

Dynamic Tactical Targeting

Phil Hanselman^{a*}, Craig Lawrence^b, Evan Fortunato^c, Bob Tenney^d, Erik Blasch^a

^a Air Force Research Laboratory, 2241 Avionics Circle, Wright Patterson AFB, OH. 45433

^b ALPHATECH, Inc., 3811 North Fairfax Drive, Suite 500, Arlington, Virginia 22203

^c ALPHATECH, Inc. 6 New England Executive Park, Burlington, MA 01803

^d DARPA, 3701 N. Fairfax Drive, Arlington, Virginia 22203

ABSTRACT

Today's Warfighter requires new capabilities that reduce the kill chain timeline. The capability to maintain track on mobile Time Sensitive Targets (TSTs) throughout the entire targeting cycle is a step towards that goal. This capability provides the strike assets with high confident, actionable, targeting information, which reduces the time it takes to reacquire the target prior to prosecution. The DARPA Dynamic Tactical Targeting (DTT) program is developing new sensor resource management and data fusion technologies for continuous coordination of tactical sensor resources to detect and identify mobile ground targets and maintain track on these known high-value targets. An essential concept of the DTT approach is the need for the fusion system and the resource manager to operate as part of a closed loop process that produces optimum collection plans against the designated high value TSTs. In this paper, we describe this closed loop approach used within the DTT system. The paper also describes other aspects of the DTT program, including overall program status, the DTT distributed architecture, details of the fusion and dynamic sensor management components, and concludes with current evaluation results.

Keywords: Dynamic, targeting, closed-loop, fusion, sensors, tracking, control

1. INTRODUCTION

DARPA's Dynamic Tactical Targeting (DTT) program is developing a capability for continuous coordination of tactically responsive organic sensors to detect and identify mobile ground targets and maintain track on known high-value targets. Central to the DTT approach is the closed-loop interaction between the Fusion and Sensor Management (SM) systems. As part of this interaction, Fusion generates an information needs (IN) requests to the SM system to provide information which is critical to maintain track on a target of interest. This capability allows a Battlefield Commander to observe fusion requests and control sensor assets with proactive tasking, collection, and the extraction of critical data. The DTT program was motivated by limitations of the current targeting process to find, identify, and prosecute time critical targets (TCTs). Current stovepiped collection management, exploitation, dissemination, and targeting systems limit responsiveness which is particularly critical for TCT prosecution. Missed opportunities are created by latencies introduced during each step of the process. Thus the challenge for DTT is to break the stovepipes and create an integrated, closed-loop, system providing "...warfighters with a manageable process and effective tools to enable sensor resource management and sensor data exploitation for effective prosecution of time critical targets" [1]

The DTT program commenced in March 2002, with a series of design meetings and Integrated Product Team activities. These efforts resulted in a development plan, which includes three incremental builds of the system. Each build entails development, integration and testing phases. The third build of the system will focus on developing a real-time system, integration with a transition partner and field exercises and demonstrations. Currently, the build one version of the DTT system is being evaluated. Build two of the system will be integrated and tested in November and December of 2005.

DARPA has partnered with multiple service research and development organization to assist in the development, testing and transition of the DTT capability. The Air Force Research Laboratory (AFRL), the Navy SPAWAR Systems Center San Diego (SSCSD), and the Intelligence and Information Warfare Directorate (I2WD) of the US Army

Communication Electronic Command Research and Development Center (CERDEC) all are working with DARPA. The DTT System and ISR Testbed currently reside within the AFRL Sensors Directorate's Advanced Recognition Capability (ARC) laboratory. The integration and the testing of the DTT System takes place within the ARC and is conducted by AFRL in-house engineers and scientist. Current plans are to field test the system with the CERDEC.

The DTT Architecture is depicted in Figure 1. The interface to the DTT user is captured in the ISR Commanders' Interface (ISRCI) Component. Commander's guidance flows into the system through ISRCI. System input (e.g. sensor reports) enters through the Registration component. The purpose of the Registration component is to reduce sensor report errors by estimating systematic sensor biases and discovering new fiducials across all the intelligence domains. The Fusion component incorporates all available data (all-source) to discover new targets and to form estimates of target locations and identities. The Prediction component predicts target motion based on the fused track estimates and mobility models in order to anticipate future gaps in coverage where tracks might be lost. Dynamic Sensor Management (DSM) generates and maintains synchronized collection plans for all organic sensor assets in response to evolving information needs, focusing coverage as needed to maintain track on high-valued targets. The DTT ISR Testbed uses Sensor/Platform capability models to emulate the external ISR environment and generate the synthetic input needed to test the system. The Testbed and the DTT system are tightly integrated to allow Monte Carlo closed loop testing. The goal is to take a robust DTT system to military experiments and exercises to demonstrate operational utility to the Warfighter.

