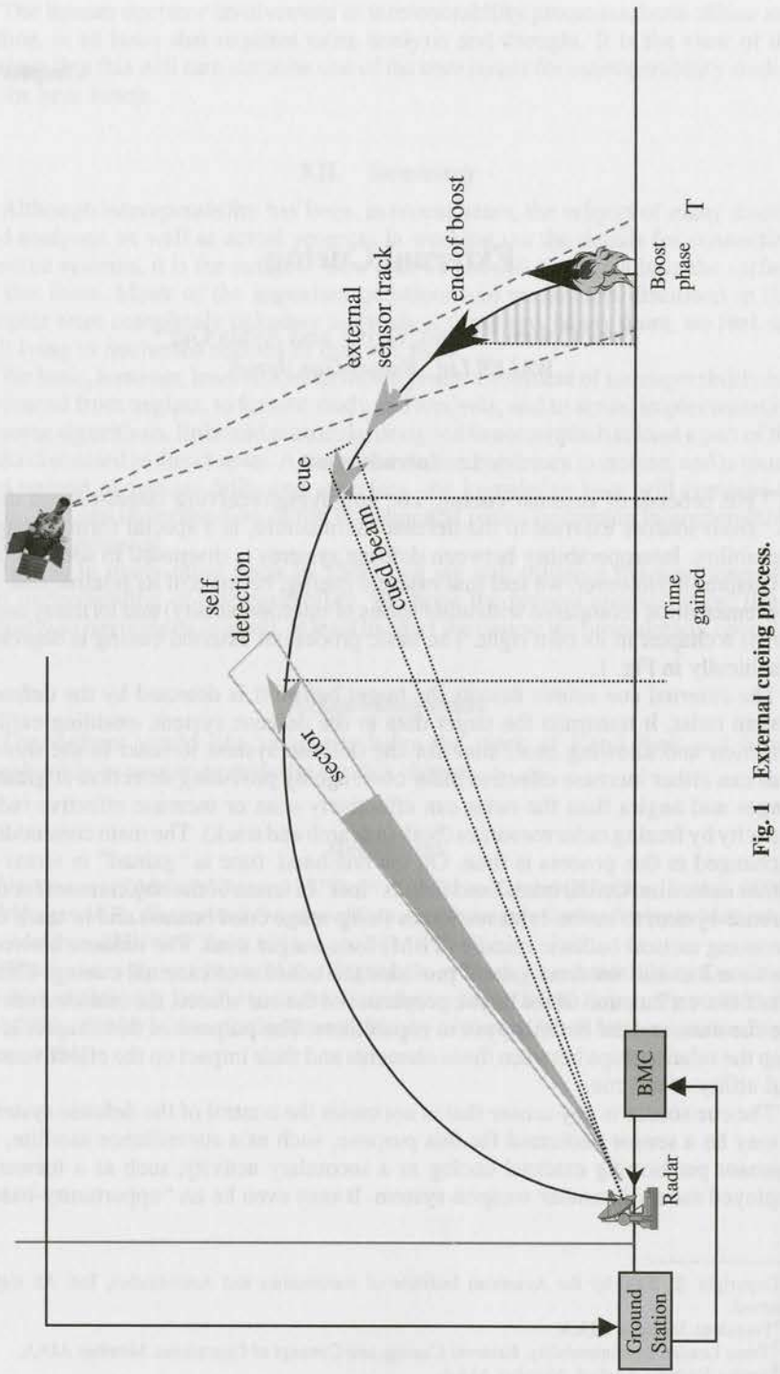

Fig. 1 External cueing process.

cue" supplied by aircraft that happen to observe a TBM launch. Cue data, likewise, may vary from general early warning and up to a very accurate state vector. Cueing may be provided once or continuously as long as the remote sensor is tracking the TBM. Delay in providing the cue may be anything from a few seconds up to minutes. The effective use of the cue data depends on correct implementation based on the cue charateristics.

There are a number of possible ways to map the connections between the elements mentioned above. In this chapter, we chose to start with defining the possible uses of cue, from the point of view of the defense system. We will then analyze each possible use and identify the merits that may be gained, the pitfalls that should be avoided, and the requirements posed by each specific use on the participating systems. For the sake of convenience, we shall limit our discussion to the relatively simple case of a single radar/single cue source throughout most of this chapter. The case of multiradar/multicue sources will be discussed only briefly, in a separate section. Finally, we shall construct a table summarizing the connections between cue source qualities, cueing system qualities, and cue use.

II. Possible Uses of External Cueing

The possible uses of external cue data are listed, arranged by order of complexity.

1) *Early warning:* Using the early data provided by the cue source to warn the population at the predicted impact point of the TBM. In this use, there is no benefit to the active defense system, but the importance of a few minutes of early warning to the passive defense layer need hardly be stressed.

2) *Early calculation of TBM launch point:* A companion use to early warning. External cue data are used to activate and guide forward deployed forces, as counter force. Since this book deals only with defensive aspects of the ballistic missile war, this will not be dealt with in great detail.

3) *Early detection:* The "classic" use of cue data. There are a number of variants here, but we shall deal mainly with two of them: First, cue as a companion to search, where the defense system radar continues its normal search functions while devoting some resources to cued acquisition at longer ranges or extreme angles. Second, cue instead of search, where the defense system radar does not perform autonomous search, but relies entirely on cue data for target acquisition and performs solely in fire control mode. Obviously, there are numerous options between these two extremes, but they can be treated as a continuum involving different "doses" of the two extreme solutions.

4) *Defense planning based on cue data:* Delaying own radar acquisition of the TBM to the last possible moment before launching an interceptor, while relying on the data supplied by the cue source for all defense planning functions. There is a range of options between assistance in acquisition through full defense planning.

5) *Interceptor launch using cue data:* In this application, cue data is used to launch, and possibly even guide, the interceptor toward its intended target.

The aforementioned definitions of possible uses, though somewhat crude, provide a good background for continuing the analysis. It is, of course, possible to refine the division presented, but we feel that for our present goals, this level of refinement will suffice. The remainder of this chapter will consist of analysis of each of the possible uses of cue as defined. The main emphasis will be placed on

uses 3 and 4, because we believe they are the ones most cue users are likely to pick as providing the highest gains, in terms of cost-benefit, to their systems.

III. Early Warning

The first use of cue data to be discussed is early warning. The goal is to provide an accurate warning earlier than is possible by the defense system's own radar. This is a relatively simple case, involving two main parameters: time and accuracy. The time gained is proportional to the range of the TBM from its impact point when the early warning is provided. Figure 2 shows three representative TBM trajectories. The X axis presents the range to the TBM impact point, and the Y axis presents the time to impact. From this graph it is easy to deduce the amount of time gained from early warning per TBM type, relative to own radar capabilities. It can be seen, for example, that if a TBM is detected at a range of 1000-km, the early warning time is between 6 and 7 min. If the same TBM is detected at a range of 800 km the early warning time is approximately 5 min. Again, if detection occurs at the 300-km range, the early warning time is reduced to less than 3 min. One may draw the following conclusions. An external sensor's contribution to early warning is significant for those TBMs of longer range than the defense system (or other detection assets) radar detection range. Additionally, the gain in time for each kilometer of detection is greater for shorter range TBMs (this is seen from the gradient of the graphs).

A point regarding accuracy should be made here. The accuracy of impact point prediction by the defense system (or by the cue source itself) depends on the accuracy of the cue supplied. Although it is not necessary to pin-point the impact point for early warning, the uncertainty ellipse should be small enough to support warning the population in specific areas. For this reason, although the gain in time for earlier detection is greatest during the TBM's boost phase, data before the TBM end of boost is of limited use.

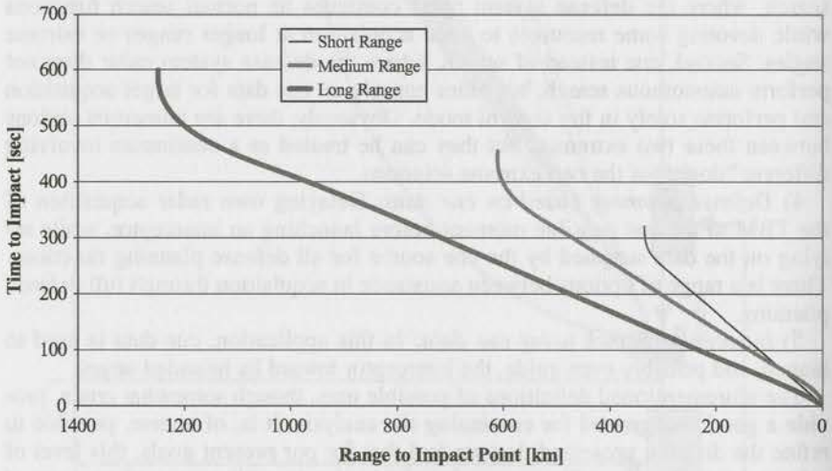

Fig. 2 TBM range vs flight time.

IV. Calculation of TBM Launch Point

Another possible use of external cue data is the early calculation of the TBM launch point. This can be done either before the TBM is acquired by the defense system itself (relying on cue data only) or after self acquisition (combining the data from external cueing and the system's own sensors). Because, obviously, it is important to estimate the TBM launch point as early as possible, the first option should be given high priority. The applicability of launch point calculation based on cue data alone depends on three parameters.

1) Arrival time of the cue. The cue should arrive before the system acquires the TBM by itself. (A more elaborate discussion of this issue will be given in the next section.)

2) Cue data. The cue data should include either the state vector of the TBM as close to launch as possible or the launch point itself. In this case waiting for end of boost is unnecessary.

3) Cue accuracy. The level of accuracy should be sufficient for the defined task.

This last point requires some elaboration. Obviously, the accuracy of the resulting launch point prediction (i.e., the size of the uncertainty ellipse) depends on the algorithms used. However, in all estimation algorithms the results will be directly related to the errors in the cue data. Figure 3 presents the length of the major axis in the uncertainty ellipse as a function of the (average) error in the cue data. In preparing the graph, we used a simple model for track calculation (assuming a Keplerian track). Looking at such a graph, based on the algorithms used in a particular system, and knowing the accuracy required for the launch point location, the reader may deduce the required accuracy of the cue source, to enable it to be used for launch point (LP) estimation.

Combining the data from two or more sensors can, of course, provide a higher accuracy than is obtained by each sensor on its own. However, achieving this higher accuracy requires implementing data fusion algorithms, and the results may not always justify the effort. To see this, we may look at Fig. 4. This graph shows the accuracy of the LP calculated by a combination of data from two sensors, as a function of the accuracy of each sensor data by itself. We have assumed a common time base, i.e., that all track reports are correctly synchronized in absolute terms. The fusion algorithm used was a simple best estimate trajectory (BET) algorithm. The BET algorithm calculates a weighted average of corresponding

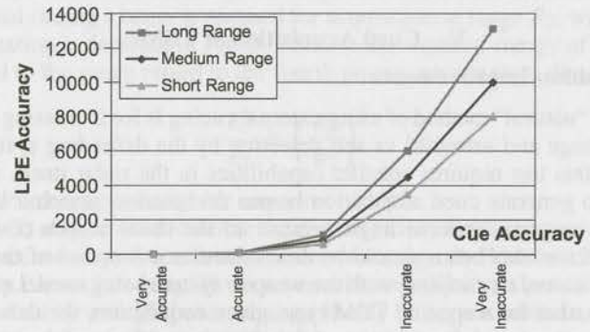

Fig. 3 Uncertainty in launch point estimation as a function of cue accuracy.

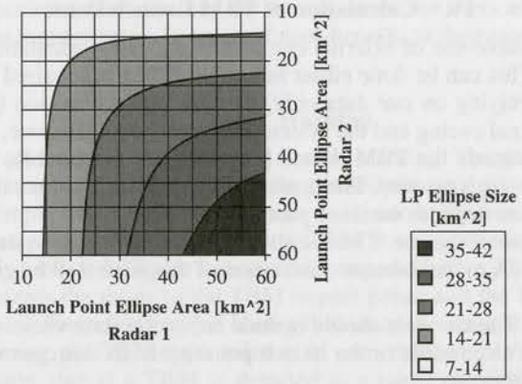

Fig. 4 Combined launch point estimation accuracy using best estimate trajectory.

track components, i.e.,

$$\bar{X} = \left[\frac{1}{\sigma_1^2}\bar{X}_1 + \frac{1}{\sigma_2^2}\bar{X}_2\right] \Big/ \left[\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}\right] \tag{1}$$

where $\bar{X}_1$, $\bar{X}_2$ are the measurements of each sensor and $\sigma 1$, $\sigma 2$ are the corresponding covariances. The maximum contribution is expected when the two sensors have similar accuracies.

Combining this graph with the graph in Fig. 3, one may deduce the contribution of external sensors relative to own system characteristics, the cue sources available, and the data fusion method considered. However, it is important to note here that data fusion should also be considered for other uses of cue data.

A word of warning should be inserted here. The results in Fig. 4 are obtained knowing the accuracy of the data from each of the sensors. In reality, the accuracy of each track may not be known. The actual measurement errors come into play here, and so does the sensor's bias, sometimes resulting in an inability to accurately define the accuracy of the combined track. If the LP itself is provided, the method of calculation must be known in order to know the size of the uncertainty ellipse.

V. Cued Acquisition of TBMs

A. Acquisition Improvements

The most "natural" method of using external cueing is for increasing acquisition coverage (range and azimuth) vs self detection by the defending system's radar. Obviously, this use requires specific capabilities in the radar itself, namely the capability to generate cued acquisition beams designed to search a limited area at a longer range or extreme angle relative to the usual search coverage. The characteristics of this beam should be determined as a function of the radar and threat qualities and its coupling with the weapon system being used. Let us assume for example, that for a specific TBM type, given acquisition, the defense system takes t_1 s to prepare a defense plan and that the time of flight of the interceptor in its longest possible trajectory is t_2 s. Then it is clear that $t_{\max} = t_1 + t_2$ is the longest

Table 1 Acquisition range gain using external cue

TBM range	Characteristics (approximate)	Cued acquisition range	Acquisition range gain
Short	$R_{\text{TBM}} < R_{\text{MSD}}$	$R_{[\text{EOB}+T_D]}$	$R_{[\text{EOB}+T_D]} - R_{\text{SDS}} <= 0$
Medium	$R_{\text{MSD}} < R_{\text{TBM}} < R_{\max}$	$R_{[\text{EOB}+T_D]}$	$R_{[\text{EOB}+T_D]} - R_{\text{SDS}} > 0$
Long	$R_{\text{TBM}} > R_{\max}$	$R_{\max}$	$R_{\max} - R_{\text{SDL}} > 0$

time the system needs for intercepting any TBM. Detecting a TBM at $t > t_{\max}$ before impact (TTG = time to go) is therefore useless from the point of view of the defense system. Because the basic equations determining TBM trajectory are well known, $t_{\max}$ may be used in conjunction with the TBM data to determine the desired cued acquisition beam range.

The actual cued acquisition beam range will generally be smaller than the maximum already defined. The actual range is a result of an optimization process based on available radar resources, the dependency of t_1 on acquisition range, the resulting energy requirements, and radar technical characteristics. For example, let us take a defense system with a given radar and cue supplied based on TBM end-of-boost data. Define R_{MSD} as the maximum self detection range of the radar search regime, and $R_{\max}$ as the maximum radar acquisition range (the range of the cued beam). In addition, R_{TBM} = range of TBM, T_D = time of detection process (radar and cue source) as well as data transfer delay, $R_{[t]}$ = range between the TBM and the radar at time t, and R_{SDX} = range of self acquisition (per TBM of type X). One may now define the maximum potential gain in acquisition range by utilizing external cueing based on end of boost (EOB) detection, as shown in Table 1.

B. Acquisition Resources

The gain in acquisition time does not come without a cost. Let us assume that a specific TBM is acquired in normal operation by a single beam with a specified probability of detection (Pd), at a given range R_1 from the radar. Assume now that using external cueing a beam is planned for acquisition at range R_2, with the same Pd. The situation is depicted in Fig. 5. Because the required energy of the beam is proportional to the range raised to the fourth power, we can state that

$$\frac{E_1}{E_2} \cong \left(\frac{R_1}{R_2}\right)^4 \tag{2}$$

where Ei is the energy needed for acquisition at Ri.

