Assessment Of Data Fusion Systems

4 authors, including:

Gee-Wah Ng
DSO National Laboratories
89 PUBLICATIONS   **1,254** CITATIONS

Josiah Tan
DSO National Laboratories
7 PUBLICATIONS   **45** CITATIONS

Assessment Of Data Fusion Systems

Gee Wah Ng, Chung Huat Tan, Thiam Poh Ng, Shun Yi Siow
Advanced Analysis And Fusion Laboratory
DSO National Laboratories
20, Science Park Drive, Singapore 118230
{ngeewah, tchunghu, nthiampo, sshunyi}@dso.org.sg

Abstract – With growing emphasis on research and applications in data fusion, it is important to know how to measure fusion system performance. While much of the current research is involved with evaluating fusion systems based on the quality of their outputs, this paper describes the expansion of this scope to include the complexity of the inputs. Since the quality of the outputs is highly dependent on the difficulty of the input scenario, the methodology described in this paper provides us with a more well-rounded interpretation of fusion system performance. Moreover, characterization and quantification of input scenarios allows for the creation and variation of test scenarios, leading to more efficient optimization of fusion systems. A series of complexity and performance metrics were designed according to a common standard to measure the various aspects of the input scenario and the fusion system solution. Finally, these metrics are combined to obtain an assessment index which provides a description of the fusion system performance.

Keywords: Tracking, association, estimation, performance evaluation, performance assessment, data fusion.

1 Introduction

Data fusion (DF) refers to the combination of data and information from more than one source. With the use of multiple sources, we are able to obtain even more accurate information, better situational awareness and decreased data uncertainty in comparison to the use of only a single source. While DF systems are widely used in medical, military and even financial fields [1], the fusion system of interest in this paper is a tracker system. The purpose of this fusion system is to trace the input scenario which comprises the trajectories of all the objects within a given area or space. Following which, the system attempts to reconstruct the trajectories of these objects and produce a global track picture which details the motion of the targets.

Throughout the entire run-time of the scenario, data about the changing scenario is obtained by sensors which are located on platforms. These platforms may be stationary and positioned on the ground or moving together with the targets within the scenario. This information is relayed to the DF system in the form of sensor reports and is received by the DF system at time instants which correspond to the respective sensor update rates. Each report thus consists of target updates which provide information about the targets at that specific time. Using the received updates, the DF system then incorporates all the information together to create the global track picture.

With the increasing use of DF systems, it is important that data fusion algorithms and systems are thoroughly tested and evaluated before deployment. The main aim of this paper is to present an evaluation methodology which will enable DF systems to be assessed. In addition, this will also provide a basis for DF systems to be compared in the event of having to decide between competing systems.

To evaluate a DF system, the approach of measuring the quality of the output in context of the difficulty of the input is taken. In the case of the DF system, the reconstructed global track picture and its constituent tracks form the externally observable outputs of the fusion system [2]. The input, on the other hand, is the real-time or computer-simulated scenario being tracked by the sensors. By assessing output and input separately, we are able to come up with performance and complexity metrics which characterize the capability of a DF system.

The complexity metrics presented in this paper measure the difficulty of the input scenario and correspond directly to the features of the scenario. The performance metrics judge the quality of the global track picture produced by the DF system. Sections 3 and 4 discuss these metrics in greater detail. Figure 1 shows the variables which affect the quality of the DF system output. However, this paper does not consider every factor in the complexity and performance metrics but only extracts those factors which are of relevance to the DF system.
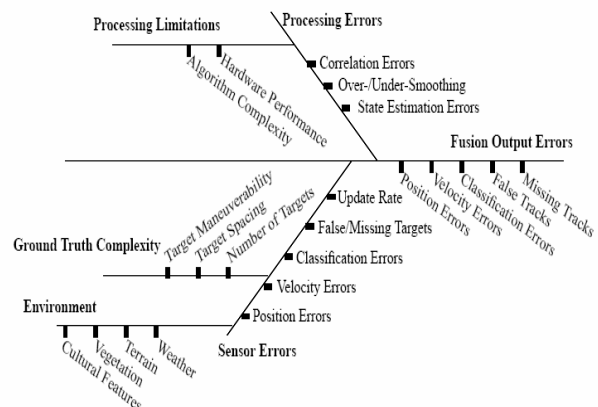

Figure 1: Contributing Factors Affecting Data Fusion System Output Performance [2]

We conclude this paper in Section 5 with the application of the evaluation tool on a few computer-simulated test scenarios and an explanation of the results obtained.

2 Assessment Methodology

To provide a basis of comparison for all the metrics, we decide on a common track-based standard for scoring every metric. As such, for every metric that is considered, each track will be assigned a score between 0 and 1 for normalization purposes. This will not favour any factors or parameters since each track would only contribute a score within this range. Separate standards of scoring metrics would lead to inaccuracies when combining or comparing these metrics together. Consequently, since every track is scored on the same scale, the resulting complexity and performance metrics can be compared without bias.

The proposed measure of evaluation comprises two indexes – the quantity assessment index and the quality assessment index. A smaller index indicates better DF system performance.

$$\text{quantity assessment index} = \frac{\text{quantity performance metric}}{\text{quantity complexity metric}} \quad (1)$$

$$\text{quality assessment index} = \frac{\text{quality performance metric}}{\text{quality complexity metric}}$$

Quantity assessment index ("How Many?") allows us to quantify the capability of the DF system by considering the number of targets and platforms and also reconstructed tracks. Quality assessment index ("How Good?") relates the accuracy and precision of the reconstructed tracks to the complexity of the target trajectories. Interpreted together, both these indexes provide a succinct description of the DF system performance.