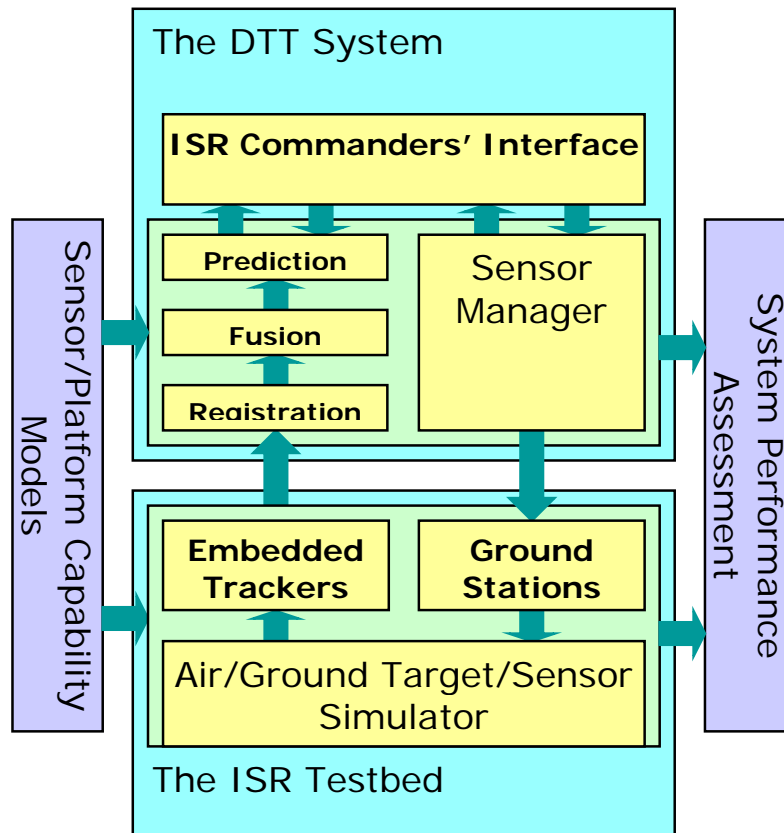


Figure 1 The DTT System and Related Components

The remainder document is structured as follows. Section two looks at the DTT architecture. Sections three and four, focus on the Fusion and DSM components and the closed loop interaction between the two. Section 5 covers evaluation and presents some recent results.

2. ARCHITECTURE

2.1. Overview

The DTT system, composed of five components (Registration, Fusion, Prediction, Dynamic Sensor Management and the ISR Commander's Interface), has been developed to run in a distributed environment with both C++ and Java applications. As many of these component's early builds were not focused on efficiency or real-time performance, this distributed environment allows the use of multiple machines to improve overall run time. In addition, this approach allows legacy functionality in either C++ or Java to be utilized without bias. As with any distributed architecture, it is critical to have an effective methodology for building object models and for allowing object models to evolve as requirements change. With both C++ and Java components, supporting the multiple implementations of these object models becomes a particular challenge. For these and other reasons, the DTT system adopted the eXtensible Distributed Architecture developed under the Adaptive Sensor Fusion program [2].

2.2. eXtensible Distributed Architecture

The eXtensible Distributed Architecture (XDA) provides a framework to:

- Allow distributed components to communicate at the object level.
- Allow direct communication between Java and C++ components.
- Automatically develop software implementations of object models.
- Allow inheritance of object attributes to extend object models.

The overall strategy is to de-couple the high-level description of the operational domain of interest (ontology) from the low-level data transport (physical schema). This ontology approach allows users to generate high-level descriptions of the entities of interest in the problem domain. Once an entity has an ontology-based definition, it can be converted to a Generic Transport Object (GTO) and instances of this entity can be transported across platforms as byte-streams (see Figure 2).

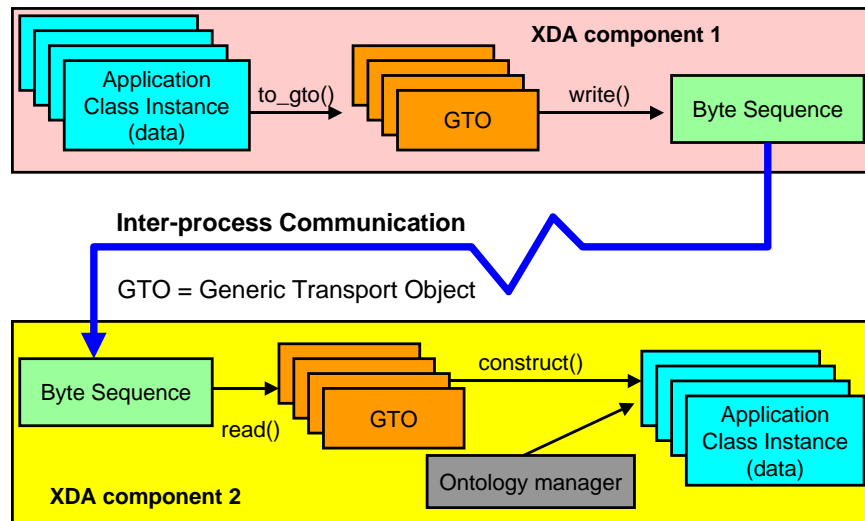


Figure 2. XDA provides a mechanism that allows distributed components to communicate at the object level.

