DESCRIPTION AND EVALUATION OF MULTI-THREAT TRACKING ARCHITECTURES

Muharrem Mane*, Daniel DeLaurentis*
***Purdue University, West Lafayette, IN 47907-2023**
Approved for Public Release 12-MDA-6969 (2 Aug 12)
Keywords: *sensor management, functional allocation, multi-threat tracking*

Abstract

The objective of a Ballistic Missile Defense System (BMDS) is to defend the United States, friends and allies, and deployed forces from ballistic missile attack. While successful intercept is the ultimate measure, the ability to accurately track incoming threats is a key prerequisite. This paper presents a taxonomy for the description of missile-tracking architectures and utilizes an enterprise-level Agent-Based Modeling and Simulation framework for the evaluation of missile-tracking architecture concepts. We find that the impact of communication latency plays an important role in the ability of the system to successfully track threat-missiles; furthermore, the way in which an architecture concept utilizes the communication network has a large impact on its performance.

1 Introduction

The Missile Defense Agency is responsible for developing an integrated Ballistic Missile Defense System (BMDS) linking land, sea, air, and space based assets to defend the United States, friends and allies, and deployed forces from ballistic missile attack. The Command and Control, Battle Management and Communication (C2BMC) element is the critical capability that links the various individual sensor, interceptor, and communications elements into one integrated system ensuring the highest capability against all types of ballistic missile threats. While effectively intercepting the threat missiles is the ultimate measure of a successful system, the ability to accurately track incoming threats is a key prerequisite. The best control algorithms onboard the interceptor cannot reduce the miss-distance between the interceptor and the threat below the accuracy of the state estimator [1, 2, 3] provided by the missile-tracking activity.

The improvement in Infra Red / Electro Optical (IR/EO) sensors and their potential utilization in airborne and space-based platforms to detect and track airborne threats may offer the ability to increase the effectiveness of a missile-tracking system by increasing situational awareness and the detection and tracking capabilities of the BMDS. When faced with the task of tracking multiple targets, effective sensor management and sensor tasking strategies are necessary to ensure efficient use of sensor resources and high quality information. Hero and Cochran [4] present a comprehensive historical description of research in this field. Formulation and solution of the efficient and effective missile-tracking problem has been typically addressed as a monolithic problem where the sensor tasking activity is coupled with the measurement fusion task. Work by Khosla and Guillochon [5] has considered a decoupled approach to this problem by separating the measurement fusion task – still a centralized task – from the sensor control task – where sensors collaborate but each sensor makes its own tracking decisions. Kreucher et al. [6, 7] consider the decentralized multi-platform sensor management problem where sensors share limited amount of information to make tracking decisions. The goal is reduction in communication requirements to complete the task of target tracking. In their application, both the sensor tasking and the measurement fusion are performed by each sensor. Work by Hendricks and Dana [8] have approached this problem as a bio-inspired decentralized management problem where sensors are indirectly aware of the actions of other sensors via a central data fusion that provides global situational awareness to all sensors, and these make tracking decisions based on the perceived impact that precious decisions had on the global picture.

Across all approaches, a common goal is pursued: a robust set of tracks that are not degraded when individual sensors are compromised and that minimize computational complexity and communication requirements. The various strategies investigated by the many researchers describe tracking architectures that can take advantage of particular sensor phenomenology, data fusion, sensor resources, and sensor management approaches in isolation to generate higher quality tracks.

The work described in this paper presents missile-tracking as a functional allocation problem at the enterprise level, and investigates the effectiveness of various functional allocation strategies in tracking increasing numbers of threat-missiles. Various missile-tracking architecture concepts are generated by utilizing a centralization taxonomy (described in detail in [9]) and evaluated by simulating a series of missile launch scenarios via a Matlab-based Discrete Agent Framework (DAF) implementation that enables the modeling and simulation of systems, and systems information exchange.

2 Statement of Problem

The problem is to evaluate a family of ballistic missile tracking architectures that have varying degrees of centralization for key functions. The evaluation is focused on their ability to provide quality track for an increasing number of threat-missiles. We define an architecture as the distribution and allocation of the necessary functions of the missile-tracking system to generate missile tracks. Figure 1 presents the track-generation information flow chain and its constituent functions for missile tracking. Sensors are the system's interface with the environment. They can be of different types (e.g. Infrared Red, Electro Optical, Radio, etc.) and have multiple modes of operations (e.g. active, passive, scanning, tracking, etc.).

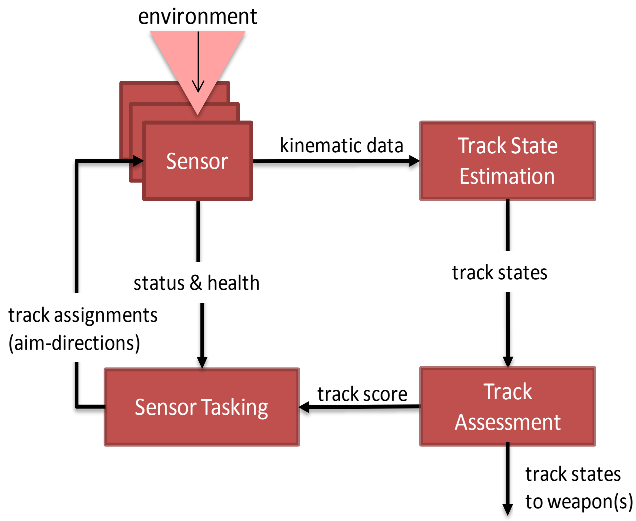


Figure 1. Track-generation function chain

A sensor's basic function is to take measurements of the environment, in this case kinematic data of observed objects. The kinematic data from sensors is filtered and used to estimate the state of the observed objects. The state estimation of an object is comprised of a description of its position and position derivatives, together called a track. The next step in the track-generation chain is the assessment of the tracks. Track assessment refers to the function of analyzing track state estimates and determining if the track quality is sufficient to enable engagement or whether further observation is required. The result of this function is a list of track scores, or track importance weights, that are sent to sensor tasking, which uses this information and knowledge about available sensors and their capabilities to re-task sensors in a manner that leads to improved track quality.