Equation (2) may be used to define the cost, in terms of radar resources, required to generate a cued acquisition beam for a certain range R. However, here too, the cue accuracy comes into play. If we assume (for simplicity) a pencil beam, then the area covered by a single beam at range R may be approximated as a circle (in two-dimensional space, where the third dimension is taken in the direction of the radar) whose radius is directly related to the radar bandwidth (BW) and the range

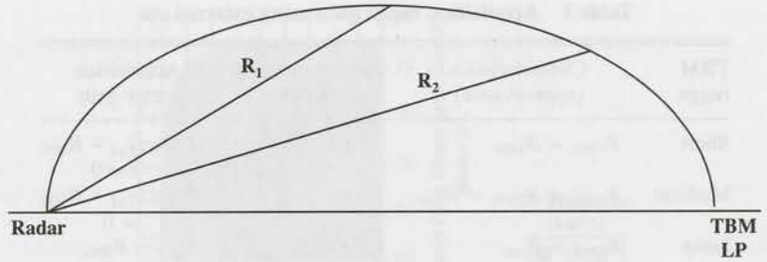

Fig. 5 Cued acquisition.

itself:

$$r = \frac{\text{BW(rad)}}{2} \cdot R \tag{3}$$

The number of beams required to cover a certain area may be calculated as the number of circles required to cover that area, as seen in Fig. 6. If the cue accuracy is (σ_x, σ_z) along the x and z axis (the y axis is taken in the direction of the radar and may therefore be neglected), to guarantee that the TBM is detected, we need to search an area of $6\sigma_x \times 6\sigma_z$ in size. The number of beams required is given by

$$N = \frac{6\sigma_x \cdot 6\sigma_z}{(2 \cdot r \cdot N\_0)^2} \tag{4}$$

where N_0 is the nonoverlap ratio between two circles, given by

$$N\_0 = \frac{k}{2r} \tag{5}$$

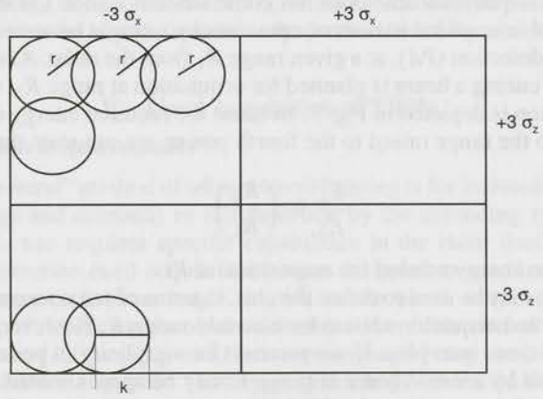

Fig. 6 Number of beams required for search.

C. Cue Contribution: Footprint

The time gained by early acquisition of the TBM may be used by the system in a number of ways. The most obvious use is to enable defense against TBMs that otherwise could not be intercepted. In other words, to increase the defense system's footprint.

Generally speaking, defense system footprint is directly related to detection range (and thus to detection time). Earlier detection of a TBM results in earlier preparation of defense plan, and therefore in the ability to launch the interceptor earlier, thus allowing the interceptor a longer time of flight (greater range) and increasing the footprint.

Quantifying the effect of earlier acquisition on the defended footprint is, however, a complex endeavor. The footprint is affected by radar coverage (both azimuth and range), interceptor trajectory (time of flight), and homing method. It requires an intimate knowledge of the weapon system and its relevant performance parameters. As a simple example, let us assume a defense system in which the interceptor flies a straight line, at constant speed, with no geometry limitations. Figure 7 shows the gain in forward footprint range as a function of the acquisition range.

In general, the following rules of thumb can be stated regarding footprint increase as a result of early detection.

1) The defense system has a maximum footprint size dependent on interceptor performance parameters.

2) If the capabilities of the radar and interceptor are well matched, the system reaches its maximal coverage without cueing

3) The contribution is expected to be larger for longer range TBMs, as long as they are within the interception abilities of the system.

4) Other benefits in interception volume may be expected, higher interception altitude, for example. These will be more significant the closer you come to the

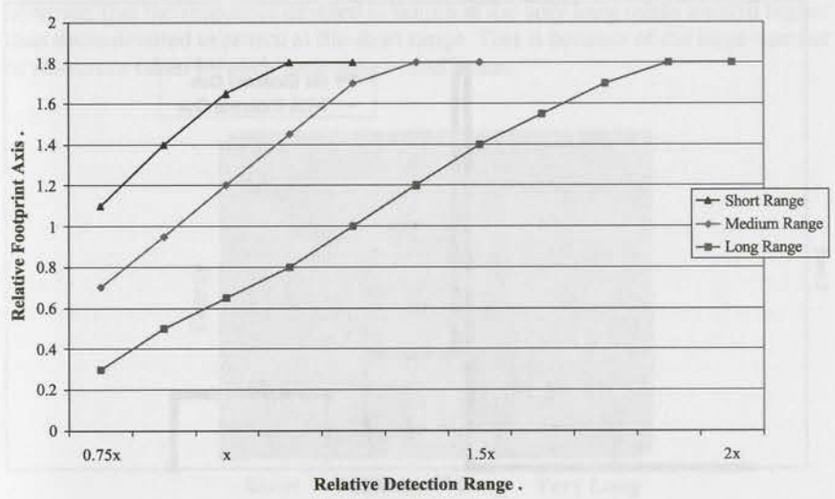

Fig. 7 Footprint gain as a function of acquisition time.

footprint boundaries, where there is more room for improvement relative to the normal operation of the system.

5) Contribution to the defense capabilities at the center of the system's battlespace is expected to be marginal.

D. Cue Contribution: Resources

The direct increase in footprint is not the only possible contribution of cueing in this mode. Early acquisition gives the defense system more time to react. Using this added time the system can now plan its use of resources more effectively, both for the specific TBM and for an entire TBM salvo (assuming cueing is available for all TBMs). To see the possible effect of using cueing to plan the tracking regime, let us look at a specific example.

Our example involves a single long-range TBM. If the system detects it by itself, there is a peak of resource allocation at the detection time. This peak is caused by the need to properly initiate a track for the TBM. This peak is avoided when external cueing is available, because the initiation of a track may be done based on the cue track. The average energy may be increased as a result of this use of long-range tracks or decreased resulting from the greater time available (lower track rate). If the cue is relatively accurate, it may be possible to delay the actual acquisition until some future point in time (delayed acquisition will be discussed later in more depth), thus both avoiding the resources peak at acquisition and investing fewer resources in the entire defense process. This basic situation is described qualitatively in Fig. 8. In this example, to enable interception using self-acquisition, a high track rate is required; however, if acquisition occurs at the long range provided by the cue, a low tracking rate is sufficient.

Other options of using the radar in the presence of cueing exist. If we assume that the cue source is constantly available and is sufficiently reliable (in terms

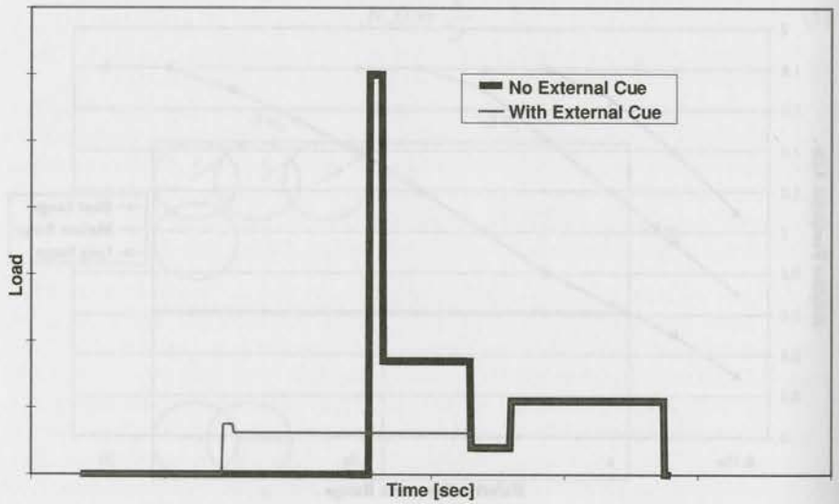

Fig. 8 Tracking load as a function of time.

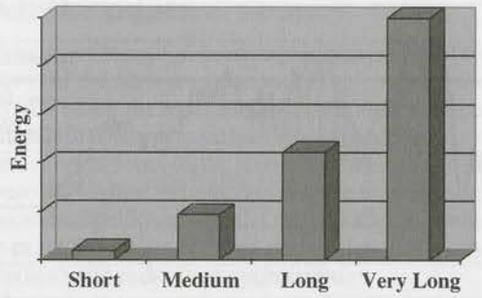

Fig. 9 Relative required energy (no cue).

of miss/false alarm rate), the defense planner may consider using cueing as a substitute, or at least a partial substitute, to search. Such practice would save radar resources, which can then be used to track and intercept more TBMs (in case the same radar is performing both search and fire control functions) or to increase the track accuracy per TBM.

Let us look at some simple examples. In the first example, a radar devotes a certain amount of energy to search. (For the current discussion, it does not matter if the radar devotes all of its energy to search or only a partial amount.) The radar searches constantly for TBMs at short, medium, long, and very long ranges. Regardless of the search mechanism used, the relative amount of energy devoted to search a given sector at each range will behave in the manner depicted in Fig. 9, i.e., in a form of an exponential function.

If a decision is made to stop searching at the very long range, and rely on cue data and cued acquisition, the resources devoted to search and acquisition are shown in Fig. 10. The resources devoted to search at very long ranges are significantly decreased, and the exponential function correlation is broken. Note, however, that the resources devoted to search at the very long range are still higher than those devoted to search at the short range. This is because of the large number of resources taken by each long-range cued beam.

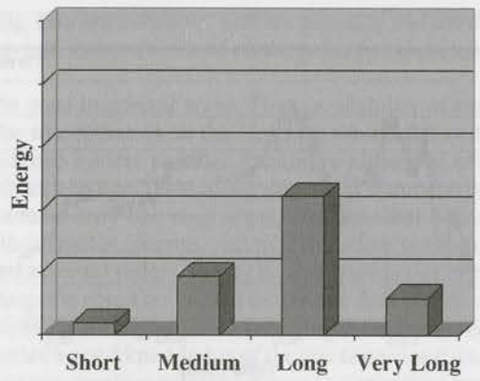

Fig. 10 Relative required energy (with cue).

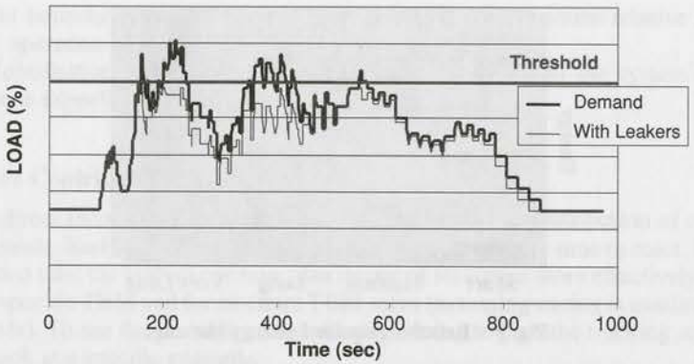

Fig. 11 Radar resources (no cue).

In a second example, a single radar has to deal with a heavy TBM attack in a short time period. In Fig. 11, the required radar load is depicted (the upper graph). Because this load exceeds the available resources, the actual results of the run indicate that a number of TBMs were not engaged and leaked because of lack of resources. The actual radar resource consumption is seen on the lower graph. Using the logic as described, of relying on external cueing for detection of very long range TBMs, the same scenario could be handled easily, without any leakers caused by resource problems (Fig. 12). Thus, wise use of external cue data may increase defense system capacity and robustness.

Having looked at this example, it is easy to construct a large number of possible search schemes, to be used in the presence of cueing, starting from not searching at all and relying on cue data for all acquisitions. Looking at the graphs presented in Figs. 9–12, it is obvious that the largest benefits are to be expected from avoiding long-range search. Partial schemes, such as diluting search as a function of radar load, not searching at a given sector, etc., may also be considered. When cueing is available, these can be performed while still providing the appropriate defense coverage.

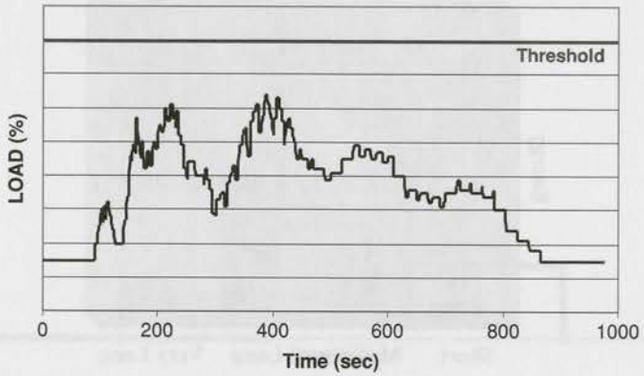

Fig. 12 Radar resources (with cue).

E. External Cueing for Acquisition: Summary

In calculating the cost vs benefit of cued acquisition per TBM type (or range), the accuracy of the cue is of the utmost importance. The cost in resources has already been discussed. While being the most natural method of using external cue data, cued acquisition entails defining special features in the defense system's radar, and also different forms of operational logic. Knowing the qualities of the cue source is important to achieve the highest benefit. We have seen, however, that with proper use the cue can provide significant benefits specifically in countering long-range TBMs (where the exact definition of "long range" depends on the qualities of the defense system studied) and radar load optimization.

These benefits become even more important when the system's performance is degraded, either because of jamming, antiradiation missile (ARM) attacks, or any other reason. Operation under degrading conditions will be discussed later in this chapter.

VI. Defense Planning Using External Cueing

A defense system establishes a defense plan against a specific target once the tracking data have reached sufficient accuracy to ensure that the target is threatening an area that the defense system is interested in defending and that the system has the capability to defend against the specific TBM (in terms of both battlespace and required resources).

Typically, the target will be tracked at a high rate up to this point, because knowing whether or not it should/could be intercepted is of utmost importance to the defender. After a defense plan has been initiated, a different tracking regime will be established for that particular TBM, which will bring the track to a required accuracy at the time of interceptor launch.

During the initial phase of the defense planning, the system has to devote a fairly large number of radar resources to reach the desired accuracy (e.g., time limitations, long range). It also has to reach decisions about the type of target being tracked, and in case the target is not alone in the sky-picture (i.e., it is accompanied by decoys, debris from other interceptions or from an earlier interception attempt by a higher tier system, a separated first stage, etc.), identification of the real target is required.

If external cueing data are available, and are accurate and detailed enough, they may be used to alleviate some of the load on the radar and also assist in the planning process.

Cue data may be used in several ways. First, availability of earlier data may be used to identify the true situation in the sky with which the system is faced. For example, if the defense system radar is tracking a cluster of objects, but cueing data were received only on one TBM along this track, it may be prudent to assume that the cluster contains only one real target, and the other objects being tracked are associated with it (motor, decoys, debris). If cueing is supplied continuously (for example, by an external radar), it may be that the history of cluster formation from the one real target is also known. All this added data could, of course, be very valuable in the defense planning process, but using it correctly requires special algorithms, and assumes a good knowledge of the cue source and its output. It also assumes that cueing is constantly available (at least in a specified sector), because the algorithm must make assumptions regarding the significance of certain objects not

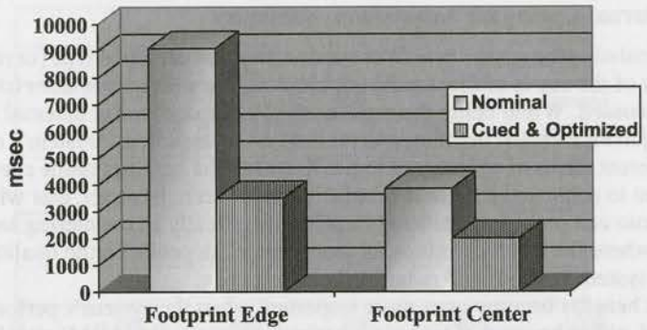

Fig. 13 Radar resources (delayed acquisition).

being reported (e.g., a cue report of one TBM, where the radar is now tracking several objects, would be interpreted as a case of breakup or decoys and not several TBMs).