The ontology approach allows the efficient implementation of a publish and subscribe (pub/sub) paradigm of object level information across a robust communication layer. For the DTT program, XDA utilizes a Common Object Request

Broker Architecture (CORBA) as the transport mechanism between components. Because the GTOs are byte sequences that are converted back into object level information, C++ and Java components can directly communicate if each understands how to transform the byte sequence (defined by the physical schema).

Given the importance of a consistent implementation of the physical schema, a driving concern for the DTT program was the ability to translate the Interface Control Document (ICD) into a physical-schema for converting byte sequences into object level information. For this reason, the DTT program developed an approach that allows the automatic generation of code from an ICD. The automated ICD is accomplished in a two step procedure. First, the ICD (defined in a Microsoft Word document) is compiled to an ontology definition. The second step utilizes the XDA ontology compiler to generate both C++ and Java code.

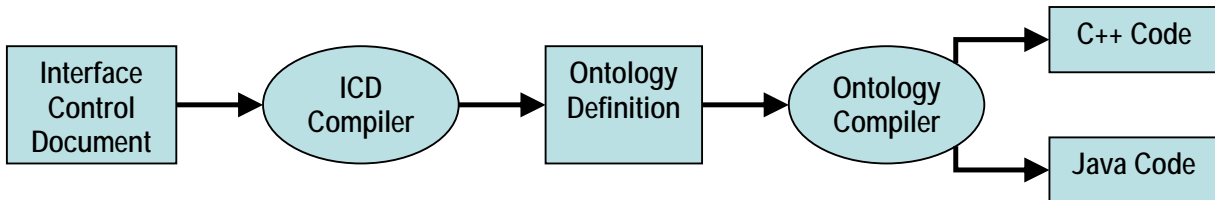


Figure 3. Automatic generation of both ontologies and code improves both development time and robustness

To further reduce complexity, interface ontologies that share many attributes are grouped together for ease of exploitation. For instance, an unregistered sensor report, a registered sensor report, and a track estimate all have attributes of: time, source, geo-location, and accuracy. While these examples may look very different to the Registration and Fusion components, they look equivalent to the ISR Commander's Interface. For this reason, the DTT system uses a single base ontology (Information Frame) that contains all of the common information with inherited specific ontologies that add additional attributes particular to details of the domain. This allows the ISR Commander's Interface to access Information Frames without concern for the details embedded for Registration's or Fusion's benefit.

3. FUSION COMPONENT

3.1. Overview

The fusion component within DTT performs the following tasks: target state estimation and sensor task generation. DTT contains two algorithms within the fusion component to perform these tasks. The All-Source Track and Identification Fuser (ATIF) provides the target state estimation while the Fusion Information Needs Determination (FIND) algorithm provide the tasks that the DSM can use to focus sensors on high priority targets. To accomplish the first task, ATIF combines data from multiple sensors to detect and identify mobile ground targets and maintain track on known high-value targets. By processing complementary data sources, ATIF produces a self-consistent, compact representation of the battle-space with low kinematic and target classification errors and high track continuity through move-stop-move target motion cycles [3]. Indeed, the data feeds of interest have complementary characteristics: SIGINT systems provide early indications of targets of interest over large fields of view, GMTI systems track moving targets, SAR and UGS trackers are effective at determining target identity and can potentially identify both stationary and moving targets while EO/ IR data can provide crucial identification confirmation if the sensor is properly tasked. Table 1 summarizes the primary properties of a set of commonly used intelligence sources indicating how they complement each other. To accomplish the second task, the FIND algorithm constructs a model of the expected value of future sensor information as a function of time, space, kinematic accuracy, classification accuracy, and geometry. These models, expressed in the form of Information Needs (INs), allow the DSM to consider multiple sensor-to-task combinations in order to maximize the expected value that the DTT system can provide.

Table 1. While no single sensor provides full information, combinations of sensor feeds allows ATIF to provide improved coverage of the battle space.

Sensor Feed	Geolocation Quality	ID Quality	Field of View	Target Motion
ELINT	Poor – Medium	Good Emitter	Wide	Typically Stopped
COMINT	Poor – Medium	Medium	Wide	Both
MTI	Good	Poor	Medium	Moving
SAR	Good	Medium	Medium	Stopped
IR/EO	Good	Good	Narrow	Both
Acoustic	Medium	Medium	Narrow	Both

3.2. ATIF Overview

ATIF receives as input the tracks and/or detection-level data from an arbitrary number of sources. Currently, the sensor types handled include GMTI, SAR, EO/IR, SIGINT, acoustic, and seismic sensors (UGS). The algorithm fuses this multi-source data into a set of all-source tracks that have improved continuity, reduced errors in kinematic state estimates, and improved identity (target type) information. Key features of ATIF’s approach include:

- A Multiple Hypothesis Tracking (MHT) approach that considers multiple association possibilities to defer data association decisions to allow time correlated data to distinguish between likely and unlikely associations.
- A hierarchical approach to data fusion allows ATIF to incorporate input tracks optimally when source-reports are available and near-optimally when only state estimates are available.
- Two-stage approach to incorporation of out of sequence measurements (OOSMs) that allows OOSMs to be efficiently incorporated as in-sequence data.
- Adaptive algorithms to focus computational resources on high priority targets.