A single system can perform one, or several, of these functions. For instance, the TPY-2 radar [10] is a monolithic system that performs all of these functions and provides tracks to a weapon system. The chain gets more complex when multiple, heterogeneous sensors are available, multiple track estimators are employed, track assessment is performed by using differing sets of data, and sensor tasking is alternately performed by a centralized authority, in a decentralized fashion, or even independently.

2.1 Taxonomy of Architecture Design Space

The allocation of the functions in the track-generation chain constitutes a missile-tracking architecture. In this research, we identify three dimensions by which to describe missile-tracking architectures (Figure 2): 1) track formation, 2) track assessment, and 3) sensor tasking. We utilize a three-level centralization hierarchy that facilities expression of the architecture: sensor, ground controller (GC), and command and control (C2). The sensor nodes represent the entities that interact with the environment and the lowest level of centralization, while the C2 represents the highest level of centralization. For instance, each sensor node can perform its own sensing, sensor tasking, track formation, and track assessment function independently of the other sensors. The Ground Controller (GC) nodes can have computational and control functions, interacting with sensors to which they are assigned, and represent a higher centralization level than the sensor nodes. A GC node can be responsible for the tasking of multiple sensors or the formation of tracks with data from multiple sensors. The same GC functions can be performed at the Command and Control (C2), but in context of richer information and a higher centralization level.

The location of the track formation indicates where this function is performed in the hierarchy, e.g. its level of centralization. For instance, if track formation occurs at the sensor, a decentralized architecture with respect to track formation is obtained; if, on the other hand, the GC node is responsible for track formation (estimating track states by fusing kinematic data from sensor nodes), a more centralized architecture is obtained (track formation at the C2 node is the highest centralization level). The location of the track assessment function indicates where track information is assessed and track scores generated. Similarly, the location of the sensor tasking function indicates where in the hierarchy sensors are managed and allocated and the level of centralization with respect to this function. Note that track formation at the sensor node implies that track fusion must occur if track assessment is performed at the GC or C2 node, while track formation at the GC or C2 implies that measurement fusion occurs at these nodes. Track fusion is the activity of correlating and combining tracks from different sensors into a single system-level track.

Multiple architecture concepts with different degree of centralizations can be described by this taxonomy. Figure 2 calls out two examples.

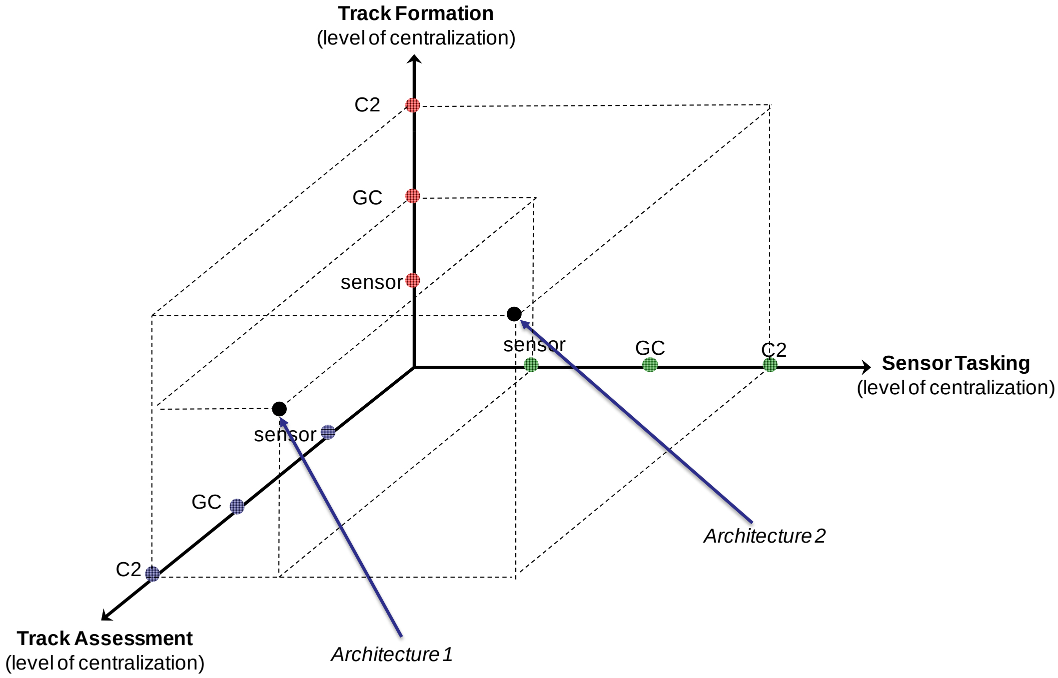


Figure 2. Missile-tracking architecture dimensions

Architecture 1 is an example of an architecture with a centralized track assessment (at the C2 node), decentralized track formation (at the GC node), and decentralized sensor tasking (at the sensor). This is also an example of an architecture where tracks are fused at the C2 node prior to the track assessment function. On the other hand, architecture 2 is highly centralized in all three dimensions – tracks are formed and assessed at the C2 node and sensors are tasked by the C2 node.

Note that, while a maximum of three centralization levels are presented here, not all of them must be used in a given architecture or, alternatively, additional levels can be introduced. For instance, it is possible to have a highly decentralized architecture where the sensors are autonomous and collaborate with each other via the C2 node, eliminating the GC nodes. On the other hand, if the C2 node performs the track assessment and sensor tasking function, the result is a highly centralized architecture. If the C2 node performs the track assessment function but each sensor is responsible for tasking itself, we have the case of a fairly decentralized architecture with a centralized knowledge base (i.e., sensors are indirectly aware of others via the track scoring results which are determined from the centrally fused information).

2.2 Implications of Function Centralization

The three levels of hierarchy and three dimensions produce 27 possible architectural combinations that describe the design space. These vary from highly centralized architectures where track formation, track assessment, and sensor tasking is performed at a C2 node to highly decentralized architectures where these functions are performed at each sensor, and anywhere in between. There are positive and negative implications for each of these architecture concepts that are at the center of the numerous trade studies of interest to BMDS designers. For instance, an architecture that has a high degree of centralization in the sensor tasking dimension is able to take advantage of the possible complementary capabilities of numerous different sensors (or sensor operational modes) to generate higher quality data and utilize assets efficiently. However, management of numerous sensors (or sensor modes) adds computational complexity and thus computational latency to generate a required solution. Additionally, the reliance on a single entity to make tasking decisions implies high vulnerability to technical failures or malicious attacks. An architecture with a low degree of centralization in the sensor tasking dimension – e.g. sensors decide on their own which targets to track – can make for a less vulnerable missile-tracking system, albeit at a cost in sensor utilization efficiency.