If the data supplied by the cue source are accurate enough, an (initial) defense plan may be created without actually acquiring and tracking the TBM. This enables the defense system to plan a more economic tracking regime for the TBM, delaying its acquisition by the radar to the most convenient time. The most complex implementation of such a technique will be to attempt to optimize the use of radar resources over an entire salvo, based on the knowledge of future events supplied by the cue. However, such an algorithm is very complex to implement, and would almost inevitably have to make assumptions about cue availability and consistency that may adversely affect the performance in case the cue source, for some reason, does not function as required.

It is possible, however, to gain a lot of time and resources by simpler algorithms. An example can be seen in Fig. 13. In this graph the total amount of radar resources invested in tracking a single TBM aimed at different defended assets (located at the center of the system footprint or on its edge) throughout its flight are plotted. Two alternative tracking regimes are shown:

1) Detect and track by own radar without cueing.

2) Optimal detection based on cue data (minimum resources by either delaying acquisition to the latest possible time or tracking for a long time at a minimal tracking rate).

In both alternatives the same radar performs both search and fire control functions. The tracking regime in the cases plotted was designed to enable interception of the TBM at the same altitude (fixed defense capability). As can be seen clearly, it is possible to use cue data to save resources while maintaining the same defense level.

To illustrate the complexities involved in defense planning based on cue data, a simple algorithm is illustrated in Figs. 14 and 15. This algorithm implements a relatively simple use of cue data. It does not attempt to optimize the use of resources over an entire salvo but rather to delay self acquisition until a suitable time for tracking the target is found, based on defense system priorities and capabilities.

Figure 14 presents the main algorithm. The cue is handled by the defense system battle management center (BMC). It is only sent to the radar when the target (TBM)

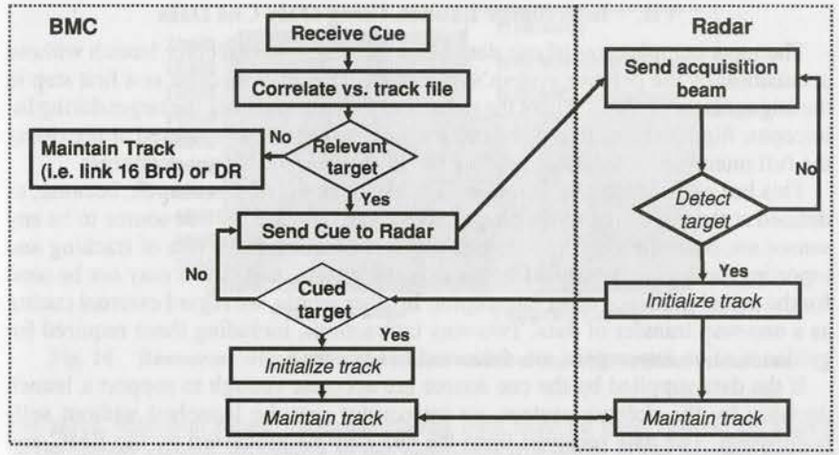

Fig. 14 External cue process flowchart.

is judged to be relevant for handling (acquisition and tracking by the radar). In Fig. 15 the process of deciding which target is relevant for acquisition is elaborated. The target is deemed "interesting" if it attacks an asset under the responsibility of the system. Targets are prioritized according to various preferences stated by the operational user (e.g., a target suspected of carrying weapons of mass destruction will have a higher priority) and according to timing considerations.

The processes in the flowcharts that are added or modified as a result of cue data are printed in boldface. It is important to note that even with this relatively simple algorithm, there are a large number of new functions that have to be developed. The exact cost-effectiveness of each cue handling policy ought to be determined by the weapon system designers based on detailed analysis of the benefits expected and the costs involved.

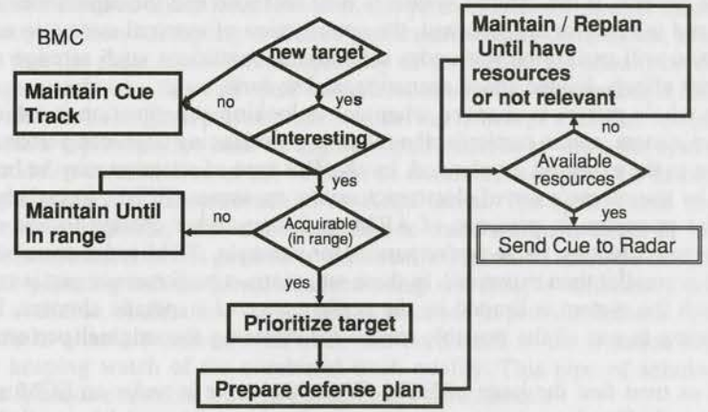

Fig. 15 Determining relevant target.

VII. Interceptor Launch Using Only Cue Data

The most complex use of cue data is for facilitating interceptor launch without acquisition by the defense system's own radar. This may be done as a first step in the engagement process, where the radar is expected to acquire the target during interceptor flight. Alternatively, the cue source may even be considered as providing the full interception solution, guiding the interceptor to its intended target.

This last realization falls, however, outside the realm of this chapter, because, as defined at the beginning of the chapter, we regard an external cue source to be any sensor not controlled by the defense system. In particular, its rate of tracking and reporting cannot be controlled by the defense system, and thus it may not be used for the actual guidance of an interceptor. In other words, we regard external cueing as a one-way transfer of data. Two-way interactions, including those required for guidance of an interceptor, are discussed in Chapter 8.

If the data supplied by the cue source are accurate enough to support a launch decision by the defense system, an interceptor may be launched without self-acquisition. The data received from the cue may be processed by the BMC and used to update the interceptor during flight. The interceptor itself, however, must be acquired and tracked by the defense system radar, if uplink capability is required.

This type of operation requires very good knowledge of the cueing system parameters, such as tracking and reporting rate, bias, etc. It also requires that the intercepting system has full confidence in the availability of cueing throughout the interception process. The defense system BMC and radar must include functions that provide the ability to operate in this mode.

The cost-effectiveness of this approach vs other uses of cueing depends heavily on the specific parameters of the defense system and cue source discussed, and will therefore be left to the consideration of each user regarding a particular system.

VIII. Radar Degraded Performance

The contribution of external cueing was analyzed so far with respect to a defense system operating in its nominal operational conditions. Contribution may be expected in a number of areas, as already described. However, it is safe to say that, in general, if the defense system is well balanced and the capabilities of the radar and interceptor are matched, the contribution of external cueing in normal operation will mainly be felt under subnominal conditions such as edge of the footprint effects, loaded attack scenarios, and so forth.

The whole picture is changed when one is looking at a situation in which the defense system, and in particular the radar, is experiencing degraded performance relative to the expected, or planned, levels. This type of situation may be brought about by the enemy's use of electronic counter measures (ECM), forced changes in radar operation in presence of ARM, or by any other change in one of the parameters affecting radar performance [for example, TBM radar cross section (RCS) is smaller than expected]. In these situations, a performance gap is created in which the system is limited by the performance of a specific element. External cueing is one of the possible means for restoring the original performance level.

Let us treat first the issue of ECM. When the radar is under an ECM attack, the general effect is an increase in the noise level, resulting in decreased *Pd* and thus in either reduced detection range or increased resources needed to maintain

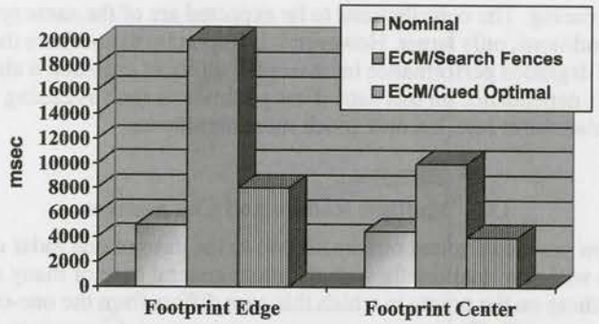

Fig. 16 Resources with and without cue under electronic counter measures.

the same detection range. External cueing data may be used instead of search at the longer ranges, according to the principles discussed in the section dealing with cued acquisition, to enable detection at longer ranges and thus restore system footprint.

Let us look at an example. Assume a defense system performance is degraded by ECM to 70% of its original value. Figure 16 shows the quantity of radar resources needed to intercept a long-range TBM at a fixed point 1) under nominal conditions; 2) under degraded performance, using the resources required to restore search performance (and thus neglecting some of the other radar tasks); and 3) under degraded performance, with optimal acquisition based on cue data.

In each case, the required tracking accuracy at any given time was the same; the radar detection range and track accuracy are a function of the case tested and the tracking rate was adjusted to achieve the required accuracy, at the correct time. Without cueing, there is a significant increase in required resources. With cueing there is an improvement.

It should be noted that to maintain the same level of performance, the case of cued acquisition under degraded performance is still costly in terms of required resources. This is because tracking under ECM requires many more resources than under nominal conditions. However, this process is much cheaper in resource consumption than attempting to restore performance using own search. In reality, given that the radar has a limited quantity of resources that should be divided among several tasks, this may spell the difference between succeeding in restoring the system to its original performance and having to accept a smaller footprint.

Another representative case of degraded performance is when the system has to shut down transmission of the radar in an intermittent or periodic fashion, for example because of the presence of an ARM threat. The system thus needs to compensate for being unable, periodically, to track a certain target or perform search functions, either in general or in a specific region. In this type of situation, defense planning and track maintenance using cue data, if sufficiently accurate cue is available, could allow the system to continue to provide defense. It is also possible to alternate between self track and cued track as the situation requires, while keeping watch of the combined track quality. This type of solution will naturally require suitable logic in the defense system BMC.

Situations of radar degraded performance cause a performance gap between the elements of the defense system, some of which may be closed by intelligent use

of external cueing. The contributions to be expected are of the same type as under nominal conditions, only larger. However, it is important to note here that while the situation of degraded performance increases the utility of cue data, it also increases the system's dependence on this data. Thus parameters such as cueing availability and miss/false alarm rate, become much more significant.

IX. Multiple Radars and Cue Sources

Up to now we have limited our discussion to the case of one radar and one cue source. We will now deal briefly with the more general case of many sources and radars, touching on the points in which this case differs from the one-on-one case.

A. Single Radar/Multiple Cue Sources

Let us look first at the case of one radar receiving multiple cues. In principle, if all cue sources are reporting on the same net, using the same reporting protocol and rules (for example, Link-16), this situation may be avoided by use of internal network procedures. However, even in this case it is possible for two cue sources to report the same track because of mistaken association (i.e., tracks supplied by different sources are missmatched). Also, it is quite possible that one or more cue sources are not operating on the same network (for example, a forward deployed collection asset of the defended country may not be linked on the same net as a satellite cueing system).

When two or more cue sources are reporting objects in the same region in the sky, two problems arise: 1) How to associate the various reports to different tracks and 2) what data to use for each track.

The first question involves the use of threat information sharing methods designed to associate between tracks provided by different systems. It is dealt with elsewhere in this book, in the chapter dealing with interoperability. The second question deals with the decision as to what data to use for the cueing process. In all cases, an algorithm to select and/or combine the data has to be provided in the receiving system.

The two possible methods of using the multiple cue source data are either to choose one of the tracks or to combine them using a data fusion algorithm. The relative merits of these two approaches may be deduced from graphs such as the one depicted in Fig. 18. Two radar systems tracking the same TBM produced this graph. One radar is deployed closer to the TBM launch point and the other is located near its intended impact point. The two tracking histories are shown qualitatively in Fig. 17.

In Fig. 18, the accuracy (the major axis of the uncertainty sphere) of each radar track is depicted, together with the two options for track combination, the "choose best" method, and data fusion [using the same BET algorithm described in Eq. (1)]. An interesting feature to note is the abrupt "jump" at the point of transition between radar 1 and radar 2 in the "choose best" graph. In real applications of the "choose best" method, it may be that some smoothing function will have to be defined to deal with these situations.

The graph in Fig. 18 represents, obviously, a specific example. Comparison of the relative utility of the two approaches will have to be conducted according to the true situation (i.e., sensor parameters and deployment, expected number of sources, etc.).

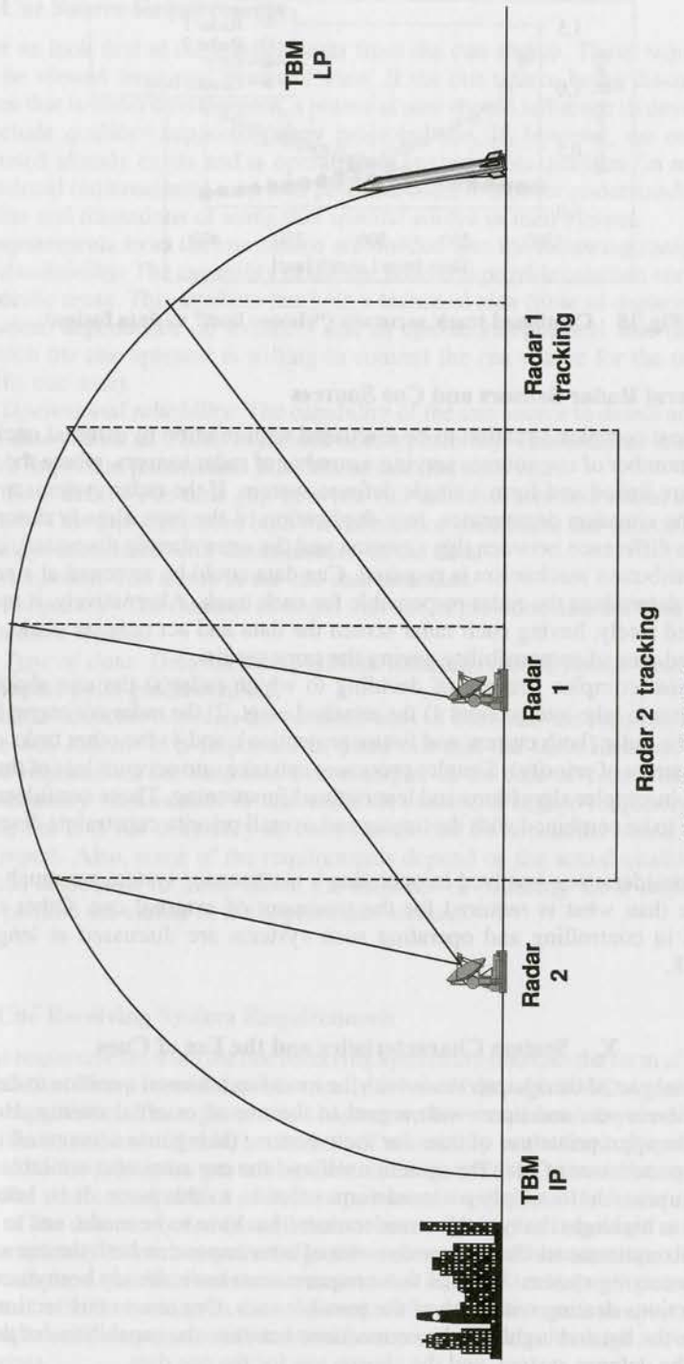

Fig. 17 Multiple cue sources.

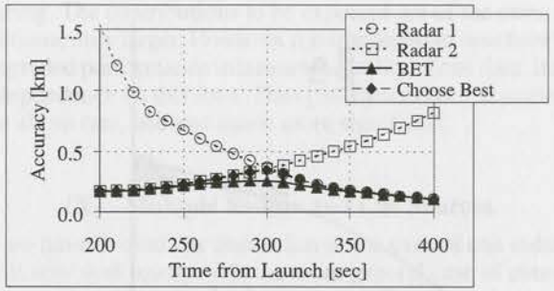

Fig. 18 Combined track accuracy ("choose best" vs data fusion).

B. Several Radar Sensors and Cue Sources

The most complex situation to be discussed with relation to external cueing is that of a number of cue sources serving a number of radar sensors, where the radar sensors are linked and form a single defense system. If the radar systems are not linked, the situation degenerates, to a duplication of the case already described. The main difference between this situation and the ones already discussed, is that a cue distribution mechanism is required. Cue data could be screened at a central unit that determines the radar responsible for each track. Alternatively, it may be distributed freely, having each radar screen the data and act only on tracks in its predefined area of responsibility, giving the same results.