Each source provides data sequentially to ATIF in the form of an Information Frame. While the details of the different sources differ, all Information Frames contain a frame ID, originator ID, sensor parameters (e.g., sensor location, region of regard), a collection time and a set of New Information. The use of an Object Oriented middleware such as XDA allowed multiple forms of New Information to be provided within a single, consistent fusion ontology. For the first build of the DTT system, valid forms of New Information include: raw reports, associated tracks with raw reports, and associated tracks without raw reports. In each case, the New Information included, at a minimum, a kinematic estimate (e.g., position estimate, range-bearing-range-rate, etc.) and, optionally, classification estimates (e.g., target type declaration and track status parameters). This general approach to input data allows ATIF to easily accommodate new sensor or tracker inputs as they become available.

3.3. FIND Overview

While the use of multiple platform and sensor assets improves ATIF’s coverage of the battle space, it makes closing the sensor-fusion loop more challenging. Adding to the complexity of this multi-asset synchronization is the fact that the DSM is being sent tasks from multiple sources (e.g., FIND tasks, Prediction tasks, Search tasks). For this reason, fusion must have an internal state-space estimator that can assign values to potential future information (derived from sensor tasks) that is commensurate with the expected performance improvement and is expressed in terms of meeting a set of specified commander’s needs. As multiple sensors may be capable of satisfying a given task (i.e., SAR could confirm that a target has left a staging area, MTI could confirm that a target is moving away from a staging area, EO could confirm that the particular target is moving away from a staging area), FIND’s tasks must be sensor independent, instead focusing on the information needs. This has the added value of centralizing the sensor modeling within the DSM component.

Our approach to formulating the FIND problem makes use of a stochastic dynamic programming (SDP) framework to maximize the time discounted reward J specified by

$$J = \sum_{t=0}^{\infty} \alpha^t \sum_i R(H_i(t)). \quad (1)$$

Here, the reward (R) is a function how well each of the track hypotheses ($H_i(t)$) meets the commander's goal for tracking accuracy and classification confidence. The value of meeting the commander's goal is discounted by a factor α^t to account for the preference for meeting the goals as quickly as possible. Although the goal of the FIND algorithm is explicitly stated only in terms of improving kinematic and classification accuracy the resulting policy produces three classes of tasks:

- Track maintenance: Sensors will be tasked to regularly revisit high priority target tracks to provide the kinematic information needed to maintain high quality estimates on these targets.
- Ambiguity resolution: Sensors will be tasked to provide classification information on high-value targets after periods of ambiguous interaction with confuser traffic.
- Finding new targets of interest: Low level background tasks will be generated to provide classification information on tracks with unknown classification to search for new high-value targets.

Our approach to solving the FIND problem involves a two step valuation. As the discounted reward most heavily weights near-term success, an exact one step solution for the expected reward is calculated. If only this term contributed to the valuation function, the resulting policy would be myopic - focusing only on short term rewards. For this reason, a second term is added to estimate the expected discounted reward associated with ATIF being closer to meeting the goals. In this case, we use the following heuristic approximation to the Bellman function based on the solution of the simpler but related Multi-Arm Bandit problem [4];

$$V(H_i, t) = \sum_i (\underline{y}_i^T \rho \underline{y}_i + R_{\Sigma}) \left(\frac{1}{\pi} \arctan \frac{(\Sigma_{Goal} - \sigma_i^2)}{\Gamma} + 0.5 \right), \quad (2)$$

where \underline{y}_i is the target type probability distribution, ρ is a diagonal matrix of priorities, R_{Σ} is the reward for meeting the specified tracking goals, σ_i^2 is the largest eigenvalue of the positional covariance matrix, Σ_{Goal} is the specified tracking accuracy, and Γ is a measure of the commander's time criticality. FIND's total valuation function is then given by combining the exact calculation of the one-step look ahead with the long-time approximation of the Bellman value.

Using this approach, FIND assigns values to tasks as a function of sensor parameters including: kinematic accuracy, classification precision, latency on task fulfillment, and geometry. This valuation model for different specific sensor tasks is captured in the Information Needs (INs) ontology (see below). While it might be possible to dynamically develop analytic models, FIND uses a numerical sampling approach to communicating this value function over the set of possible parameter values.

4. DYNAMIC SENSOR MANAGEMENT (DSM) COMPONENT

4.1. Overview

The Dynamic Sensor Manager (DSM) component provides a capability for automated construction of synchronized multiplatform / multisensor plans (platform routes and sensor schedules) in response to information needs generated by the other DTT system components (Fusion, Prediction, Registration, and the Commander's Interface). The DSM architecture has been designed to provide scalable responsiveness within the DTT feedback architecture. DSM is the execution control element of the DTT closed-loop control system, in that it responds to retasking information requests

and develops revised sensor collection plans. Since DSM must be responsive to the full dynamic range of tasking frequencies anticipated for the system, precautions have been taken in developing a flexible and scaleable architecture and algorithms for replan management as an intrinsic DSM capability.