The level of centralization in the track assessment dimensions can be interpreted as the level of situational awareness in the system. The result of the track assessment function is a track priority list that is used in the sensor tasking function to guide allocation of sensors. Hence, coordination in the prioritization of tracks results in an indirect coordination of sensors that can result in efficient sensor utilization. Architectures with a centralized track assessment function achieve this global situational awareness by prioritizing tracks based on the information available from all sensors. Hendricks and Dana [8] present one such example where a centralized data fusion node maintains a global situational awareness that is accessed by all sensors. Note, however, that this does not guarantee efficient use of resources unless the track assessment function is also centralized (combined with the data fusion in [8]) or sensors collaborate as proposed by Kreucher et al. [6], where sensors inform each other of their target tracking decisions.

Track formation – the activity estimating the state of an object – can also result in varying levels of track quality depending on the level of centralization. A centralized track formation function can benefit from the fusion of measurement from multiple sensors and sensor types by reducing measurement covariance prior to filtering, resulting in higher quality tracks. This, however, can result in high data loads on the communication network, given the high frequency of measurement transmission when compared to a relatively low frequency of track transmission that would be required if track formation were decentralized – e.g. each sensor contributes tracks to the GC or C2 node.

The remainder of this paper will make use of the taxonomy presented here to generate a set of architecture concepts for a missile-tracking system and introduce a modeling and simulation approach for assessing the performance of said architectures.

3 Methods and Approach

A modeling and simulation infrastructure has been developed to model missile-tracking architecture concepts along the functional dimensions at various levels of centralization and to quantify the difference in architecture performance in terms of its ability to generate high-quality tracks. Additional performance metrics, such as resilience, robustness, timeliness of solution, etc., are important and can be quantified by the model, but they are not a focus of the study presented here.

The approach uses an Agent-Based Modeling (ABM) [11, 12, 13, 14] formulation where the systems of the missile-tracking system are represented by agents. In particular, the Discrete Agent Framework (DAF) has been created (see [9] for a detailed description) as a means to simulate and evaluate missile-tracking architecture concepts. In DAF, individual systems are modeled as agents – instantiations of a system or entity – whose behavioral model can be flexibly specified (e.g., physics-based or heuristic, as appropriate for the system of interest). System operations are the embodiment of the functions that each agent is designed to accomplish. Information exchange is the underlying network that links the agents to each other and enables the modeling of operational interdependencies and the exchange of data/information or any other type of interaction between systems (agents). This results in the construction, and eventual simulation, of networks of interacting and interdependent systems.

The next few sub-sections describe the functions modeled for a missile-tracking system developed and utilized in this study to assess the performance of different architecture concepts that center on different levels of centralization of functional allocation. Note that considerable research centers on the development of theoretical and computational methods that effectively solve each missile-tracking sub-problem. Because the goal of this work is not to evaluate the performance of different methods and algorithms that address each function, we present and utilize simplified models of each function that capture the behavior of the function and enable comparison of architecture configurations.

3.1 Sensor Function

A sensor's basic function is to sample the three-dimensional environment and generate kinematic data of observed objects. A sensor will receive aim-directions from the sensor tasking agent that define where its field of view (FoV) should point within its field of regard (FoR). For the infrared sensor modeled here, an object is detected if it is in the FoV of the sensor and if the irradiance of the object is above the detection threshold of the sensor. If an object is detected, a measurement of its position is generated. The measurement-generation capability of the sensor is modeled as a function of the sensor resolution and the slant range to the observed object. The measurement noise is modeled as Gaussian with a standard deviation equal to the sensor resolution (in radians for bearing measurements and meters in range measurements).

We assume that all sensors are passive sensors – they only observe the environment – and are able to generate bearings-only measurements; that is, they report no range information. While there is a considerable body of work in multiple-target tracking with bearings-only measurements [15, 16, 17, 18] that considers means of achieving reliable range estimates – e.g. by moving the sensor platform or by coordinating multiple sensors and triangulation – we capture this behavior by parameterizing the range resolution of the passive sensors.

3.2 Track Formation Function

The objective of the track formation function is to use measurement data from sensors to estimate the states of the observed object(s), e.g. position and its derivatives. We use a nonlinear Kalman filter to estimate an object's position (in three dimensions) and its derivatives by using kinematic data generated by sensors [19]. Although many other filters are possible (unscented Kalman filters, particle filters, and central-difference filters are discussed in the tracking literature [20, 21, 22, 23]), variations of the Kalman filter are still widely used in modern systems and ballistic missile tracking [24, 25, 26, 27] due to the low computational cost and relative ease of manipulation to address various special tracking cases. To capture the increase in tracking performance due to measurements from multiple sensors, we use a covariance intersection technique [19,28] to fuse measurements prior to filtering. This enables the quantification of the change in performance of missile-tracking architectures when varying the centralization level along the track formation dimension. For instance, a centralized architecture in the track formation dimension is expected to generate higher quality tracks by using the covariance intersection than a decentralized architecture where each sensor generates its own tracks and is unable to take advantage of the covariance intersection to increase track quality. We also use the covariance intersection approach to fuse tracks from different sensors into a single system-level track. Track fusion is performed when tracks are formed at the sensor node and the track assessment function occurs at the GC or C2 node (e.g. non-centralized track formation).

3.3 Track Assessment Function

Different prioritization criteria exist for different sensor tasking objectives and for different times during missile-tracking. For instance, after initial detection, the objective would be to discriminate the object and determine if it poses a threat. In this context, tracks would be scored based on a discrimination confidence level; tracks that have not been discriminated get higher priority. Similarly, if the nature of the object is known, the prioritization criteria would be the confidence in the classification of the object or threat, again, with tracks that have not been classified having higher priority. If, on the other hand, detection of an object occurs much later after launch, the tracks that are estimated to be closer to a defended area might take higher priority.