The most complex process of deciding to which radar(s) the cue should be referred has to take into account 1) the attacked asset, 2) the radar coverage, 3) the load on the radar (both current and future projection), and 4) the other tasks of the radar (in terms of priority). Simpler processes can take into account less of the data resulting in simpler algorithms and less optimal functioning. These considerations will have to be combined with the timing and overall priority constraints described earlier.

The considerations involved in operating a multisensor system are much more elaborate than what is required for the treatment of external cue. Other issues involved in controlling and operating such systems are discussed at length in Chapter 8.

X. System Characteristics and the Use of Cues

The final part of this chapter deals with the most fundamental question to defense system developers and users with regard to the use of external cueing: How to choose the appropriate use of cues for their system, taking into account all of the relevant parameters of both the system itself and the cue source(s) available.

It is impossible to supply a closed-form solution to this issue. It is, however, possible to highlight the type of considerations that have to be made, and to point to certain requirements that the various uses of cues impose on both the cue source and the receiving system. Some of these requirements have already been discussed in the sections dealing with each of the possible uses. Our aim to this section is to complete the list and highlight the connections between the capabilities of the cue source, the defense system, and the chosen use for the cue data.

A. Cue Source Requirements

Let us look first at the requirements from the cue source. These requirements may be viewed from two points of view. If the cue source being discussed is a system that is under development, a potential user should influence its development to include qualities supporting their projected use. If, however, the cue source discussed already exists and is operational, analyzing its qualities (in relation to operational requirements) may lead potential users to a better understanding of the benefits and limitations of using that specific source in their system.

Requirements from the cue source are divided into the following categories.

1) *Availability:* The capability of the cue source to provide constant surveillance of specific areas. This attribute has both a technical side (time of deployment and operation, dependence on weather) and an operational/political side (the extent to which the cue operator is willing to commit the cue source for the needs of a specific cue user).

2) *Operational reliability:* The capability of the cue source to detect and supply up-to-date cue on all TBMs in its area of responsibility. This attribute is measured by two important parameters: false alarm rate and miss rate.

3) *Time delay:* The time gap between cue measurement and actual receipt of cue data. This factor includes both the calculation time in the cue source itself and the delays associated with dissemination of the data.

4) *Accuracy:* The errors in the cue data provided.

5) *Frequency:* The number of times that cue is provided and the time between consecutive reports.

6) *Type of data:* Data include position, full state vector, launch point estimation/impact point prediction, etc.

Table 2 connects between the possible uses of cues and the required attributes of the cue source. It is important to point out that the table includes the least severe requirement for each use. For example, in the case of "cue with search," the frequency requirement is one time. Obviously, a higher frequency will be useful, but it is not necessary as cued search can be conducted based on a one time report. Also, some of the requirements depend on the actual qualities of the systems discussed. In those cases, the methods of deducing these requirements were pointed out earlier in the appropriate sections.

B. Cue Receiving System Requirements

The requirements from the cue receiving system are stated in the form of required capabilities. These capabilities are mostly achieved through BMC algorithms and radar capabilities. Throughout this chapter, we have explained the need for these algorithms, and have hinted, to the best of our ability in the limited space of one chapter, to the principles governing their realization. Table 3 should be treated as a checklist where the specifics of each requirement depend heavily on the characteristics of the defense system(s) and the cue source. It is also important to note that the complexity of each required algorithm changes with the situation discussed. For example, a cue distribution mechanism that only has to deal with point defense systems, with no overlap in defense capabilities, will be much simpler than one developed to disseminate information between systems with large, overlapping footprints.

Table 2 Requirements from cue source

Requirements	Use					
	Early warning	Launch point prediction	Cue with search	Cue instead of search	Defense planning	Interceptor launch
Availability	Low	Low	Low	Very high	N/A	Very high
Reliability	High	High	High, to avoid wasting radar resources	Very high	N/A	N/A
Time delay	N/A	As soon as possible (to avoid launcher movement)	Depending on radar cued-search attributes	Like cue with search	Depending on defense plan algorithm requirements	Very small delay
Accuracy	Depending on size of population clusters	Depending on launch point prediction algorithm requirements	Depending on radar cued-search attributes	Like cue with search	Depending on defense plan algorithm requirements	Very good accuracy
Frequency	Once	Once	Once	Number of times/ continuous	Continuous	Continuous
Type of data	Alert	Track of launch point	State vector	State vector	State vector	State vector plot

Table 3 Requirements from the receiving system

Requirements	Use					
	Early warning	Launch point prediction	Cue with search	Cue instead of search	Defense plan	Interceptor launch
Cued search beam	N/A	N/A	√	√	N/A	N/A
Cue distribution mechanism	N/A	√	√	√	N/A	N/A
Defense plan logic based on cue	N/A	N/A	N/A	N/A	√	√
Radar load in presence of cue	N/A	N/A	√	√	√	√
Defense plan execution using cue	N/A	N/A	N/A	N/A	N/A	√
Multiple cue handling (data integration)	N/A	√	√	√	√	√

Tables 2 and 3, when treated on the basis of the analyses and observations provided in the other sections of this chapter, may be of assistance in selecting and implementing the proper use for external cueing in a given situation.

XI. Summary

External cueing is one of the mechanisms by which a defense system can be assisted by other forces or devices operating in its region. The basic commodity that external cueing has to supply is time. The time gained by receiving the cue may be used by the system in a number of ways, which have been defined and analyzed throughout this chapter.

The benefits that the system derives from external cueing may be expressed in terms of timing of critical processes, availability of resources at key moments, the ability to compensate for performance gaps caused by various counter measures, etc. We have shown a number of ways to quantify these benefits, which may be used for performing cost-effectiveness analyses of the various methods for using the cue.

These benefits, however, do not come without a cost. The cost of receiving external cues lies in having to implement the capability to treat them properly. This requirement, obviously, is more costly as the use of cueing becomes more complex.

The information provided in this chapter, along with the chapter dealing with interoperability, is intended to provide the reader with a mapping of the trade space involved in the complex problem of selecting and implementing appropriate mechanisms for connecting between defense systems. Exact solutions, however, should be defined by the reader based on specific situations.

V. Measures of System Effectiveness and Risk Management

Chapter 14

Analytical Methods, Measures of Effectiveness, and Simulations

Dror Cohen,* Ronia Lapid,† and Ayala Gur‡
WALES Ltd., Ramat-Gan, Israel

I. Introduction

THE present chapter is devoted to analytical methods and simulations employed in the development and analysis of TMD systems performance. Let us stress, at the outset, that it is not the authors' intent to provide an in-depth presentation of all known and familiar operations research techniques used in weapon systems development. Such a thorough treatment merits a book unto itself, and may not be of interest to prospective readers of this volume.

The goal we have set for ourselves is to highlight several techniques and issues that are either unique to theater missile defense (TMD) systems or in which problems and trends that may be relevant to future weapon systems developments are first brought to the surface through TMD analysis.

Indeed, it is the view of the authors that TMD, which was the last major development in weapon systems in the twentieth century (together with information warfare), highlights certain trends and problems that will characterize the future development environment of weapon systems, and hence is a good framework in which to discuss the required analytical methods and computerized tools.

Two main issues will be discussed in this chapter. The first deals with the question of what are the important measures that characterize TMD system performance—the so-called measures of effectiveness (MOE). In dealing with this issue we will identify several MOE specific to TMD. We will also attempt to establish a generalized view of the role of MOE based on the experience gained in their development for TMD, and to highlight some methods and ideas, which may be useful for other systems as well. The second issue discussed is the use of simulations and computerized tools for TMD systems analysis. We shall attempt

*President. Member AIAA.
†Executive Vice President. Member AIAA.
‡Senior Systems Analyst. Member AIAA.

to highlight some unique applications and analytical resolution of issues that arose in the study of antitactical ballistic missile (ATBM) systems, and point to possible generalizations for other applications in the future.

The goal of this chapter is, therefore, not to teach the reader specific approaches and implementations, but rather to clarify the logic and thought processes that brought about the resolution of specific issues, so that the reader may apply them to his or her problems, which may be different from our own. This chapter should be read in that spirit.

II. Measures of Effectiveness

A. General

Measures of effectiveness (MOE) are those quantities, typically achieved through various methods of data gathering (actual weapon system test, simulation run, analytic calculation, etc.), that are used for evaluating the performance of a specific element/weapon system/architecture in achieving a defined goal. MOE may be used to compare alternative defense architectures, deployment options, defensive strategies, firing doctrines, etc., and may also be used for determining optimal development goals.

Because this definition is broad almost to the extent of being useless, an attempt should be made to classify performance qualities into narrower groups, containing quantities that have some common denominator, making it easier to identify useable MOE. Such a classification will be attempted in the current section.

B. Classification of MOE

MOE may be classified in a variety of ways. The classification described here involves grouping at three levels, as follows.

1) MOE of level one: Basic quantitative "one on one" performance parameters. These include parameters such as weapon probability of kill [against a specific tactical ballistic missile (TBM) under specific engagement conditions], radar detection range, footprint of a certain weapon system (against a specific threat), etc. MOE of level one (sometimes referred to as MOP—measures of performance) are often used by engineers and scientists engaged in developing parts of the weapon system, or as building blocks for higher level MOE.

2) MOE of level two: Quantities that measure weapon system/architecture performance against a specific (many-on-many) attack scenario. Level two MOE are developed from level one MOE and the threat scenario chosen. The connection between levels one and two MOE is not immediate, but is affected by the decisions of the battle manager (e.g., deployment, rule of engagement). An example of a MOE of level two is the leakage rate (see further discussion to follow). MOE of level two are the ones most generally used in weapon system/architecture analysis and comparison.

3) MOE of level three: Quantities that measure attributes of the defense system performance, which are independent of the attack scenario (i.e., are inherent in the system itself) or an aggregate of MOE (level one or two) over an entire range of scenarios. Examples of such qualities are the system's robustness, flexibility, etc. These weapon system qualities, sometimes referred to as MOM (measures of merit), are usually not considered as part of the MOE spectrum, because they are

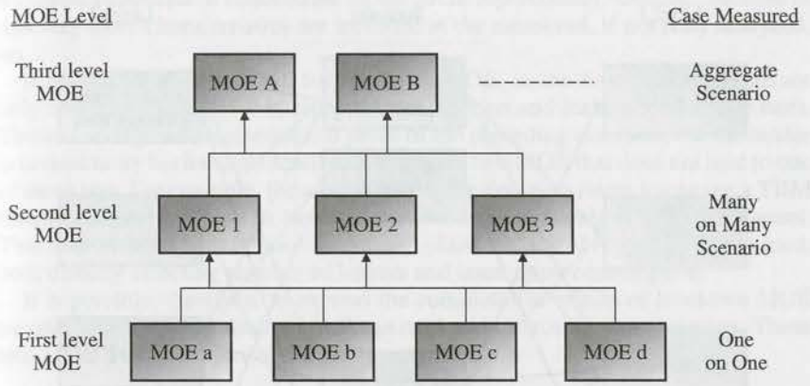

Fig. 1 General hierarchy of MOE.

thought to be nonquantifiable. In this chapter, we shall attempt to show that, at least for some of these qualities, this assumption is inaccurate.

Figure 1 illustrates the general hierarchy of MOE.

C. Level-One MOE as Building Blocks for Level Two

The level-one MOE are concerned with element or subsystem performance against a single threat. As such, they are useful in engineering and development applications, as demonstrated in other parts of this book. The focus of the present chapter is on the use of analytic methods and tools in the study and development of architectures, rather than subsystems. Our interest in level-one MOE stems, therefore, from their role in determining the architecture performance against a full-attack scenario. That is, their integration to level-two MOE.

Level-one MOE are not linked directly to the level-two MOE. Rather, they are linked through the set of operational decisions of the commanders and operators of the overall defense architecture. For example, a particular weapon system's probability of kill P_k against a particular TBM (level-one MOE) may be high, but if the architecture planner chooses to deploy, or use the system in such a way that its performance would be degraded, then this quality (the high P_k) will not be realized. In that case, to achieve a low leakage or leakage rate (level-two MOE), one would have to depend on the performance of a different defense component.

Figure 2 illustrates the connections between several level-one MOE, the operational decisions and threat scenarios, and the level-two MOE. For example, enemy doctrine affects the time line of specific events in the defender's architecture, because it affects the time line of the attack, and thus the sequence of events in the sky to which the architecture is expected to respond. The same MOE (time line) is also affected by the defender's rules of engagement (ROE), which dictate, for example, interception altitudes. Figure 2 is illustrative and does not in any way attempt to include all the level-one or level-two MOE.

Understanding the mechanisms governing these interconnections is essential for effective planning and operation of a defense architecture.

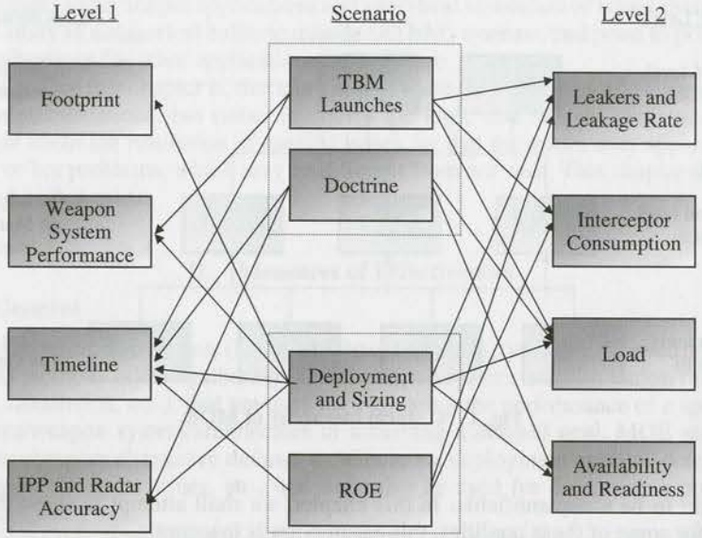

Fig. 2 Connections between MOE and operational decisions.

D. Level-Two MOE

The level-two MOE represent aspects of the performance of the architecture against a specific threat scenario. Level-two MOE are interconnected in various ways. Figure 3 illustrates these connections using several (obviously not all) level-two MOE.

Typically, the connection between level-two MOE is one of direct cause and effect. For example, radar load problems encountered during the attack may cause certain TBMs to pass undetected, thus increasing the leakers (and the leakage rate).

Even from this partial figure, it can be seen that the level-two MOE contain a certain hierarchy of their own, in the sense that some of them represent "causes" most of the time, whereas others represent "effects." This distinction is important. Level-two MOE are the ones most often used in architecture analysis. Because, most of the time, budget and timing constraints limit the resources devoted to analysis, analysts seek to concentrate on as few measures as possible. Obviously,

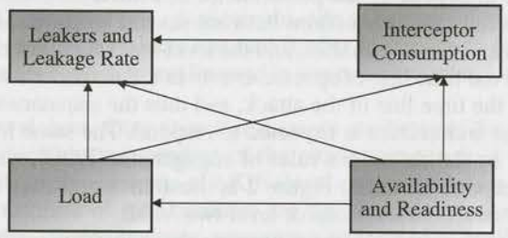

Fig. 3 Connections between level-two MOE.

it is most beneficial to concentrate on the MOE representing "effects," because in this way more characteristics are included in the measured, if not fully analyzed, set.

It is possible to show that, for level-two MOE, in the final analysis there are only two main "effects"—interceptor consumption and leakers (or leakage rate). There is no rigorous mathematical proof of the preceding assertion, but the reader is invited to try his hand at identifying any level-two MOE that does not lead to one of these two. For example, the availability of the defense system to engage a TBM threat may be expressed in terms of the resources available at a given moment. This may in turn result in poor/no defense plans, or lack of resources due to load, both directly affecting number of leakers and interceptor consumption.

It is possible, therefore, to express the commutative effects of level-two MOE by only two measures: leakers (leakage rate) and interceptor consumption. These two MOE are the subject of the next section.