4.2. Information Needs

DSM is driven by Information Needs generated by the other DTT components. An Information Need (IN) is a prioritized, sensor-independent tasking request characterized by spatial/temporal satisfaction criteria (i.e., quality requirements and other context dependent collection constraints). Depending upon the source, the level of abstraction (or specificity) of an IN can vary widely. For example, a command level IN might specify a high-level information requirement in an imprecise manner such as “Find, identify, and track all SCUDs in area of interest (AOI) 1 that start moving over the next 10 hours.” On the other hand, the Fusion component might request an IMINT collection on a specific target under track within the next five minutes in order to resolve critical tracking ambiguity. The level of specificity for a Fusion generated IN might require that collected image quality ensure a specified probability of correct classification for the target under track.

There are a few features common to all INs. For example, each IN will specify a collection target. The collection target defines the location, in ground coordinates, where the information is to be collected, which may be a specific spot, a line of communication (LOC), an area, or a predicted track (moving target). Each IN specifies a time window during which the collected information is of value to the requestor, together with a value (ordinal scale of importance) or priority (discrete ranking of IN categories) for the satisfying the information need. As a specialization, an IN may specify a value as a function of the collection quality or some geospatial or temporal attribute of the collection.

One of the most critical DTT interactions is between the DSM and the Fusion component. In order to support the mission of tracking high valued targets, an interface has been developed in which Fusion provides continuous target track prediction updates to DSM, in addition to future collection valuation tables. This interaction provides a continuous collection plan management and plan repair capability focused on maintaining track on selected, high value targets. A Fusion information need may require either kinematic or target classification information. In either case, the IN will be associated with a short-term moving target prediction (less than five minutes) and will have value represented as a function of collection quality. Table 2 is a sample value table provided as part of a Fusion kinematic need¹. A Fusion generated IN represents a request to provide information which is critical to maintain track on a target of interest.

Quality (TLE)	Revisit Period (seconds)			
	5	10	15	20
5 Meter Accuracy	1769	1651	1483	1265
25 Meter Accuracy	1692	1397	628	385
125 Meter Accuracy	1185	272	124	79

Table 2. Fusion will provide a table of relative values of information based on the quality of the collection and the achieved revisit rate.

4.3. DSM Top-Level Design

The top level design of DSM includes the following two main components:

- **Replan Manager** - Controls the dynamic execution of DSM, enabling timely asynchronous planning. The decisions within this component include both when (based on plan and asset state and incoming task queue) and how (which tasks, whether or not to change routes, and for what plan horizon) to replan. In addition, the

¹ In the next build of the DTT system, this interface will be generalized to allow value to be a function of both TLE and classification confidence within a single generic information need.

Replan Manager is responsible for maintaining the clock, making sure plans are delivered on time, health status messages, and the asset state (receiving the state from PASTE and extrapolating to the plan start time).

- **Plan Generator** - Responsible for actually computing optimal platform routes and sensor schedules based on guidance from the Replan Manager. The algorithms within the Plan Generator must support wide area surveillance balanced with focused track hold planning.

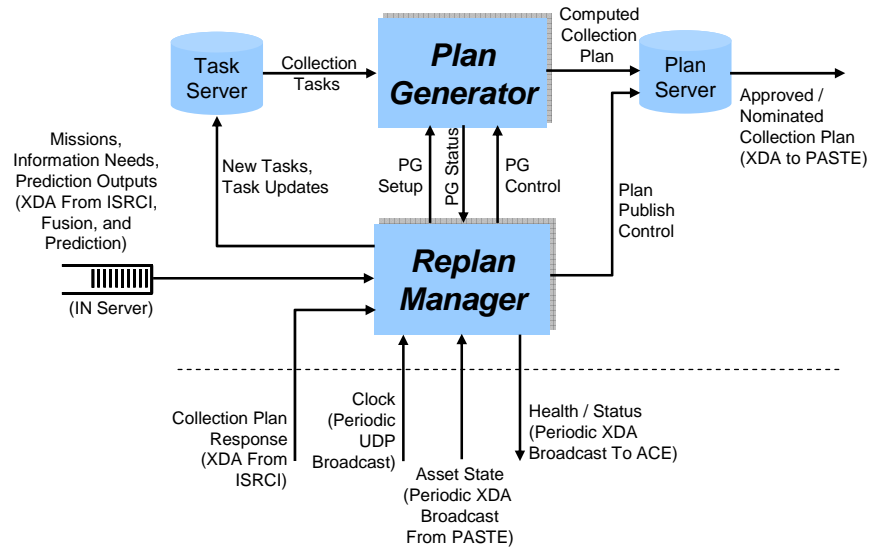


Figure 4: DSM High-Level Architecture.