In this paper, we ignore the discrimination and classification functions that can be part of the track prioritization task and assume that the criterion of interest is the quality of the track (e.g. track error covariance). Covariance-control-based and information-based methods are other approaches used to prioritize targets and assign sensors [29]. Covariance control methods attempt to satisfy some covariance requirement or goal, typically dictated by weapon capabilities. Information methods, on the other hand, try to maximize the information gain of specific sensor-target pair(s). We use the 2-norm of the error covariance matrix to score tracks and provide an importance or priority weight to the sensor tasking function. Because the sensor tasking problem will be formulated as an error covariance minimization problem, tracks with higher error covariance have higher priority over tracks with lower error covariance.

3.4 Sensor Tasking Function

Sensor tasking as a feedback control system seeks to effectively and efficiently allocate sensor resources to achieve or maintain a certain level of situation awareness. This problem is encountered in many applications where inputs from sensors are used to direct future actions of sensors and sensor platforms to achieve some goal, like robot navigation and autonomous vehicle guidance [30, 31, 32], target tracking [33], and many more. For target tracking, the problem of sensor tasking is to determine how to select sensors, sensor modes, and sensor search patterns to maximize the integrated effectiveness of sensors which may be located at different platforms, against a set of mission requirements [34].

With ever-increasing sensor capabilities and diversity in sensor types, challenges in the formulation and solution of these problems have motivated many researchers to investigate approaches to efficiently generate high quality and timely solutions. Generally, research efforts fall under two categories: a) solution techniques to the resource allocation problem and, b) development of metrics to use as objective functions of said problems. Researchers have approached the sensor tasking problem with a Markov Decision Process (MDP) strategy [35], linear programming formulations and solutions [36], artificial neural networks [37, 38, 39], and many more, as summarized in [33, 34]. Metric development approaches – typically used as objective function in the problem formulations above – use information theoretic criteria like entropy or discrimination gain to establish prioritization for sensor tasking [40, 41, 7].

For the purpose of this research, we formulate the sensor tasking problem as a linear programming problem that aims to minimize the track error covariance by deciding the aim-directions of the available sensors, e.g. which tracks a given sensor should cover. Once an aim-direction command is communicated to the sensor, the sensor will report measurements of what it sees in the assigned field of view (see [9] for the mathematical formulation of the sensor tasking problem).

The level of centralization along the sensor tasking architecture dimension can be modeled by varying the number of sensors that a given sensor tasking node controls. For instance, in a highly decentralized architecture, each sensor can perform its own sensor tasking function, ignoring the actions of other sensors. In a highly centralized architecture, on the other hand, all sensors can be tasked by s single sensor tasking node that ensures sensor coordination and efficient use of sensor resources. When coupled with the allocation of the track assessment and track formation functions, this enables the modeling and comparison in performance of different architecture configurations.

3.5 Simplifying Assumptions

In order to caution the reader about use of the conclusions drawn in this paper, the following simplifying assumptions should be noted. First, the threat trajectories are represented only by the kinematic state of the primary object thread. Thus, secondary objects and deployment events are ignored in the simulation and analysis. Second, sensors are assumed to have perfect resolution. In other words, two closely-spaced objects in sensor space are reported as two objects without respect to the distance between the objects and the resolution of the sensor. This is of particular importance for IR sensors since these type of sensors have essentially no resolution in range and the scenario includes closely-spaced launches. Finally, several association functions are assumed perfect, including measurement-to-track association in sensors, measurement-to-measurement association across the sensors, and track-to-track association across the sensors.

4 Experimentation and Analysis

In this paper, we consider six functional allocation strategies (architectures A1–A6) for a missile-tracking system comprised of sensor and C2 nodes (Table 1).

Table 1. Centralization levels of example missile-tracking architectures

Architecture/Task	Track Formation	Track Assessment	Sensor Tasking
A-1	C2	C2	C2
A-2	C2	C2	Sensor
A-3	C2	Sensor	Sensor
A-4	Sensor	Sensor	Sensor
A-5	Sensor	C2	Sensor
A-6	Sensor	C2	C2

Architecture A-1 represents a highly centralized architecture where all functions are performed at the C2 node, while architecture A-4 is a highly decentralized one where all functions are performed by each sensor independently of each other. Architecture A-2 has decentralized sensor tasking but centralized track formation and track assessment. The goal is to assess the performance of a functional allocation strategy where system-level tracks are generated via measurement fusion and sensors are indirectly aware of each other's actions via the centralized track assessment, but still make independent decisions with respect to which missiles to track. Architecture A-3 is decentralized in the track assessment and sensor tasking dimension but centralized in the track formation dimension. The idea here is to capture the difference in architecture performance when

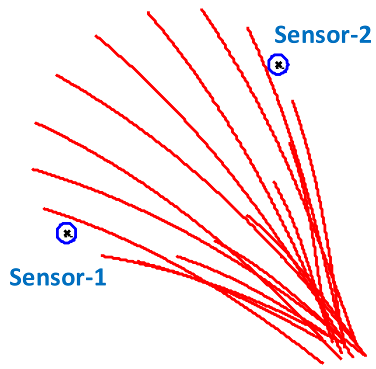


Figure 3. Scenario and sensor placement example

measurements are fused to create system-level tracks instead of when tracks are fused (as in A-4). Architecture A-5 represents an architecture where track assessment is centralized, but all other functions are decentralized. When compared to architecture A-4, this architecture isolates the impact of a centralized situational awareness and can point to the potential positive or negative effects of the indirect coordination of sensors, via the centralized track assessment function. Finally, by having sensors perform the track formation function, architecture A-6 represents an architecture where tracks from sensors are fused at a central node. When compared to A-1, this provides the means to quantify the difference in architecture performance between measurement and track fusion.