E. "Natural" MOE—Leakage Rate and Interceptor Consumption

Our first step will be to define several basic terms and quantities. If we assume a simple defense architecture made of one weapon system defending one attacked asset, we may define the following basic parameters: 1) attack size N is the number of TBMs that attack the defended asset (DA), 2) leakers L is the number of TBMs (out of N) that hit the DA, 3) interceptor consumption IC is the number of interceptors fired during the attack, and 4) kills $-$ K is the number of TBMs (out of N) that are killed by interceptors.

Using these basic values, it is possible to calculate the following parameters. Leakage rate:

$$\mathrm{LR} = \frac{L}{N} \tag{1}$$

Interceptor consumption rate:

$$\mathrm{ICR} = \frac{IC}{N} \tag{2}$$

Effective interceptor consumption [(a variant of Eq. (2)]:

$$\mathrm{EIC} = \frac{IC}{K} \tag{3}$$

When the situation is made more complex by adding more weapon systems and DAs, the following variants are seen, in addition to the basic definitions:

1) L, or LR, per DA (or group of DAs).

2) IC, ICR, or EIC per weapon system per DA.

3) Weighing of interceptors according to "interceptor cost." Interceptor cost represents the value attributed to an interceptor type under given circumstances (not necessarily directly reflecting its price).

4) LR or L per DA, weighed according to the DA value (importance, vulnerability).

5) Any of the above, for a given subperiod of the attack scenario.

In any of the preceding variations, these MOE express a very "natural" view of the TMD war. A defense architecture is deployed to kill TBMs, and if possible, to

do so efficiently (i.e., using the least interceptors). Therefore, its success may be measured in terms of how well it achieves these two goals.

The difference between the quantitative and the "rate" versions (e.g., between L and LR) implies knowledge of the attack scenario (in this case, the total number of attacking TBMs). Obviously, the leakers L by themselves are of limited utility to the analyst if the entire attack scenario is not known (four leakers out of four attacking TBMs is one result, four out of four hundred is a totally different matter). The LR does not assume a knowledge of the scenario (a 4% LR means the same thing under any attack scenario: out of any 100 attacking TBMs, four are expected to leak). On the other hand, the defense commander does not care that the LR is, say, 10%. All he cares about is whether 1 or 10 TBMs hit the ground in his DA. This causes a situation in which the LR level considered acceptable by a decision maker/commander varies with the scenario. This makes the LR a difficult tool to use for comparison of architectures across a range of attack scenarios, though very important as a basic analysis parameter.

The same logic applies to interceptor consumption. The defense planner tries to minimize the rate of consumption of interceptors per TBM whereas the commander is interested in the numbers of interceptors remaining for future battles.

An important point that should be clarified is that, while the L (LR) and IC (ICR) are the two most commonly used MOE to measure architecture performance, they are not independent. In the simplest case of a single-system architecture, having full coverage and an unlimited number of interceptors, and with a ROE in which one interceptor is allocated per TBM, the relationship between IC and L is given by the interceptor P_k. Because in this case the expected number of leakers is

$$L = (1 - P_k)IC \tag{4}$$

F. Combining the "Natural" MOE—The Baseline System Effectiveness Approach

As has been stated, leakers and interceptor consumption are the most often used level-two MOE when analyzing architecture performance. Furthermore, these two measures "contain" all of the information needed to compare architectures although, naturally, it is sometimes necessary to turn to less all-encompassing level-two MOE to reveal information which is "hidden" within these two measures. Finally, as explained in the previous section, these two MOE are linked.

The idea behind the baseline system effectiveness (BSE) MOE is to construct a single measure, which combines all of the knowledge contained in level-two MOE about architecture performance against a given threat scenario. The need for such a measure arises for two reasons: First, to reach a simple and easily understandable comparison between two architectures, and second, To supply a single number, representing the results of level-two analysis, when constructing higher level MOE.

Based on the considerations outlined in the preceding sections, it is clear that the BSE should consist of a combination of L (LR) and IC (ICR). For the purpose of the present chapter, we shall use the L and IC as basic values, although a similar analysis can be performed using LR and ICR (mixing rates with actual values is, however, not advisable).

It is possible to define a L–IC plane, as presented in Fig. 4. The performance of a particular architecture against a specific threat scenario can be mapped as a single point on this "plane." The BSE is defined as the "norm" of the point on

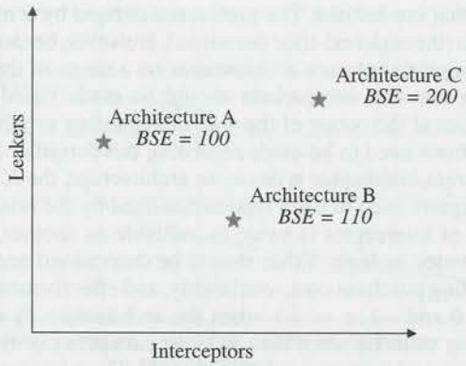

Fig. 4 BSE defined on the L–IC plane.

the plane. Because an increase in either one of the parameters (IC, L) indicates that the architecture performance is less effective, a smaller BSE value indicates a better architecture.

In general the norm between two partially dependent values such as L and IC includes the values themselves and a correlation factor. Therefore, the BSE is defined as

$$\text{BSE} = \sqrt{L^2 + IC^2 - 2rL \cdot IC} \tag{5}$$

where r is the correlation factor between L and IC.

Mathematically, this equation is true only if r is constant in all points of the L–IC plane. The relationship between IC and L is the effective P_k [Eq. (3)], and r represents this P_k. It is determined according to the actual P_k of the defense system throughout the battlespace on all the threats, the ROE used and the number of interceptors available. It is clear that r is not constant in the entire L–IC plane because, for large attacks relative to the number of interceptors, leakers will be caused by lack of interceptors as well as system performance limitations. As a result, the BSE is not actually a norm in the L–IC plane. However, it will be shown that the sensitivity of the BSE at a specific L–IC point, to changes in the value of r, is low. Also, the change in r with L and IC (given a specific architecture) is gradual and slow, and so for all practical purposes r may be viewed as a constant within any relevant section of the plane.

Equation (5) represents a basic version of the BSE. However, one more factor may be included, and that is the architecture planner's, or decision-maker's, preference between L and IC. If we designate this preference factor α, Eq. (5) becomes

$$\text{BSE} = \sqrt{(\alpha L)^2 + \left(\frac{1}{\alpha} IC\right)^2 - 2rL \cdot IC} \tag{6}$$

which is the final version of the BSE.

The value α represents the preference of the decision makers as to whether or not to save interceptors for later attacks at the cost of a less effective defense against the present attack, α may be deduced from the decision-maker's stated priorities and previous decisions. For a specific scenario, these decisions are seen in the defense

levels and ROEs that are defined. The preference defined by α may change during any given scenario (through real-time decisions). However, because we are defining a measure to distinguish between architectures on a range of threat scenarios, we may safely claim that this comparison should be made based on the decision-maker's preferences at the outset of the war, disregarding subsequent changes.

A few observations need to be made regarding this formula: 1) When there are a number of different interceptor types in an architecture, the *IC* is the sum of the number of interceptors used of each type normalized by the relative value of each type. If one type of interceptor is twice as valuable as another, its weight in the *IC* value will be twice as high. Value should be determined according to various parameters including purchase cost, availability, and effectiveness 2) r is a negative number between 0 and -1. $r = -1$ when the architecture P_k is 1. 3) $\alpha \in (0, 1]$ (i.e., α can take any value between zero and one, except zero). 4) Clearly, low BSE is good (i.e., better architecture performance). 5) The equation is not universal. Once the range of threats (scenarios) is defined, r is determined by the architecture. Different architectures may have different values of r.

Equation (6) is useless unless we define α and r in some meaningful way. For a given application (seeking an architecture to defend a specific country/region against a specified range of threat scenarios) r may be found by testing several candidate architectures against a representative subset of the threat scenarios in a battlefield simulation. The r value is the average P_k over all tested cases. Figures 5 and 6 show an analysis of r and α for a number of architectures (different numbers of interceptors) over a range of attack sizes. Using such graphs it is possible to choose the relevant r and α values for comparing architectures.

We will now analyze the sensitivity of the BSE to the specific values of r and α. From Figs. 5 and 6, representative values of r and α were chosen ($r = -0.6$, $\alpha = 2$). Figure 7 shows the equi-BSE lines for this case. To test sensitivity, the impact

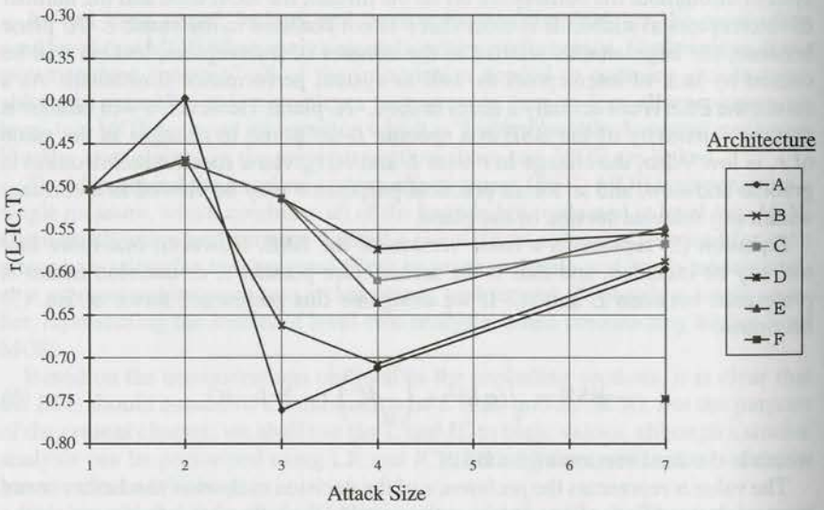

Fig. 5 Calibration for different architectures: r.

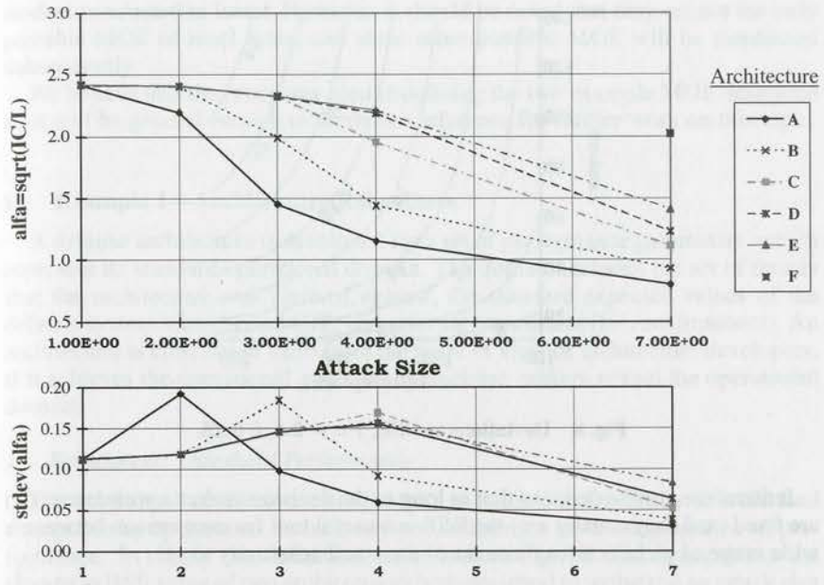

Fig. 6 Calibration for different architectures: α.

on the BSE if r or α were varied, was calculated. The possible values of r are between -0.8 and -0.4, which is a deviation of ± 0.2 relative to the chosen value. Figure 8 shows the maximal error in BSE caused by a deviation of 0.25 in r. As expected, little sensitivity can be seen.

The range of possible values for α is 1 to 2.5 (from Fig. 6). We repeat the same calculation testing $\Delta\alpha = 1$. This is presented in Fig. 9. We see a much greater sensitivity than in the case of r.

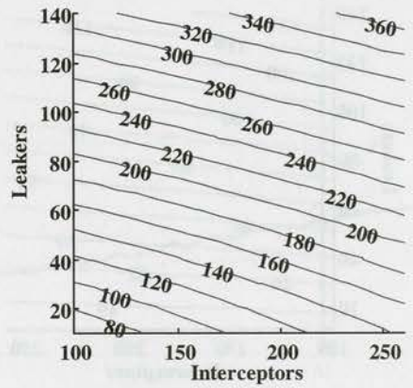

Fig. 7 Equi–BSE lines.

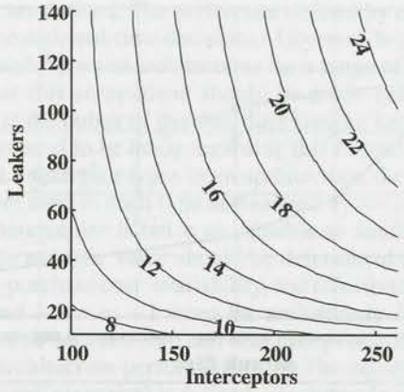

Fig. 8 Deviations in BSE, $r = -0.6 \pm 0.25$.

It therefore may be deduced that as long as the decision-maker's preferences (α) are fixed (and they usually are) the BSE is a useful tool for comparison between a wide range of architectures (because of the small sensitivity to r).

G. MOE of Level Three

MOE of level three are quantifications of general qualities of the architecture. They should reflect either architecture performance over a large set of attack scenarios or architecture potential performance (i.e., independent of the scenario). Third-level MOE, being quantifications of abstract terms, are difficult to define and implement, but are important for gaining a clearer understanding of the mechanisms governing the architecture performance and thus also architecture design. In the current section, two such MOE will be presented in some detail as examples. These MOE (robustness and flexibility) were developed and used in architecture

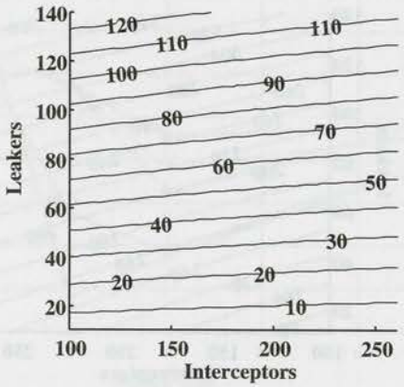

Fig. 9 Deviations in BSE, $\alpha = 2 \pm 1$.

studies conducted in Israel. However, it should be noted that they are not the only possible MOE of level three, and some other possible MOE will be mentioned subsequently.

We believe that the processes used in defining the two example MOE described next will be general enough to serve as a reference for further work on this topic.

H. Example 1—Architecture Robustness

A defense architecture is developed for a set of performance parameters, which represent its standard operational domain. The domain includes the set of threats that the architecture was planned against, the standard expected values of the defense system's level-one MOE, and external parameters (i.e., environment). An architecture is considered valid from the point of view of architecture developers, if it achieves the operational goals set by decision makers within the operational domain.

1. Robustness: Threshold Performance

The robustness of a defense architecture is defined as its ability to withstand changes in its operational domain without suffering a severe degradation of performance. To clarify this definition, let us look at Fig. 10. The figure shows the change in BSE value of two architectures both designed to withstand an attack size of up to M attacking missiles (the figure is notional and does not represent results for any particular architecture).

We indicate as BSE_t the highest BSE value that the decision makers are willing to tolerate, i.e., it represents the operational requirement. Clearly, both architectures in this figure are valid, because they achieve the required performance over the defined operating domain (up to M attacking missiles). However, when we exceed

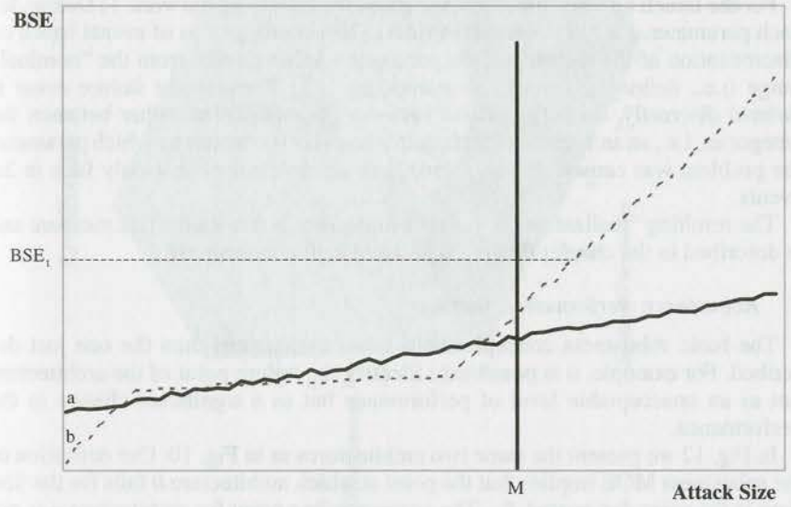

Fig. 10 Architecture BSE as a function of attack size.

the expected number of attacking missiles, architecture a continues to function adequately, whereas b experiences a rapid deterioration in its performance. We say that architecture a is more robust than architecture b, with respect to attack size.