4.4. Plan Generator Overview

In [5], the Multi-Asset Synchronizer (MAS) developed under the DARPA AIM program was presented. In that paper, a framework for ISR planning was introduced that is based on a hierarchical decomposition of the planning problem. In the first phase, platform routes are generated via approximate dynamic programming (ADP). In the second phase, detailed sensor scheduling is performed on the fixed routes generated in the first phase. The DSM Plan Generation component extends the MAS route generation capability to provide continuous, responsive plan generation for tactical airborne assets. DSM plan generation algorithms provide a scaleable capability to address a richer set of information needs and adapt to a rolling planning horizon. The DSM plan generation algorithm has been developed to enable interrupt and recovery operation to support real-time dynamic replanning by integrating the detailed sensor scheduling phase into the route generation phase.

The Network-Flow Problem (NFP)-based value-to-go approximation used in MAS has been shown to be effective for large-scale ISR planning problems covering long planning horizons over a large geographical area in which many assets are being used to collect thousands of tasks. The problem is significantly different for DSM, though. The DTT problem is focused on a relatively small area of interest, concerned with look-ahead horizons no longer than 60 minutes, and controlling a relatively small number of airborne platforms against a few time-sensitive targets. This suggests that a value-to-go approximation that is less global in outlook than the MAS NFP, but higher fidelity locally (in time and space) might be more appropriate for DSM. Within DSM, a greedy roll-out algorithm is being employed to accomplish this goal (see [6] for details).

The focus for DSM-controlled sensors in the DTT program is EO/IR and Radar. While other sensors (SIGINT and UGS) play in the DTT testbed, they are not directly controlled by DSM. In Table 3 we summarize the collection quality metrics of interest to the Plan Generator component of DSM. In order to support these requirements, it was necessary to construct models for each sensor mode that relate collection geometry and target types of interest to the quality measures.

Quality Measure	Description
Probability of Detection	Probability of detecting targets of interest
Resolution	For imagery (SAR, EO/IR), the ground sampled distance of the image
Probability of correct classification	Probability of correctly classifying targets of interest
Target location error	Requirement on the length of the semi-major axis of the EEP90 ellipsoid

Table 3: DSM Collection Quality Measures.

5. PREDICTION

Target tracking typically includes filtering collected measurements, estimating the current target location, and predicting the future target path. The tracker prediction time horizon is a few time steps ahead so as to assist in the placement of the platform sensors. Since the DTT system is controlling many assets and tracking numerous targets, a new effort was included to determine if a long-term prediction of target paths could be ascertained to help DSM with platform sensor placement.

The main reason why the prediction component was added includes the fact that if an sensor asset was tasked to look away from a tracked target, it might not return for many time steps and thus the target track estimate covariance would have exceeded the sensor field of view. When such a large covariance exists, the track maintenance problem defaults to a search for a “new” target. A long-term prediction was deemed necessary to reduce the search space of a tracker for re-acquiring the target. Thus, the problem is not a true new target search, but one of “re-acquiring” target search.

The initial attributes of the prediction component include using road networks, target paths, and context information to reduce the search space to the most likely target locations. Road networks and terrain information provide lines-of-travel information which affects the velocity of the target prediction. Target paths include the kinematics and the expected use of the vehicle (i.e. personnel carrier moving to the front line). A third component is context information which includes target groups traveling together, zones of engagement, and mission priorities. Together, the three attributes of prediction effectively reduce the possible target locations for extended time horizons to assist the sensor manager in directing sensors for re-acquiring tracks.

6. EVALUATION

To evaluate the DTT system, we address various performance metrics over extended operating conditions (EOCs). The performance metrics that satisfy DTT objectives are 1) geolocation accuracy, 2) track life, and 3) commander needs satisfaction. Since geolocation accuracy is dependent on sensor model fidelity, a functional analysis of improved target precision based on registered sensor data was completed. Commander needs satisfaction is based on the user, the scenario context, and a INs decomposition, all of which are being defined. Thus, for the current DTT assessment, track life was determined to be the critical metric of evaluation. Track life is measured for only moving targets since stationary targets would have an infinite track life. The *moving target track length* (distance) was processed over various maneuvering targets (i.e. different speeds) to get an estimate of a normalized track life.

The extended operating conditions include 1) number of targets, 2) nominated area of interest (NAI), and 3) sensor models. The NAI and high value target number was set for each evaluation run. The simulation models incorporate environment conditions, sensor exploitation capabilities, and target behaviors. Since the DTT evaluation goal was to assess a real-world performance capability, the most optimistic sensor assumptions were not used, but rather expected sensor operating performance capabilities given the associated environment details. The DTT tracker (Fusion Component) conducts track-to-track association, which requires a measurement-to-track single INT tracker input that was modeled as a simple parametric tracker. To determine the DTT performance over the EOCs, a Design of Experiments (DOE) was formulated using different scenarios

5.1. Scenarios

To effectively evaluate system performance, three types of parameters are assessed: (1) *constants* (e.g. NAI); (2) Monte Carlo *variables* (e.g. sensor variation), and (3) *factors* that change (e.g. number of targets) [7, 8]. **Table 4** shows the DOE analysis for the target, sensor, and environment EOCs. Constant parameters were determined from either unmodelable effects, issues that did not affect the outcome, or real-world operating conditions. Constant target behavior includes emissions and on-road travel while variable target behavior includes move-stop-move cycles, paths, and speed. The sensors and platforms are a result of the mission scenario. The sensor variables included exploitation accuracy and sensor factors including an On/Off bias selection. The largest category of scenario OCs is from the environment which includes weather, NAI, facilities of target initiation/termination, terrain/road conditions, and ‘no-fly zones’. For each scenario, the NAI was fixed and available terrain and weather data provided.