4.1 Scenario Setup

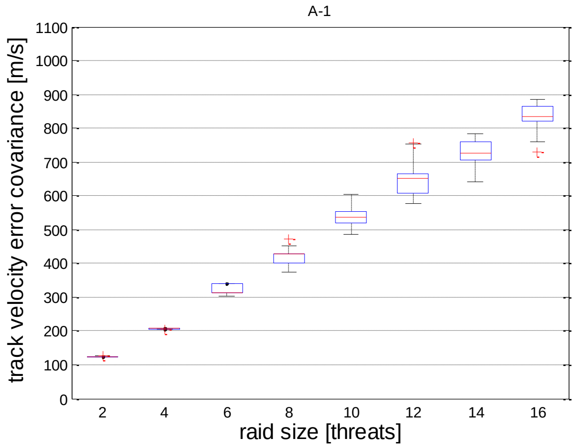
We simulate a set of scenarios for raid sizes varying from one to 16 simultaneously launched missiles and their detection and tracking by one IR satellite (in geosynchronous orbit) and two airborne and pointable IR sensors via DAF. Figure 3 presents a two dimensional view of the sensor locations (IR satellite location not shown because it can see all missiles for the entire simulation time) and assumed missile trajectories (flight is from left to right). Note that the scenario is three-dimensional but is presented as two-dimensional here for ease of visualization. Furthermore, we assume the sensors have a large-enough field of regard to enable them to track all threats for the duration of the simulation. While this may not always be the case in an actual engagement, the assumption simplifies analyses by removing the uncertainty of sensor coverage.

Track error covariance is the performance metrics used to compare the six architecture concepts. The simulation model also considers the impact of the amount and type of information exchange between agents. A message that moves through the communication network and that contains a measurement is smaller than a message that contains a track. A measurement message contains a set of coordinates that indicates the perceived position of the observed object, the time it was taken, sensor location at time of measurement, measurement id, and perceived irradiance of the object; while a track is a set of matrices that describe the position, velocity, acceleration, jerk, jolt and their covariance in the ECEF (Earth Central Earth Fixed) coordinate system. In this analysis we assume that a message containing a measurement is 0.46 kbits and a message containing a track is 15.93 kbits in size. A message from an agent that performs the track assessment function to an agent that performs the sensor tasking function contains all known tracks as well as a vector indicating the priority value of each track; hence it is slightly larger than a message containing a track (15.96 kbits), which is the output of an agent that performs the track formation function. Finally, a message from the sensor tasking to the sensor contains the aim-directions of the sensors controlled by the agent that performs the sensor tasking function. These consist of azimuth and elevation angles and are 0.1 kbits in size. The communication latency and traffic are, therefore, a function of the type of messages between agents, which depend on the type of function performed by each agent. For instance, if a sensor agent performs the sensor tasking function (e.g. it tasks itself), then no messages containing aim-direction move through the communication network.

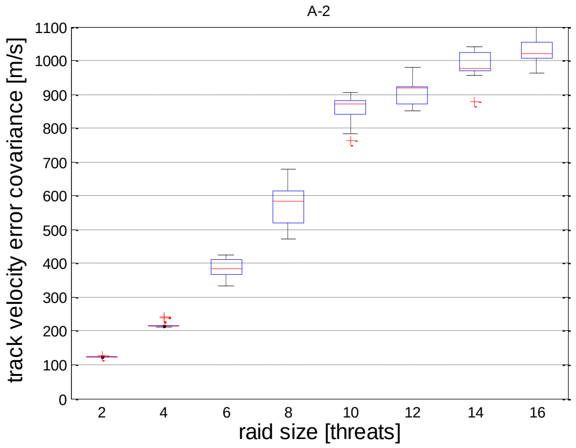
4.2 Simulation Results

A simultaneous threat launch and tracking scenario is simulated for 100 seconds. To capture the impact of variation in the quality of measurements and tracks, each scenario is repeated 100 times and the statistics of the results are analyzed. Figure 4 presents the average track error covariance over the 100 second simulation interval of each architecture concept. The box-plot represents the 25th and 75th percentiles of the error covariance. Because the confidence intervals do not overlap, we can conclude with 95% confidence that the medians of each scenario do differ.

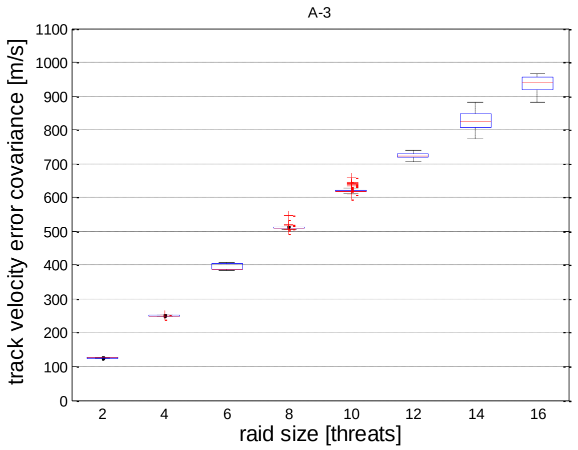
These results indicate that the performance of all architectures decreases (e.g., results in



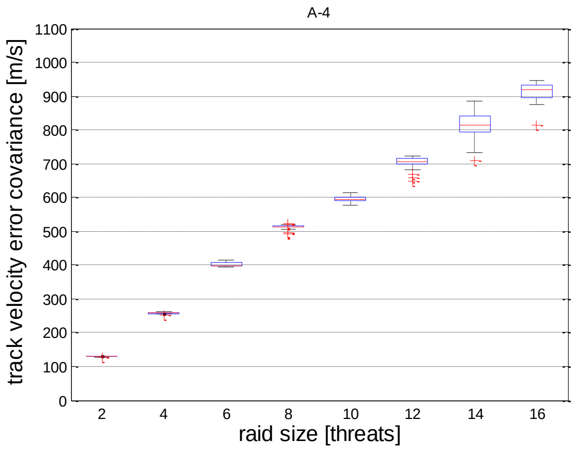
a) A-1: centralized in all three dimensions