The robustness MOE is based on decision-maker–defined threshold requirements of the defense effectiveness expressed in terms of BSE, LR, or any other suitable measure. When an architecture ceases to be able to achieve the designated threshold performance, we say the architecture fails. The MOE is based on the ratio between the number of events (or parameter values) tested and the number of events in which the architecture failed.

$$\text{Robustness}\,(A) = \frac{\sum_{\text{all events}} P_{\text{event}} \cdot I_{\text{fail}}}{\sum_{\text{all events}} P_{\text{event}}} \tag{7}$$

where

A = the tested architecture
P_{event} = the probability of event occurrence (parameter value)
I_{fail} = 0 if the architecture does not fail in this event and 1 if the architecture fails in this event

From the equation it is clear that a lower value for the robustness MOE indicates a more robust architecture. Figure 11 presents the robustness as a function of two different concurrent parameters. Though the peak performance is identical for both architectures, A has a higher effectiveness over a wider range of parameters. Therefore, according to the measure, architecture A is more robust than architecture B.

To actually implement the robustness measure, two questions have to be answered: 1) How to define the probability of each event or parameter value. This needs to take into account the likelihood of unexpected events. 2) How to prioritize between failure events caused by different parameters.

For the Israeli architecture study, we chose the following answers: 1) Define, for each parameter, a set of "nominal" values. Then create groups of events based on discretization of the distance of the parameters value chosen from the "nominal" range (i.e., define 1α events, 2α events, etc.). 2) Because the failure event is defined discreetly, do not prioritize between parameters but rather between the categories, i.e., an architecture that fails in a 1α event (no matter by which parameter the problem was caused) is less robust than an architecture that only fails in 2α events.

The resulting "realization" of the robustness idea is termed the risk measure and is described in the chapter dealing with architecture assessment.

2. *Robustness: Performance Stability*

The basic robustness concept admits other realizations than the one just described. For example, it is possible to identify the failure point of the architecture not as an unacceptable level of performance but as a significant change in the performance.

In Fig. 12 we present the same two architectures as in Fig. 10. Our definition of the robustness MOE implies that the point at which architecture b fails for the first time is the point designated P_f. The corresponding point for architecture a is not seen in the figure (it may occur at an attack size greater than shown in the figure).

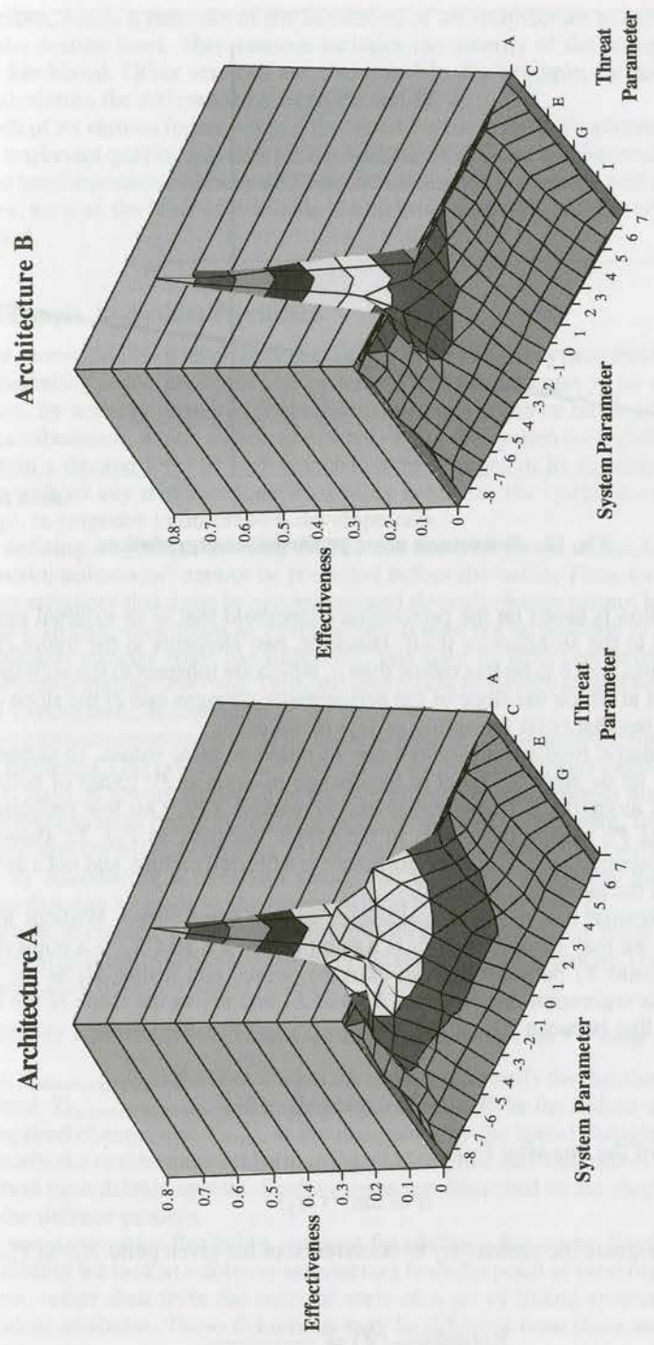

Fig. 11 Multiparameter robustness.

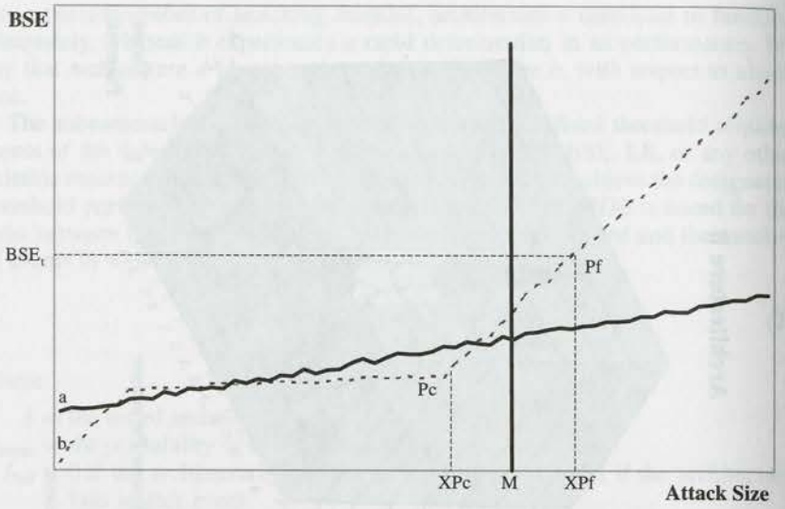

Fig. 12 Robustness using performance degradation.

The definition is based on the performance threshold that is an external number, not related to the architecture itself. However, two elements in the figure clearly define architecture b to be less robust than a, which are inherent to the architecture: 1) the point at which the slope of the performance changes and 2) the slope of the line following that point (steep line = less robustness).

An alternative robustness measure may be based on these values. To define such a measure, let us mark the point of the change in slope as Pc (point of collapse), its location along the X (parameters) axis is marked XPc. Our first problem will be to locate Pc among the points in the graph, because, as Fig. 12 shows, the graph representing architecture performance is a blurred picture, and not a uniform function of the parameter changed.

Let us assume that n points (parameter values) were tested. Without loss of generality, we may assume that X_1 is a point below X_{pc} and X_n is a point above. Take any point X_i between the two extreme points, and define M_1 as the slope of the linear regression line between X_1 and X_i and M_2 as the slope of the linear regression line between X_i and X_n.

Then,

$$X_{pc} = X_i \mid \max(M_2 - M_1) \tag{8}$$

The angle of the line after Pc is

$$\alpha = \tan^{-1}(M_2) \tag{9}$$

If we designate the probability of occurrence of the given point X_{pc} as P_{xpc}, we may define

$$\text{Robustness } (A) = \frac{1}{P_{xpc} \cdot \sin\alpha} \tag{10}$$

As before, this is a measure of the likelihood of an architecture to behave at less than the desired level. This measure includes the severity of the collapse as well as its likelihood. Other versions are also possible, for example, by inserting into the calculation the difference between Pc and Pf.

In all of its various forms, we feel that the robustness measure allows evaluation of an important quality, relevant for the analysis of defense architectures. The particular implementation chosen will depend on analyst preference and on external factors, such as the level of available simulations and tools and the types of data supplied.

I. Example 2—System Flexibility

The second MOE of level three that we are presenting is system flexibility. System flexibility is the ability of the system to adapt to changes in its operational domain, by actively changing its performance parameters or mode of operation. Unlike robustness, which measures to what degree the system (or architecture) can maintain a desired level of performance under changes in its operating environment, without any modifications, Flexibility measures the system's capability of change, in response to unforeseen developments.

In defining a system flexibility MOE, one must be aware of the fact that the "unknown-unknowns" cannot be predicted before the battle. Thus, the full range of circumstances that must be accommodated through change cannot be known in advance. In fact, even when an event occurs, it may not be possible to understand immediately exactly what has happened, and only its effect on defense system performance may be observed. Thus, the analysis of flexibility should be performed at the system level, independent of the scenario.

Two more observations should be made before we define an MOE for flexibility. First, because a TMD system must operate during a battle that may extend over only several days, it is very desirable to be able to implement a change that improves the defense system performance as quickly as possible. Therefore, time must be factored into the MOE definition. In our analyses, we chose to deal with the timing issue by introducing a threshold value, that is, system flexibility measures the change that can be made to the system within a threshold time T_t.

Second, to change the system to improve its performance, one must understand why a change is necessary, and to identify the change that is required. This sequence of events may be described as follows:

$$\text{Flexibility Requirement} = T_{\text{fault-recognition}} + T_{\text{change-identification}} + T_{\text{change}} < T_t \tag{11}$$

where $T_{\text{fault-recognition}}$ is the time it takes the system to identify the fact that a fault has occurred, $T_{\text{change-identification}}$ is the time required to analyze the failure and identify the required change, and T_{change} is the time taken by the actual changing process.

Clearly, the requirement set forth in Eq. (11) implies the existence of an efficient near-real-time debrief system. Such systems are discussed in the chapter dealing with the defense process.

To measure system flexibility, we must first define a few terms. For the analysis of flexibility we look at a defense architecture from the point of view of the defense process, rather than from the point of view of a set of linked systems and their individual attributes. These definitions may be different from those used in other analyses.

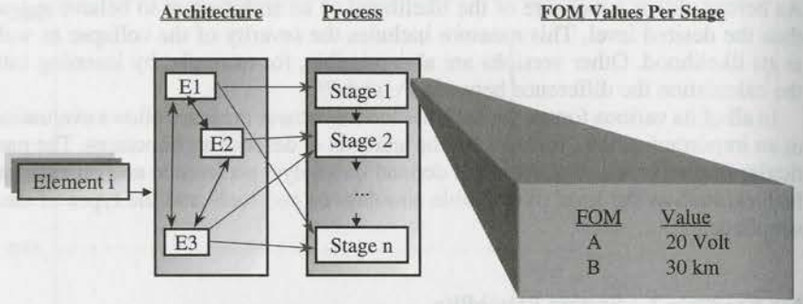

Fig. 13 Architecture, process, and FOM values.

An architecture is defined as a linked set of systems that perform an integrated function. The flexibility is analyzed in accordance with the process and subprocesses involved. A process is defined as a series of steps that can be measured by a particular figure of merit (FOM)—a FOM is usually a level-one MOE. For example, a FOM for the detection of a TBM is the detection range (average); another possible FOM for the same process is the detection probability. Note that for the analysis of flexibility, it is immaterial whether one or more sensors perform the detection. The value of the FOM is what matters. Figure 13 presents the general concept. Each architecture element participates in a number of stages of the process. At each stage, the element has required values (or range of values) and an actual value for the various performance parameters.

The analysis of flexibility now becomes an analysis of the FOM. If we analyze the entire defense process and come up with a series of FOM that are not scenario dependent, and represent the important subprocesses, it is possible define the flexibility of each FOM as the maximum change capability given a feasible change in the relevant elements. The system flexibility is limited by the FOM that can be changed the least:

$$\text{Flexibility } (A) = \min\{\max[\text{change } (f)]\}_{f=\text{FOM1},}^{\text{FOM}_n} \tag{12}$$

To use Eq. (12) we must 1) identify the scenario-independent FOM for each subprocess of the architecture; 2) find the feasible changes in the defense system that affects the FOM; and 3) find the maximal change possible in each FOM.

The analysis leading to the achievement of the these goals will be specific to the type of systems that compose the architecture analyzed. We shall give one example of such an analysis.

In Fig. 14, the process of engaging a TBM by a fragmentation warhead interceptor is analyzed. The leading FOM for this process is the system battlespace. This FOM is analyzed, identifying two nonscenario dependent FOM and the system elements that can affect them: fragment velocity and total impact energy.

To determine the impact of changes in element values on the FOM it is necessary to test variations of the values. Table 1 was generated using a fusing simulation and relevant intercept parameters. It provides representative results for the amount of change possible in each element and its effect on the FOM.

The table shows, for each parameter of the interception end-game, how much a change in its value from the nominal value required will affect each of the various FOM identified. For example, a decrease of 10% in the fusing time (fusing

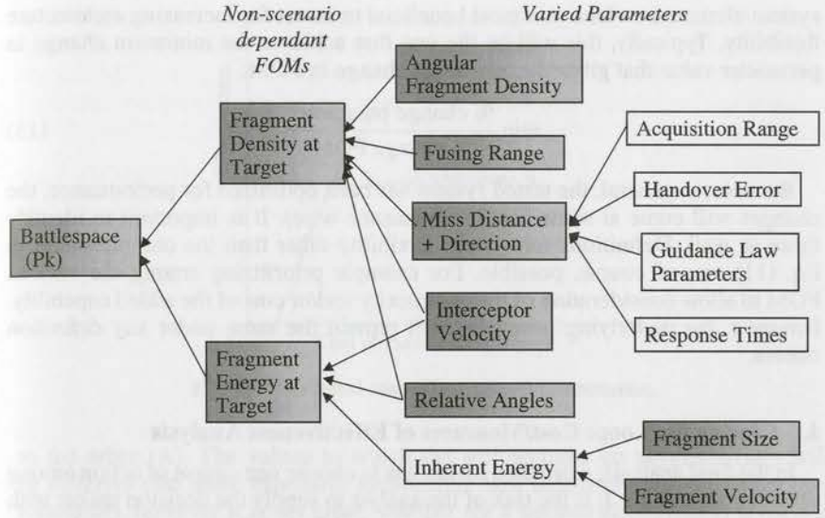

Fig. 14 Battlespace breakdown to FOM (fragmentation warhead).

earlier by 10% relative to the required fusing time) causes a change of 37% in the fragments density at the required impact point.

There are two ways of looking at Table 1. First, find the maximum change in FOM available and return to Eq. (11) to calculate architecture flexibility. In some cases, depending on the FOM, any positive change is clearly beneficial. In other cases it is the absolute size of the change that is relevant. Second, identify the

Table 1 Representative results—FOM change as function of parameter change

Parameter	Change, %	Fragment density change, %	Total energy change, %	Maximum energy change, %
Velocity	−10	2	−6	−8
	10	20	27	6
	20	7	11	8
	30	11	15	28
Angle	−10	0	−26	−26
	10	24	57	25
	25	0	57	57
	35	−5	58	67
Fusing time	−20	50	57	4
	−10	37	39	0
	10	−28	−29	−1
	20	−56	−56	−8
MD	−10	−7	−6	−3
Distribution	+10	15	18	3
	+20	1	5	−10

system element in which it is most beneficial to invest for increasing architecture flexibility. Typically, this will be the one that achieves the minimum change in parameter value that gives the maximum change in FOM:

$$\min \frac{\%\text{ change parameter}}{\%\text{ change FOM}} \tag{13}$$

Because, in general, the tested system has been optimized for performance, the changes will come at some cost (performance-wise). It is important to identify these as well. Definitions for system flexibility other than the one presented in Eq. (11) are, of course, possible. For example prioritizing among the various FOM to allow consideration of the complexity and/or cost of the added capability. However, the underlying principles will remain the same under any definition chosen.