OC Category	Parameter	Flat Terrain	Mountainous
Targets	Targets	6, 12, 25	1, 2, 5
	Moving Confusers (Dens. / AOI)	Low(10), 50 , High (1000)	Low (0), Med (10), High (25)
	Routes-Stop-Move	Variable	Variable
Sensors	Initial Start Points	Variable	Variable
	Bias	On/Off	On/Off
Environment	‘No Fly’ Zones	Variable Area Locations	Variable Area Locations

Table 4: DOE of Test Scenarios

Utilizing the static environmental conditions, different target densities and sensor models were selected to test a variety of scenarios and evaluate the DTT system performance. Figure 5 represents the design space of different scenarios available for a given MIDB (Mission Data Base), NAI, and ATO (air task order of mission objectives).

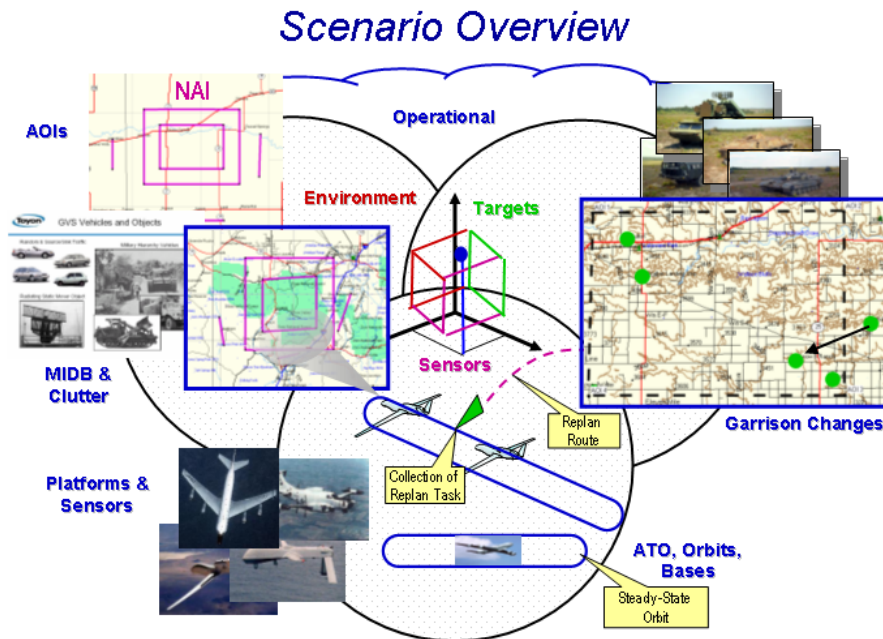


Figure 5: Scenario Parameters.

After the selected scenario conditions for each run were determined, a simulator was used to instantiate the truth information, such as actual target behaviors and sensors measurements. The simulator used the mean values and variance bounds associated with selected factors to create a half-day scenario which afforded a longitudinal performance analysis based on ergodic assumptions.

5.2. Metrics

The key metric was *track life*. Track life is a function of many parameters such as prioritized target number, NAI, target behavior, and number of confuser targets. Given that simulation time was discounted, we assessed the track-life metric over timeliness (responsiveness), accuracy (geolocation), and confidence (target identification). Timeliness and confidence were combined into a single track life metric output as it was assumed that if the DTT system could maintain track over long time interval, it must be labeled correctly (i.e. confidently identified) and tracked (i.e. DSM allocates sensor positions to afford timely updates). The trick to the analysis was designing the track-to-truth associator. For discussion brevity, we basically selected a tight validation region that forced an accurate position target requirement and assessed whether the DTT track outputs fell within the validation region. Since the system was evaluated over a number of DOE cases, we chose to normalize the results based on the number of scenarios run (i.e. an aggregated set of runs would also aggregate the number of targets and metrics). The track length metric would be determined for all the tracks, including the confuser targets. Thus, we chose as the key metric

$$\text{Track Length} = \frac{\sum \text{object track length}}{\sum \text{objects}} \quad \text{where Track Life (Hr)} = \frac{\text{object track length (Km)}}{\text{velocity (km/hr)}}$$

Choosing the normalized length (over targets and distance) gives a realistic analysis of the overall DTT performance for various conditions. Note that the evaluation of the specific functions of DSM, tracking, and registration were not conducted as that was deemed as internal processes of the algorithms themselves.

5.3. Results

The preliminary results, shown in Figure 6, demonstrate that the DTT system was able to correctly track the targets of interest over many runs and operating conditions.