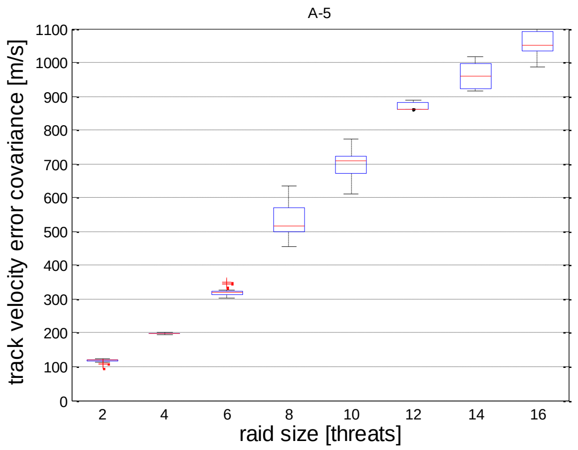
b) A-2: decentralized in sensor-tasking



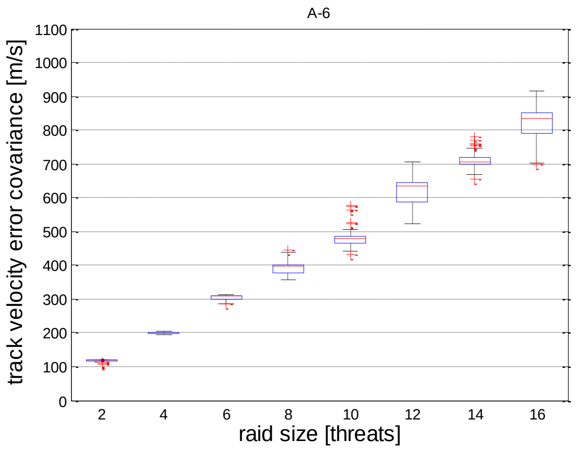
c) A-3: centralized track-formation



d) A-4: decentralized in all three dimensions



e) A-5: centralized in track-assessment



f) A-6: decentralized in track-formation

Figure 4. Track velocity error covariance error as a function of raid size

higher error covariance) as the number of threat missiles that must be tracked increases. Recall that the goal of missile-tracking is to achieve and maintain the smallest possible error covariance for all tracks. Therefore, as the number of missile increases, less time is spent covering each track.

4.3 Track Performance Comparison

A more direct comparison of the architecture performance is presented in Figure 5, where the mean error covariance of the architectures is plotted in the same figure.

When compared in this manner, there is an obvious grouping of architectures as the raid size increases. Architecture concepts A-1 and A-6 perform very similarly and better than the other concepts. In these architectures the track assessment and sensor tasking is centralized at the C2 node, which points to the advantage of

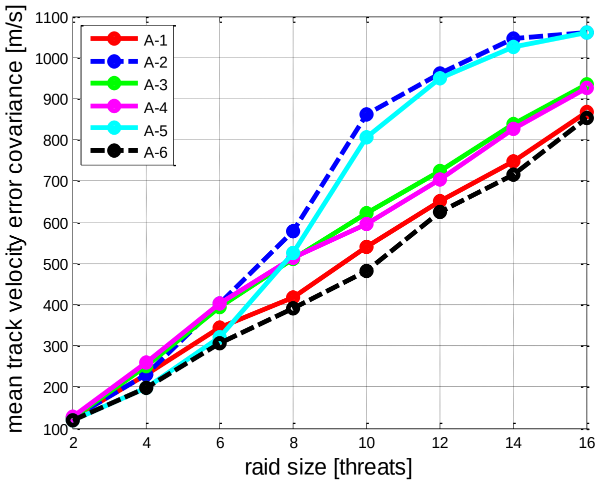


Figure 5. Architecture performance comparison

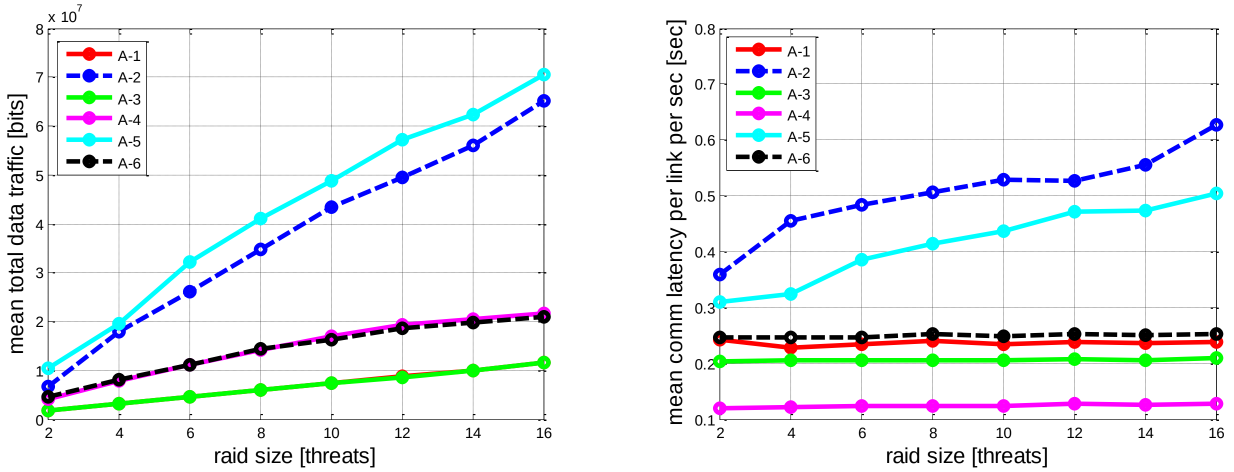
sensor coordination that is enabled by the C2 node. Architectures A-3 and A-4 have decentralized sensor tasking and track assessment, which means that sensors operate independently of each other. Because of the lack of coordination between sensors one would expect worse performance. This does happen for small raid sizes (smaller than six threat missiles), but as the number of missiles to track increases, their performance improves. Architecture concepts A-2 and A-5 become the worst performing concepts for large raids. The difference between these concepts and the others is the spatial decoupling of the track assessment and the sensor tasking functions. In architectures A-1 and A-6 these two functions are performed at the C2 node, while in architectures A-3 and A-4 they are performed at the sensor; in architectures A-2 and A-5 track assessment is performed at the C2 node and sensor tasking at the sensors.

These architecture concepts use the communication network to exchange information, which means that as the raid size increases, the offered load, and consequently the communication latency, increases, causing delays in the information exchange and causing the system to inefficiently use the sensor resources to obtain and maintain low covariance on all available tracks.

4.4 Communication Network Performance

Figure 6 shows how the amount of data in the network (Figure 6a) and the resulting communication latency (Figure 6b) for all architecture concepts and for all raid sizes. We assume that all communications occur via a satellite link with a 100 kbps bandwidth. Architecture concepts A-1 and A-3 generate the smallest amount of data (for the entire 100 second simulation) because they only send measurements via the communication network. Recall that in these architectures the track formation function is performed at the C2 node. In architectures A-4 and A-6, on the other hand, track formation is performed at the sensor, which means that sensors send tracks to the C2 node. In both of these sets of architectures, the track assessment and sensor tasking functions are co-located (e.g. in A-3 and A-4 the sensors perform both of these functions, while in A-1 and A-6 they are performed at the C2 node).