J. Closing the Loop: Cost/Measures of Effectiveness Analysis

In the final analysis, a decision maker has to choose one course of action among several possibilities. It is the task of the analyst to supply the decision maker with tools to assist in arriving at his or her decision. Most of the time, the mechanism employed in arriving at final decisions is cost-effectiveness (cost/MOE) analysis. The cost of an architecture can be defined in a number of ways. Each definition is appropriate depending on the question being asked.

The BSE MOE that was previously discussed in this chapter includes the *cost of consumed elements* (interceptors). This is appropriate when comparing different methods of employing a given architecture (deployments, ROE) or comparing architectures with similar elements in which the purchase cost is not a main consideration. In principle, the BSE approach can be expanded to include any type of consumed entity. However, when comparing the merits of one architecture vs another, or when deciding on the sizing (the number of purchased elements) of a given architecture, the other elements of cost come into play, including the development cost and purchasing cost.

When considering purchase cost we have to take into account both the cost per element and the number of elements purchased. Let us look first at the number of purchased elements. For a fixed threat using any relevant effectiveness measure, the resulting graph of system effectiveness vs acquisition cost will appear as the one seen in Fig. 15. The graph is asymptotic, which means that above a certain number of elements, the contribution of any additional element becomes marginal. The effectiveness measured cannot be significantly improved with more of the same, considering the reference threat. It is important to note that in reality it is not possible to calculate the asymptotic effectiveness of each element of the defense architecture separately. For example, let us look at launchers and interceptors. If you buy too many interceptors, there will not be enough launchers to launch them at the given scenario time, and their incremental effect will be nullified. However, taken in combination, the effect of Fig. 15 will be clearly seen and the most cost effective mixture of elements can be identified.

We next consider the cost per element of an architecture. In suitably developed architectures there is a relationship between the cost of an element and its performance; usually a more expensive element will perform better. Figure 16 compares two architectures, one that is more expensive (B) to purchase relative

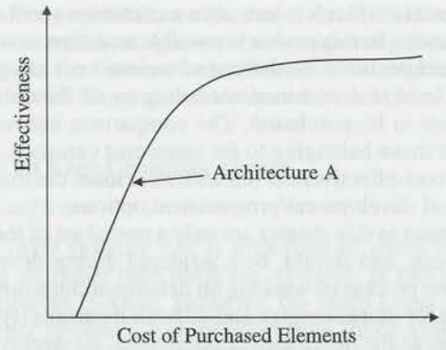

Fig. 15 Notional cost-effectiveness phenomena.

to the other (A). The values to which the architectures go asymptotically will be different (the more expensive architecture will ultimately provide better performance), however it is not clear, whether for a desired effectiveness level it is necessary to buy more quantities of one architecture, or fewer quantities of a more expensive architecture.

It should be mentioned that the picture depicted in Fig. 16 also occurs when we look at purchased elements vs other MOE (such as robustness) that do not directly measure architecture performance. However, the exact shape of the graph will be different.

The full cost of the architecture includes, along with the purchase cost, the non-recurring development costs. These costs are important to consider when determining which architecture option to buy or how to further develop the architecture. Basically, two approaches exist to compare architectures when the full cost is taken into account:

1) Integrating the development costs of each architecture into the previously described purchase cost by setting the initial cost of each architecture according to the non-recurring costs. In this case it is possible to compare different cost and effectiveness levels of the same architectures.

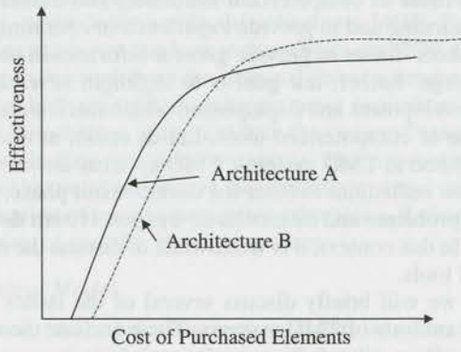

Fig. 16 Notional cost-effectiveness phenomena—comparison.

2) In many cases it is difficult to calculate a continuous cost effectiveness function for an architecture. In this case it is possible to define several cost categories and the different architectures are defined at various cost categories. The defined architectures cost level is determined according to all the development required and all the elements to be purchased. The comparison between architectures is performed only for those belonging to the same cost category.

The analysis of cost-effectiveness (cost/MOE) closes the loop and allows intelligent comparison of development/procurement options.

The MOE presented in this chapter are only a partial set of the qualities of TMD architectures that can, and should, be considered during development, procurement, and use. In the process of working on defense architecture development, we found that the need for more complex and difficult measures (typically, level-three measures) increases as the development progresses. As deployment of these systems nears, the need will arise to study complex qualities, which may be paramount for the correct utilization of the systems, either in their originally intended role, or in possible secondary roles. Some qualities of TMD architectures that should be considered for MOE development are deterrence value, diversity (i.e., the possibility of dealing with non-TBM threats, such as cruise missiles), interoperability quality (i.e., the degree to which an architecture is capable of interoperability with friendly systems), and combat readiness.

The field of MOE for TMD systems (beyond the most commonly used leakage rate) remains largely an uncharted territory, never before treated in a carefully laid out, systematic approach. This is true for other systems as well. We hope that this chapter will constitute an initial step to fill this gap. It is certain that a great deal more work needs to be done before a fully systematic theory of MOE can be created.

III. Computerized Models and Simulations

Computerized models and simulations of various levels are today an essential part of any weapon system development process. The use of computerized tools starts with checking the basic idea for the new weapon system in battlefield models, to assess its possible contribution, and continues throughout the development process as engineering models, system level representations, and battlefield level tools are used by the developers and analysts. Even after development is completed, widespread use is made of computerized simulators and trainers, to facilitate operational crews' learning and to provide experience in operating the system.

It is not the authors' intent to provide general information about computerized tools and their usage. Rather, our goal is to highlight several aspects of TMD weapon system development and deployment, which merit somewhat unorthodox and innovative use of computerized tools. Let us stress, at the outset, that these aspects are not limited to TMD systems. TMD systems are one of the first major systems for the new millenium to enter the development phase, and thus allow us a look at the new problems and dilemmas of weapon system development for the future battlefield. In this context, it is worthwhile to discuss the role of simulations and computerized tools.

In this section we will briefly discuss several of the issues that have special importance in the analysis of TMD systems. These include the ways to handle an uncertain, ever changing threat, the use of simulations in preparing for interoperability between weapon systems, the importance of analysis of near-real-time decision-making effects on battle results, and the role of simulators and trainers in maintaining system readiness against a rarely appearing (low likelihood) threat. A brief discussion on the future of simulations as interoperable entities, using DIS/HLA protocols, will be presented at the end this chapter.

A. Dealing with an Ever-Changing Threat

1. *Threat Characteristics*

One aspect of TMD systems that is important to stress is that they are, by nature, responding to, rather than initiating, systems. In this sense they are akin to surface-to-air missiles (SAMs). However, the much larger operating envelope (at least of upper-tier systems), and the short time frame available for response, create a unique response requirement from the system. The TMD system has to handle a situation in which the threat generation occurs far away, and the enemy has almost complete freedom in choosing the timing and characteristics of the attack (launch locations, salvo size, and distribution) as well as a wide latitude in determining the characteristics of the threat itself (payload, trajectories). This is particularly true when discussing isolated operations by terror organizations, or attacks mounted as part of a conflict in which the country attacked by TBMs is not a participant (as happened to Israel during the 1991 Persian Gulf War). The TMD system must therefore be able to respond to an unknown, uncontrolled threat. Moreover, because the TMD systems development processes are widely reported in the mass media, including technical details, the system has to be prepared, from the start, to deal with responsive threat, i.e., the means employed by the enemy to counter the effect of the defensive system. The rate at which various responsive threats can be developed and the specific development options that will be chosen by the enemy are unknown to the defense planner. However, it may be possible to define a maximum capability based on existing technologies and enemy engineering and economic capabilities.

The preceding discussion leads us to conclude that in analyzing a defense system's performance, it is important to take into account the numerous dimensions of the threat. This goes against the usual practice of defining a few highly detailed reference attack scenarios and using them throughout the development and evaluation process.

There are several techniques for dealing with such threats. One technique was already discussed, when dealing with the robustness MOE. Returning to Fig. 9, and the related discussion, we can look at the robustness MOE as a way of systematically analyzing excursions from the nominal threat scenarios. In this way, it is possible to analyze both single and multiparameter excursions. Another method, discussed later in this chapter, is to introduce the "enemy" into the simulated environment, i.e., conducting two-sided wargames. This approach, while very beneficial from an analytic point of view, is too cumbersome and costly to be used for a systematic evaluation of various parameter excursions. Rather, it should be used for in-depth analysis of specific, interesting cases.

2. *Statistical Threat Model*

Another approach, and one that has been successfully utilized in Israeli studies, is that of creating a statistical representation of the threat. The main idea behind this approach is to create a computerized tool that will automatically generate

reasonable attack scenarios. The term "reasonable" is taken to mean scenarios that will be considered plausible from the point of view of an intelligence specialist. This, in turn, means that not only all of the parameters of an attack fall between well defined boundaries but also that the combination of all parameters makes sense from an intelligence point of view. A simple example will serve to clarify this point. Suppose that the random threat generation model draws n missiles as the enemy's initial stock. Then, although it is possible to draw $m > n$ launchers as initial stock, that will not be a reasonable scenario.

In the model actually designed, some numbers are drawn randomly using distributions defined by intelligence experts, whereas some others are derived from a database whose point of entrance is determined by the random draws. For example, in our particular realization, the number of available missiles and launchers in certain enemy countries is fixed by an intelligence assessment. The number out of those that participate in a given attack is drawn, based on the probability of participation by each enemy country and possible combinations of participants, that were provided by intelligence assessment.

Statistical threat modeling allows the generation of many reasonable attack scenarios that can be used in the analysis. In our realization, the statistical threat model has been linked to a battlefield simulation and used to study the performance of a given architecture. Notional results of this analysis are presented in Fig. 17.

The figure presents the number of leakers suffered by the architecture under different attack scenarios, arranged by one particular parameter of the attack (in this case, attack size). Each point on the graph represents the result of one or more scenarios. The analysis was performed on 1000 scenarios generated by the statistical threat model. The parameters that were varied between the scenarios include attack size, attack timing, and attack distribution (launch and impact points). It can be seen from the graph that there is a general trend in the behavior of the defense system—as the attack size increases the number of leakers increases. For

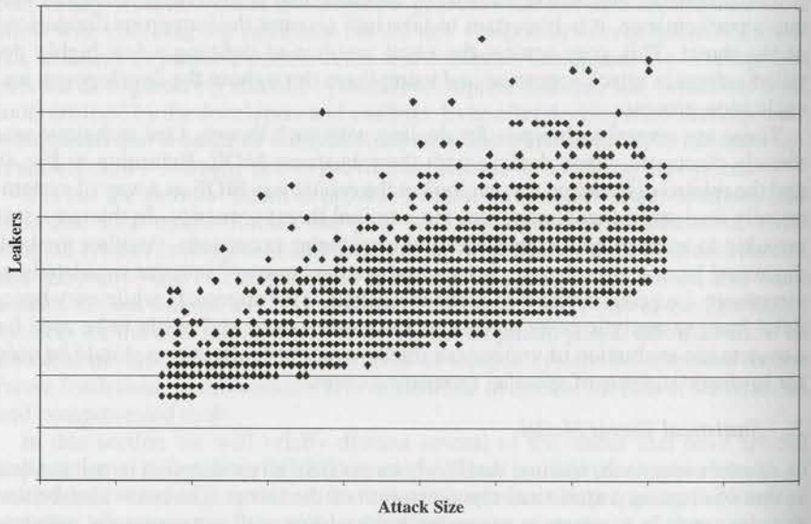

Fig. 17 Example of the use of a statistical threat model.

any attack size there are cases when the resulting number of leakers is higher than for other cases. This is caused by attack parameters other than attack size (e.g., distribution of TBMs between the targets).

Looking at Fig. 17 and the data used to generate the graph, the many possible uses and the amount of information revealed by the use of a statistical threat model become apparent. One possible use is to test for the number of events in which the architecture fails to achieve a certain threshold performance (as is defined in the robustness MOE). Another use might be to identify the points in which a particular combination of parameters causes a severe deterioration in performance (the "spikes" in the figure) and to analyze them in depth.

The issue of systematically describing a many faceted and rapidly changing threat is something that should be dealt with by developers of weapon systems for the future battlefield, both in TMD as in other areas. The approaches suggested here might serve as a starting point for such a treatment.

B. Computerized Tools in Interoperability Analysis

1. *Interoperability Problem*

The TMD weapon system is characterized as being, in itself, a "system of systems," that is a set of systems (radar, battle management center, interceptors) that work together to perform a specific mission (for example active defense against a TBM threat in the reentry phase). The TMD architecture is characterized from the outset as being a "family of systems," that is, a group of weapon systems, each of which performs a specific defensive mission that contributes to the same overall goal. In our case, the goal is TMD and the missions include passive defense, active defense in boost, midcourse, and reentry phases, as well as counter force. This characterization implies more than mere semantics. Most other defense systems either interact by mutual noninterference (i.e., predefined division of areas of responsibility) or mutual assistance against a set of threats. The level of data-sharing is at most a common battlefield picture. TMD systems are planned for *continuance* of the defense effort against a single threat (i.e., upper and lower tier interception of the same target) and are meant to share (in one method or another) all of the relevant interception information.

This very tight interoperability between weapon systems, both in time and in mission, needs special handling if performed within a single defense force, but especially in a situation of coalition warfare. This situation, in which a defense architecture that has a single mission comprises a number of separate systems, linked by data and/or vocal connections, and sometimes employing a dual command structure, calls for very elaborate definition of intersystem links, and for continuous training. Situations involving coalition warfare bring into the "family of systems" problems related to different command structures and rules, different understanding of the activity going on in the battlefield, and even language barriers. All of this in a system that has to collaborate in an interception process lasting at most only several minutes and sometimes only a few seconds.

2. *Interoperability Study Tools*

The appropriate use of simulations and computerized tools can significantly contribute to dealing with these issues. The most important of these are human-in-the-loop (HIL) simulations. Our experience in this field is based on

running the Israeli test bed (ITB)* in a number of experiments, both with and without coalition (U.S.) partners. The use of a test bed with extensive HIL capabilities to conduct interoperability exercises enables analysis of the aforementioned interoperability issues and an evaluation of various possible solutions. These include coordination assistance tools, threat information sharing (TIS) techniques, roles of operators of the different weapon systems, etc. Any simulators used for this purpose should include adequately accurate representations of all the weapon systems involved, including the HIL capability.

This type of operation can either be carried out through the use of a single, highly complex, simulation, or by connecting a number of specific system simulations through distributed integration simulations/high level architecture (DIS/HLA) (see later section). Working on interoperability in HIL simulations requires several unique analysis techniques, because all of the verbal communication between the systems and all the decision-making processes occur outside of the simulators. For example, in an exercise with human participation the reasons for the failure to intercept a TBM include, in addition to architecture effects (resources, P_k, etc.), the potential impact of the human participants. Leakers can result from wrong coordination, command decision, or faulty operation. It is almost impossible to determine these types of causes using recordings of the simulation because only the performed actions are recorded and not the reasons for the actions. Recording the voice coordination channels provides some assistance for this analysis. However the need to synchronize between the voice and data files and the difficulty in efficiently analyzing the voice channels makes this task virtually beyond the capabilities of a reasonably sized analysis team.

3. *Use of Observers*

An alternative method, developed and used during numerous ITB experiments, involves the employment of trained observers, sitting at key points in each weapon system operating post, monitoring the coordination process while it occurs, including the generation of real-time reports. In this way the coordination between the decision process and the actual events is immediately provided, enabling the analysis of the impact of human intervention on the battle performance. A typical setting of such an experiment is depicted in Fig. 18.