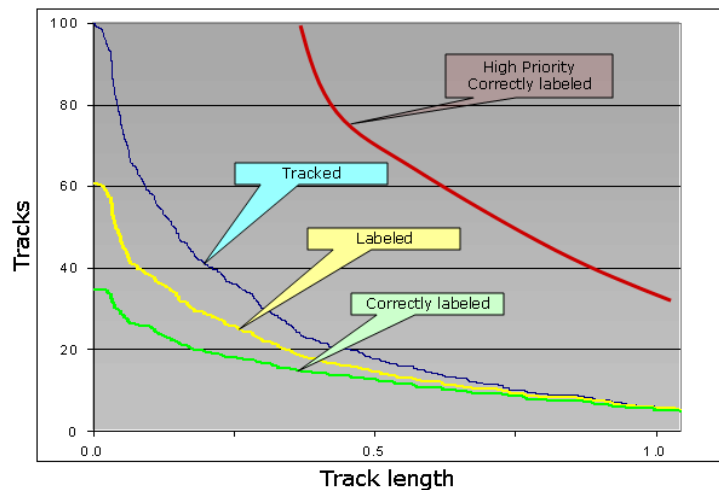


Figure 6: Scenario Parameters.

Three observations are noted. The first is that the correctly labeled tracks are a subset of the entire tracked set. As the DTT system initiates tracks, a confirmatory ID is not available until enough evidence is accumulated to positively ID the target. Thus, we would not expect all tracks to be labeled at all times (tracked to labeled difference). Secondly, since

targets have the ability to cross, there are cases in which the positive track label is assigned to another target for a short length of time. During the interval of miss-association, the labeled short track length is attributed to another target. However, after an incorrect label is assigned, the DTT system realizes the conflict and initiates new track-ID pair and correctly reassociates the correct ID to the target (labeled to correctly labeled). The final observation is that the extended track life is a result of the simultaneous track and label information. Short tracks that are not targets of interest receive few requests for position updates and thus are dropped. The targets of interest are repeatedly measured and thus have an extended track life (which is demonstrated as the merging of all three cases to the longest track segment). Future analysis will include discounting the short confuser tracks in the normalized track life, determining a minimum time threshold for considering a target tracked (i.e. 5 minutes); and a focus on only the highest priority targets. The DTT goal of “extending” track life is notional plotted as reference to the baseline (top right) to show that if the high-priority targets are selected, DSM would be able to allocate sensors on target producing longer tracks, while Fusion would be able to consistently (relative normalization of number of tracks) exclude short track segments.

7. CONCLUSIONS

To reduce the kill chain timeline, DTT has focused on closing the loop between a Fusion-based tracker and a Dynamic Sensor Manager. The key developments of the closed-loop coordination include the automated requesting, selection, and satisfaction of Information Needs. Additional factors that drive the successful closed-loop coordination for extending track life include sensor registration, long-time horizon prediction, and an ISR commander’s interface.

ACKNOWLEDGEMENT

The performance evaluation scenario design was developed by Mike Pribilski, Track-to-truth Associator programmed by Brian Roscoe and Josh Gunsett, and scrutiny of the sensor models by Ron Dilsavor and Pat Toole. Thanks goes to Steve Haupt and the GD team for processing all the scenarios and integrating the various algorithms.

8. REFERENCES

- [1] DARPA Dynamic Tactical Targeting, *Proposer’s Information Package*, BAA #01-43, September, 2001.
- [2] B. Krikeles, A. Lusignan, E. Starczewski, “A Framework For Distributed Data Fusion”, *2001 MSS National Symposium on Sensor and Data Fusion*, June 2001.
- [3] Coraluppi, S., C. Carthel, M. Luetgen and S. Lynch, “All-Source Track and Identity Fusion,” in *Proceedings of the National Symposium on Sensor and Data Fusion*, June 2000, San Antonio TX, USA.
- [4] Bertsekas, D. P., *Dynamic Programming and Optimal Control*, Athena Scientific, 1995.
- [5] C. T. Lawrence, W. H. Bennett, A. J. Newman, and T. J. Budris, “Synchronized Collection Management of an ISR Confederation,” *Proceedings of the MSS 2002 National Symposium on Sensor and Data Fusion*.
- [6] C. T. Lawrence, W. H. Bennett, and M. P. Merriman, “Dynamic Sensor Management for Coordinated Target Detection and Tracking of High Value Targets,” *Proc. MSS 2003 National Symposium on Sensor and Data Fusion*.
- [7] T. D. Ross, J. Bradley, L. Hudson, M. O’Conner, “SAR ATR - So What’s The Problem? - An MSTAR Perspective”, *Proc. SPIE*, Vol. 3721, Algorithms for Synthetic Aperture Radar Imagery VI, April 1999.
- [8] E. Blasch, “Performance Metrics for Fusion Evaluation,” *Proc. MSS 2003 NSSDF*.
- [9] E. Blasch & P. Hanselman, “Information Fusion for Information Superiority,” *IEEE NAECON*, pp. 290-297, 2000.
- [10] R. Tenney, Hebbert, R., N Sandell, JR. “A tracking filter for maneuvering sources”, *IEEE Tans. Auto. Control*, Vol.22, 2, April 1977, pp. 246-251.

* Phillip.Hanselman@wpafb.af.mil, (937)255-5668, x4056