In architecture concept A-2 sensors send measurements to the C2 node and receive track assessment from the C2 node. Similarly, in architecture A-5, sensors send tracks to the C2 node and receive track assessment from the C2 node. As previously mentioned, it is this exchange of information which uses the communication network that results in higher data traffic in these concepts. This in turn



a) Communication network data traffic
b) Communication network latency

Figure 6. Performance of communication network

impacts the communication latency in these architectures and the change in track quality performance. Figure 6b presents the mean communication latency of each communication link per second – there are three communication links in this example: one link from the IR satellite and one link from each airborne sensor to the C2 node. Communication latency is a function of the offered load in the network and the number of messages in the network. For instance, latency is smaller when only measurement messages are sent through the network than when measurement messages and sensor status messages (e.g. location) are sent. This is evident in Figure 6b, where architecture A-4 has the lowest latency because it only sends tracks to the C2 node. Architecture concept A-3, on the other hand, sends measurements as well as sensor status messages. Note that in these architectures (as well as architectures A-1 and A-6) the offered load is well below one and congestion and associated latency is insignificant; the latency here is only a function of the number of messages that are sent through the network. In fact, in all architectures, except A-2 and A-5, offered load is well below the congestion point, as is evident by the constant latency as the raid size increases. Recall that in these four architectures the sensor tasking and track assessment functions are co-located and the sensor field of view is such that it can only cover one track at a time. As a result, regardless of the number of tracks that must be covered, sensors can only provide a fixed number of measurements and/or tracks in a given time interval.

Architecture A-2 has more message types that go through the network (track formation and track assessment are performed at the C2 node while sensor tasking at the sensor) than architecture A-5, hence the higher communication latency. Furthermore, due to the physical decoupling of the track assessment and sensor tasking function, the offered load in these architectures increases with raid size, with expected impacts on latency and track quality.

5 Summary and Conclusions

Development of specialized and highly capable sensor, computing, and network resources creates a multitude of functional allocation possibilities that can increase the effectiveness of the missile-tracking task within the BMDS.

We presented a taxonomy for describing a missile-tracking architecture as function of centralization level of the constituent functions of the system. We utilize this taxonomy to generate six architecture concepts that explore the design space of such a system via two centralization levels for the track-generation, track-assessment, and sensor-tasking functions.

Comparison of the six architectures reveals that centralization of the track assessment and sensor tasking function results in system architectures that are able to generate low covariance tracks as the number of missiles increases. Similar low covariance (although not as low) is also obtained when these two functions are performed by each sensor. The important architecture feature that negatively affects performance for large raid sizes is the physical de-coupling of the track assessment and sensor tasking functions (represented by architecture concepts A-2 and A-5). In these concepts the system relies more heavily on the communication network to exchange track assessment and sensor tasking information, which in turn results in higher communication latency and ultimately, lower track quality.

The work presented here does not consider other system performance metrics like robustness and resilience. These are important performance differentiators that have wide-ranging implications on the ability of the system to complete the mission. Furthermore, the features of architectures A-2 and A-5 (e.g. physical decoupling of track assessment and sensor tasking) that negatively impact their performance may present the ability of the system to resist failure more effectively. The ABM framework (DAF) developed for this work is able to evaluate such metrics and ongoing work considers modeling of these architecture performance characteristics and exploration of the trade space between quality, resilience, and robustness of different missile-tracking architectures.

References

1 Blackman, S. and R. Popoli, R., *Design and Analysis of Modern Tracking Systems*, Artech House, Norwood, MA, 1999.

2 Kim, J., Menon, P.K., and Ohlmeyer, E., “Motion Models for use with the Maneuvering Ballistic Missile Tracking Estimators,” AIAA Guidance, Navigation and Control Conference, Toronto, Aug 2010.

3 Lukacs, J. and Yakimenko, O., “Trajectory-Shaping Guidance for Interception of Ballistic Missiles in the Boost Phase,” *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, Oct 2008.

4 Hero, A.O., and Cochan, D., “Sensor Management: Past, Present, and Future,” *IEEE Sensors Journal*, Vol. 11, No. 12, Dec 2011, pp. 2064-3075.

5 Khosla, D. and Guillochon J., “Distributed Sensor Resource Management and Planning,” *Proceedings of the SPIE Signal Processing, Sensor Fusion, and Target Recognition*, Vol. 6567, May 2007.

6 Kreucher, C., Kastella, K., Hero, A.O. III, “Sensor Management Using an Active Sensing Approach,” *Signal Processing* Vol. 85, No. 3, Mar 2005, pp. 607-624.

7 Kreucher, C., Kastella, K., Hero, A.O. III, “Multi-Platform Information-Based Sensor Management,” *Proceedings of the SPIE Defense Transformation and Network-Centric Systems Symposium*, Vol. 5820, May 2005, pg 141-151.

8 Hendricks, L.C. and Dana, S., “Bioinspired Resource Management for Multiple-Sensor Target Tracking Systems,” 2011 IEEE Sensors, Oct 2011, pp. 1189-1192.

9 Mane, M. and DeLaurentis D., “Sensor Platform Management Strategies in a Multi-Threat Environment,” AIAA InfoTech@Aerospace 2012, Garden Grove, CA, 19-21- June, 2012

10 Jane’s Radar and Electronic Warfare Systems [online source], URL: http://articles.janes.com/articles/Janes-Radar-and-Electronic-Warfare-Systems/AN-TPY-2-United-States.html [accessed 26 March 2012]

11 Grimm, V., Revilla, E., Berger, U., Jeltsch, F., Mooij, W.M., Railsback, S.F., Thulke, H.H. Weiner J., Wiegand, T., and DeAngelis, D.L., “Pattern-Oriented Modeling of Agent-Based Complex Systems: Lessons from Ecology,” *Science*, Vol. 310, No. 5750, Nov 2005, pp. 987-991.

12 Borshchev, A. and Filippov, A., “From system Dynamics and Discrete Event to Practical Agent Based Modeling: Reasons, Techniques, Tools,” The 22nd International Conference of the System Dynamics Society, Oxford, England, 25-29 Jul, 2004.