In this case the coordination between two different weapon systems is tested. The command structure of each weapon system battle management center (BMC) includes a commander, various intermediate functions including assistant commanders, work managers and specialized officers, and finally coordinators. The lowest level trained is the execution and coordination level where the actual launch and hold fire commands are performed. There are two different communication links between the two systems. A data link is used to automatically transfer data for the creation of a unified sky-picture. A voice link is used for coordination. The commanders generate policy decisions and guidelines based on the battle picture. These are transferred to the coordinators via the various intermediate functions. The task of the coordinators is to execute the defense based on commander guidelines. This includes coordinating the defense with additional weapon systems if necessary.

The location of the observers is determined according to the analysis goals and is one of the important parameters to be determined during the planning of the

*The ITB is a product of Tadiran Electronic Systems, Ltd., Israel, and is jointly developed and operated by the U.S. and Israeli governments.

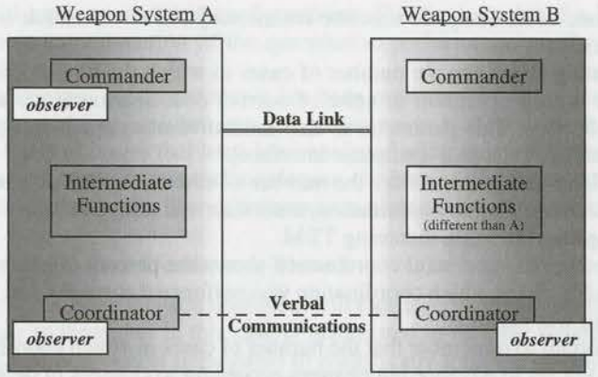

Fig. 18 Typical interoperability experiment setup.

simulation runs. Because the purpose of the analysis (in this example) involves the impact of the commander's guidelines and the coordination on the defense performance, observers are placed at the commander and coordinator levels only.

The observers record the exchange of verbal coordination between the coordinators on each track together with appropriate battle-picture data (track numbers, time, and events). These observers later form a part of the analysis team and can, based on comparison of their notes, direct the focus of the team to specific events, thus avoiding the need for elaborate search among piles of recordings and computer reports. The participants in each experiment are also interviewed about their perception of what transpired immediately after the run, to analyze their situational awareness and gain insight into the possible methods of improving performance. Typically, participants are trained military personnel.*

4. *Interoperability Measures*

In analyzing interoperability, particular emphasis is placed on the measurement of quantities that are significant in determining the effectiveness of the coordination process. The purpose of the coordination process is to generate an appropriate defense plan for each TBM. The appropriate defense plan is not fixed throughout the battle, but changes according to command-level decision making. The commander's decisions are based on battle development parameters including damage actually suffered during the combat, remaining inventory of weapons per ATBM system, intelligence information, etc.

The measures for coordination efficiency should therefore be based on the number of cases (attacking TBM) for which it succeeded/failed in bringing about the desired defense plan. Such measures include the following.

1) Coordination errors are the number of cases where the coordination resulted in a defense plan that was contradictory to the prevailing policy at this point in the run.

2) Timing problems are the number of cases in which the coordination worked correctly, but time line problems prevented carrying out the resulting defense plan

*For the development of a large part of this methodology, we are indebted to Dr. Colin Kessel, of PAMAM, Ltd.

(the existence of an adequate timeline for interception coordination is of utmost importance).

3) Operating errors are the number of cases in which the human operator performed the wrong operation in spite of correct coordination (such as pressing the wrong button). This parameter is also indicative of operator inexperience or inappropriate HCI (human–computer interface).

4) Double interceptions reflect the number of cases in which as a result of incorrect coordination or no coordination, more than one weapon system incorrectly launched against the same incoming TBM.

5) Percentage of successful coordination shows the percent of cases (out of all the required cases) in which coordination was performed correctly and the correct defense plan was executed.

It is important to remember that the number of cases in which coordination was required is not set in advance but changes according to changes in defense policy and therefore should be monitored by the observers during the run.

To reach relevant conclusions in these analyses, it is necessary to perform a number of runs with different operators and with different scenarios. In most cases the results are reported as averages over all of the runs performed.

5. *Subjective Workload Analysis Tool*

Figure 19 depicts another important analysis tool employed in interoperability analysis in the Israeli test bed, the subjective workload analysis tool (SWAT). The SWAT is a measure of the level of workload a particular operator felt during the run. At the end of each run, the participants' reply to a specially designed questionnaire that includes questions about perceived workload and perceived degradation of performance. The SWAT number is arrived at according to the participants' answers. By comparing the SWAT of different command configurations as reported by different subjects performing different functions, it is possible to determine which configuration places the least stress on the operators and what functions

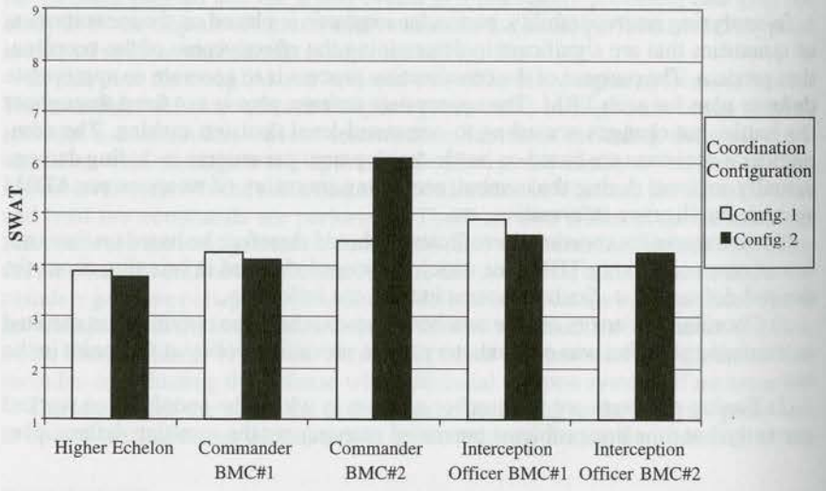

Fig. 19 SWAT example.

need special tools or modified configurations. The final results of a HIL experiment include a combination of the quantitative performance results, observers, reports and the SWAT.

The methodology presented above has been extensively used in interoperability exercises, yielding insights into the proper ways to deal with this complex issue. The types of issues that have been analyzed with this methodology include BMC structure, coordination methodologies, the benefit of various computerized coordination tools and the flow of information within the command structure.

C. Training for Rare Occurrence Threats

TMD systems, once they are deployed, will face a very difficult problem, which is not unique to them, but in this case is manifested in its most severe form. The problem is that it is almost impossible to realistically train the operators of such systems in the performance of their combat tasks. In fact, it is impossible to perform such training systematically at regular intervals.

A number of reasons for this problem may be cited: The enormous cost of the consumed assets (interceptors) during live firing, the lack of "targets of opportunity," and the relative rarity of actual combat conditions (all of these as opposed to, for example, SAM systems). In the first stages of deployment, it will be possible for operational crews to participate in weapon system development tests, but even these are rare, require a unit to drop its alert level including redeployment, and will become rarer as development ends and deployment progresses. All of the aforementioned considerations are even more prominent for the exercising of a full architecture (including interoperable systems).

Simulations, in particular those that include HIL, will form a part of the effort to solve this problem. The requirements from simulations to perform these tasks need to be identified and developed as early as possible.

The major role that wide scope simulations (other than the simulators that are part of the weapon systems themselves) can play in training exercises is that of emulating the other defense systems deployed in the battlefield. Thus, a given unit can train with other (emulated) units of its own architecture, without materially degrading the entire defense architecture's alert level. Alternatively, a unit can train with friendly forces simulatively deployed on the battlefield. To be able to effectively emulate peer weapon systems for actual weapon system coordination, it is necessary to guarantee a uniform "truth" picture using something like the DIS/HLA protocol. In addition, the actual communication system and protocol used by the two systems must be available to the simulations. A simulation used in such an exercise must have all of these capabilities to be valid as a training tool.

Another aspect of using simulations as training aids is the need for a highly qualified team of simulation operators able to support the activities of both the real and the simulated weapon systems. Aspects of communication across long distances (to avoid physically moving various units), remote displays, etc., have also to be taken into account.

D. Impact of High-Level Decision Making

The TBM–ATBM war, fought mostly by remote forces, is strongly affected by decisions made at higher echelons regarding force deployment, ROE, and defense

policy. The effect of these high-level decisions on battle results is more easily traceable in a conflict fought mainly by machines than by individual human combatants. This will be true of many of the battles to come in the 21st century, the TMD war being the first example.

Analysis teams working on defense architecture development should not overlook this major influence on the effectiveness of the defense architectures. Both TBM and ATBM systems are agile in terms of the ability to change battle plans quickly, sometimes even in the midst of a single attack, and the effect of such changes should be analyzed. Also, changes in battle plan and operational objectives carried out during the war have an impact on the resulting performance of the weapon systems. Neglecting to include these factors in the simulations and analyses may cause erroneous conclusions. For example, choosing between either launching large numbers of interceptors against early salvos or conserving interceptors for later stages of the conflict can affect the measured effectiveness of the architecture.

Obviously, there are several dimensions to the problem: timewise (real-time decisions vs non-real-time), attacker vs defender decisions, the perception of the enemy's doctrines and plans, the ability to make decisions based on available (rather than "God's eye view") information, etc.

Several approaches to this problem have been tried, each yielding some benefits to the analysis process. No single method that provides all of the answers has been identified. We shall now list the most beneficial approaches and discuss each one of them in brief. However, it should be pointed out that none of them offers a complete solution.

1) Rely on running a large amount of scenarios generated by a statistical tool (as already discussed). Choose the most severe cases (the "spikes" in Fig. 17) and analyze what caused the architecture failure. Determine if decision-maker actions (deployment, ROE) could improve the results. In this approach, the analysis focuses on possible defender responses, whereas the attacker is looked at as "arbitrary." On one hand, this brings to the surface unforeseen situations. On the other hand, the logic that guides the attacker's decisions is not explained and the probability of the occurrence of any such decision or action by the enemy is not addressed. Obviously, this form of analysis can only deal with non-real-time decisions.

2) Include a higher echelon element in HIL simulations (e.g., add a higher echelon unit to the structure depicted in Fig. 18). In this way, the high-level real-time decisions of the defending side may be effectively analyzed. This method is effective only if actual higher echelon personnel (high ranking officers) participate in the training and simulations. This approach has been proven to be very productive to test the defense architecture, but as in the previous method it is problematic if testing of the impact of decision making of the attacker is desired. It is also costly because of the complexity of tools required and numbers and types of manpower required.

3) Insert into an automatic simulation a model to change policy decisions based on clearly defined criteria. This method allows for some flexibility in representing the battle and its results. It also allows treatment of attacker responses, and is relatively inexpensive to implement. However, responses are limited to a few, well-defined situations, and the simulation will make predictable decisions, based on only a small fraction of the data available at each phase of the battle. Otherwise, the required databases and algorithms become too complex to be manageable within a reasonable amount of time and resources.

Table 2 Benefits/drawbacks of alternative approaches

Method	Benefits	Drawbacks
Statistical threat	Can test many options Can identify problematic situations Limited expense	No responses to events Limited "sense" to the events
Higher echelon	Can change defender policy in response to events Participation of higher-ranked individuals in study	Participation of "expensive" personnel No attacker decision making complex to perform
Automatic decision making	Can change policies of both sides	Limited, predefined choices
War game	Can change policies of both sides using actual decision makers	Participation of "expensive" personnel Complex to perform

4) Use of a two-sided war game. This approach entails the creation of an independent attacker, defender, and management teams. The management team generates a situation and provides the attacker and defender sides with relevant parameters. Each of the two sides makes decisions at regular intervals based on management-provided results and events. The decisions of each side are then fed into a battlefield simulation that generates the battle results. The management team, in accordance with the collection devices available to each side, filters the results of the simulation. This is, obviously, the most complete approach, but it is the most expensive to implement, because of the large manpower requirements, and allows the analysis of only a few, not necessarily representative, scenarios.

None of the approaches just described is definitive, and any analysis should include a combination of methods to enable a wide understanding of the TMD problem. Table 2 summarizes the benefits and drawbacks of the methods described.

E. Simulations as "Systems"—DIS/HLA

The previous sections have discussed a number of important issues in the TMD environment that need to be analyzed using advanced simulations and methods. In many cases, the problems analyzed include a number of weapon systems. It is extremely difficult to create and maintain individual simulations that contain adequate models for all weapon systems operating in a given theater (both own and allied) including HIL, and also include all of the connections between the systems. The limitations result both from the required complexity of such

a tool as well as the required level of understanding of all participating defense systems.

There are complete simulations for single weapon systems that are developed for analysis and training. The idea of a protocol that will allow simulations to work together to create a joint representation of a fuller battlefield, seems therefore to be a step in the right direction. It remains to be seen whether this alone will suffice, or whether other measures need to be taken.

For the benefit of readers unfamiliar with the terminology, we shall provide a very simplistic definition of the main terms involved. DIS is a standard communication protocol that allows real-time simulations to maintain a uniform simulated environment, by sending and receiving protocol data units (PDUs). PDUs represent entities (identity, location, status), events (e.g., interceptor launch) and decisions (e.g., hit/miss). By implementing DIS in a set of simulations it is possible to have one simulation provide the threat, another provide TMD systems and yet another provide airbreathing targets.

The HLA is the next phase after DIS in simulation connectivity. It allows also connection of non-real-time tools. In Israel, there is limited experience in using DIS, particularly for large-scale exercises, and so the remarks made here are based on an overview of DIS without considering details and variations of its use. However, a few remarks still seem in order.

1) Work on interoperability between weapon systems requires development of new coordination tools (software) and procedures in all systems that are to interoperate. If a single simulation is used as the main study tool, it is relatively simple to develop the new software and include it in the simulation for testing before implementation decisions are made. If different simulations are used for different systems, the proposed software needs to be implemented in all of these systems, and therefore it must be developed several times (for each system) resulting in increased testing cost.

2) The way most simulation developers cope with the need to communicate with other simulations is by creating HLA "federations" with particular simulations. Other simulations wishing to participate in exercises need to abide by the rules of the earlier defined federation or create a new ad-hoc federation, which involves further work on the communication between the tools. More standardization is required for this trend to become the common practice in simulation development.

Simulations and computerized tools will have a much larger role in development and deployment of weapon systems in the near future. The DIS/HLA approach to connecting simulations is a very significant step on the way to achieve the future goals, some of which were outlined here. We hope that this chapter assisted at least in pointing out some of the issues that should be handled and the ways to approach them.

Chapter 15

Theater Missile Defense Systems Readiness and Training

Karel Pick* and Joseph Zack†
WALES Ltd., Ramat-Gan, Israel

I. Introduction

ANTITHEATER ballistic missile (ATBM) systems, like many military systems, face the problem of having to operate successfully on very short notice after being dormant for long periods of time. Moreover, one of the most important uses of theater ballistic missile defense (TBMD) would be in countering a first strike/surprise attack during the opening stages of a war. Under these circumstances, a long warning time will not be available, and the system may be called upon to act literally within minutes from the first received warning. This implies that once fielded, an ATBM system must maintain a very high level of operational readiness even after years of peacetime. However, there are factors peculiar to ATBM systems that make it difficult to achieve and maintain the readiness level necessary under the aforementioned circumstances.

The most obvious factor, and the hardest to overcome, is the difficulty in training the ATBM units under "real-life" conditions. Most of the time, the ATBM radar will not detect any target. Even if it can be made to detect and track various other targets such as aircraft, this will not correctly simulate the operation of the system against TBMs, because of the large differences in target behavior, mostly trajectory and velocity. Training by exercising the system against TBM-like targets (including firing of interceptors) is limited by the high cost of such practice and by safety considerations. If the interceptors' stock is limited, the training may be reduced to the use of simulators with little actual hands-on experience with the real system. This means that the operation of the defense system as a whole, from radar to battle management and control (BMC) to launcher to interceptor, is very difficult to test. The problems facing the independent operation of the ATBM weapon system are intensified when interoperability with other systems or with allies is considered.

*Systems Analyst.

†Senior System Engineer for Advanced Weapons and Technologies. Member AIAA.