13 Bonabeau, E., “Agent-Based Modeling: Methods and Techniques for Simulating Human Systems,” Proceedings of the National Academy of Sciences, 2002, Vol. 99, pp. 7280-7287.

14 Epstein, J.M., “Agent-Based Computational Models and Generative Social Science,” Complexity, 1999, Vol. 4, No. 5, pp.41-60.

15 Nardone, S., Lindgren, A., and Gong, K., “Fundamental Properties and Performance of Conventional Bearings-Only Target Motion Analysis,” *IEEE Transactions on Automatic Control*, Vol. 29, No. 9, Sep 1984, pp. 775-787.

16 Sworder, D.D., Boyd, J.E., Hutchins, R.G., and Elliott, R.J., “Bearing-Only Tracking from a Stationary Platform,” Conference Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers, Vol. 2, Nov 2001, pp. 1428-1432.

17Hanselmann, T. and Morelande, M., “Multiple Target Tracking with Asynchronous Bearings-Only-Measurements,” 10th International Conference on Information Fusion, Jul 2007, pp. 1-8.

18 La Scala, B. and Morelande, M., “An Analysis of the Single Sensor Bearings-Only Tracking Problem”, 11th International Conference on Information Fusion, Jul 2008, pp. 1-6.

19Julier, S.J. and Uhlmann, J.K., "A Non-divergent Estimation Algorithm in the Presence of Unknown Correlations," Proceedings of the American Control Conference. Albuquerque, NM, June 1997, pp. 2369-2373.

20 Van Zandt, J.R., "Boost Phase Tracking with an Unscented Filter," MITRE Corp. 2002.

21 Hough,M.E., "Nonlinear Recursive Filter for Boost Trajectories," *Journal of Guidance, Control & Dynamics* Vol. 24, No. 5, Sept–Oct 2001, pp. 991-997.

22Julier, S.J., "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators," *IEEE Transactions on Automatic Control*, Vol. 45, No. 3, Mar. 2000, pp. 477–482.

23 Lin,X., Kirubarajan,T., Bar-Shalom,Y., and Maskell, S,. "Comparison of EKF, Pseudo-Measurement Filter, and Particle Filter for a Bearing-Only Target Tracking Problem," *Proceedings of SPIE, Signal and Data Processing of Small Targets*, Vol. 4728, 2002.

24 Kalman, R.E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME, Journal of Basic Engineering*, Vol. 82, Mar 1960, pp. 34–45.

25 Aidala, V., "Kalman Filter Behavior in Bearings-Only Tracking Applications," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-15, No. 1, Jan 1979, pp. 29-39.

26 Siouris, G.M., Chen, G., and Wang, J., "Tracking an Incoming Ballistic Missile Using an Extended Interval Kalman Filter," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 33, No.1, Jan 1997, pp.232-240.

27 Saulson, B.G. and Chang, K.C., "Nonlinear Estimation Comparison for Ballistic Missile Tracking," *Optical Engineering*, Vol. 43, No. 6, May 2004, pp. 1424-1446.

28 Julier, S.J. and Uhlmann, J.K., "Generalized Decentralized Data Fusion with Covariance Intersection," *Handbook of Multisensor Data Fusion*, CRC Press, New York, 2009.

29 Li, X., Chen, G., Blasch, E., Patrick, J., Yang, C., and Kadar, I., "Multi-Sensor Management for Data Fusion in Target Tracking," *Proceedings of SPIE Signal Processing, Sensor Fusion, and Target Recognition*, Vol. 7336, May 2009.

30 Lumelsky, V.J., Mukhopadhyay, S., and Sun, K., "Dynamic Path Planning in Sensor-Based Terrain Acquisition," *IEEE Transactions on Robotics and Automation*, Vol. 6, No 4, Aug 1990, pp. 462-472.

31 Stopp, A. and Riethmuller T., "Fast Reactive Path Planning by 2D and 3D Multi-Layer Spatial Grids for Mobile Robot Navigation," Proceedings of the 1995 IEEE International Symposium on Intelligent Control, 27-25, Aug 1995, pp. 545-550.

32 Brunet, C.A., Gonzales-Rubio, R., Tetreault, M., "A Multi-Agent Architecture for a Driver Model for Autonomous Road Vehicles," Canadian Conference on Electrical and Computer Engineering, Montreal, Canada, Sep 1995, pp. 772-775.

33 Musick, S., and Malhotra, R., "Chasing the Elusive Sensor Manager," Proceedings of the IEEE 1994 National Aerospace and Electronics Conference, Vol. 1, May 23-27, 1994.

34 McIntyre, G.A. and Hintz, K.J., "A Comprehensive Approach to Sensor Management, Part I: A Survey of Modern Sensor Management Systems" IEEE *Transactions on Systems, Man, and Cybernetics*, Apr 1999.

35 Malhotra, R., "Temporal Considerations in Sensor Management," Proceedings of the IEEE National Aerospace and Electronics Conference, Vol. 1, May 1995, pp. 86-93.

36 Nash, M.J., "Optimal Allocation of Tracking Resources," Proceedings of the IEEE Conference on Decision and Control, 1977, Vol. 1, pp. 1177-1180.

37Molina Lopez, J.M., Jimenez Rodriguez, F.J., Casar Corredera, J.R., "Fuzzy Reasoning for Multisensor Management," IEEE International Conference on Systems, Man, and Cybernetics, 1995, Vol. 2, pp. 1398-1404.

38 Zhongliang, J., Hong, X., Xueqin, Z., "Information Fusion and Tracking of Maneuvering Targets with Artificial Neural Networks," Proceedings of IEEE International Conference on Neural Networks, 1994, Vol. 5, pp. 3403-3408.

39 Brownell, T.A., "Neural Networks for Sensor Management and Diagnostics," Proceedings of the IEEE National Aerospace and Electronics Conference, 1992, Vol. 3, pp. 923-929.

40 Hintz, K.J., "A Measure of the Information Gain Attributable to Cueing," *IEEE Transactions on Systems, Man, and Cybernetics*, 1991, Vol. 21, No, 2, pp. 434-442.

41 Kastella, K., "Discrimination Gain to Optimize Detection and Classification," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, Jan 1997, Vol. 27 No. 1, pp. 112-116.

Copyright Statement