Table 1 summarizes the metrics used for computing the quantity and quality assessment indexes and how the main metrics are combined from their sub metrics. Further details of these metrics can be found in Sections 3 and 4.

Main metrics	Sub metrics
Quantity complexity metric	No. of targets + No. of platforms
Quality complexity metric	Target separation + Platform separation + Target-Platform separation + Sensor report complexity
Quantity performance metric	No. of false alarms + No. of missing tracks
Quality performance metric	Correlation accuracy + Individual track quality

Table 1: Summary of metrics.

Overall assessment is comprised of the quantity and quality assessment indexes as they describe different aspects. Knowledge of both these indexes is vital for full appreciation of DF system performance. If the quantity assessment index is large and the quality assessment index is small, we can conclude that, in comparison to the input scenario, the output global track picture is incomplete and probably contains erroneous tracks. However, the reconstructed true tracks mirror the actual target paths closely. Conversely, a small quantity assessment index and large quality assessment index relates to a numerically accurate global track picture which has tracks which deviate greatly from the actual target paths.

3 Complexity Assessment

The performance of a data fusion (DF) system is graded on the accuracy of the reconstructed global track picture. One important consideration in interpreting the performance of a DF system is the complexity of the scenario it attempts to reconstruct. Given a more complex and difficult scenario, a DF system would be expected to give poorer performance and less accurate track picture reconstruction compared to a simple scenario input. As such, assessment of a DF system solely based on its output performance only shows one side of the picture. Accurate evaluation of a DF system can only be performed by taking the difficulty of the input scenario into account.

Two main complexity metrics are proposed in this paper. The first metric, the quantity complexity metric, measures the difficulty of the scenario in terms of the number of objects within the scenario. Here, the scenario is evaluated at face value, without sensor reporting. The second metric, the quality complexity, measures the difficulty based on the trajectories of the targets as well as through the reports received from the various sensors.

3.1 Quantity Complexity

Within a scenario, a greater number of objects means that at any given time, the DF system would have to deal with a larger number of target updates and fuse more data together to reconstruct the global track picture. This would result in a more complex task with room for more errors. The scope of objects within a scenario is not limited to targets only. In fact, the number of platforms in the scenario also contributes to the degree of difficulty.

3.1.1 Number of Targets

Difficulty of the scenario increases with the number of enemy targets due to the increasing load on the sensors. When reports are received by the DF system, they have to be correlated with fused tracks that already exist in the fusion system. The DF system has to decide if a newly received report should be used to update an existing fused track, or if it should be used to initiate a new track. This decision-making process is known as data association. The increased number of reports received would lead to a greater probability of error in the data association process.

3.1.2 Number of Platforms

This metric measures the number of friendly platforms in the input scenario. Reports received from the various sensors have to be distinguished as originating from the targets or from friendly platforms. Hence, increasing the

number of platforms also increases the difficulty of data association.

3.2 Quality Complexity

At any given time, an object can be characterized by its location and its maneuverability. Within a given scenario, the position of a target and its relative position to other targets define its location. Variables which affect maneuverability are the velocity, acceleration and sudden changes in direction of the objects within the scenario. The quality complexity metric thus aims to combine factors which affect the nature of movement and the state of the targets.

3.2.1 Target Separation

Throughout the entire duration of the scenario run-time, the scenario is sampled at regular time intervals. At a certain sampling instant, a snapshot of the scenario is taken and we are able to calculate the distance between an arbitrary pair of targets (i,j). This distance is termed the inter-target distance. In addition, an inter-target distance threshold is also specified. This threshold represents the minimum distance between a pair of targets such that they are discernible as distinct and separate targets.

The inter-target distance threshold is pegged to a minimum score of 0, and any inter-target distance that falls above this threshold is given a score of 0. The least possible inter-target distance of 0m is pegged to a maximum score of 1. The inter-target score between two targets i and j at a given time t is given below in (2).

$$\text{inter target score}_{i,j}(t) = \begin{cases} 0 & \text{if } \text{dist}_{i,j}(t) \geq r \\ 1 - \dfrac{\text{dist}_{i,j}(t)}{r} & \text{if } \text{dist}_{i,j}(t) < r \end{cases} \quad (2)$$

where r is the inter-target distance threshold, and $\text{dist}_{i,j}(t)$ is the distance between targets i and j at a given time t.

Having defined the method of scoring inter-target distances, we can now calculate a target-pair separation metric for a pair of targets (i,j) by taking the average of the inter-target scores obtained from each sampling instant over the total number of samples taken. This target-pair separation metric quantifies the ease with which a pair of targets can be distinguished from one another over the entire scenario run-time.

$$\text{target pair separation metric}_{i,j} = \frac{\sum\limits_{t \in \text{samples}} \text{inter target score}_{i,j}(t)}{n} \quad (3)$$

where n is the number of time instances.

Finally, the target separation metric is calculated by taking the average of all the target-pair separation metrics over all the target pairs and multiplying the result by the total number of targets.

$$\forall i, j \in \text{targets}, (i, j) = (j,i) \wedge i \neq j,$$

$$\text{target separation metric} = \frac{\sum \text{target pair separation metric}_{i,j}}{N_p} \times N_k \quad (4)$$

where N_p and N_k are the total number of target pairs and targets in the scenario respectively.

3.2.2 Platform Separation

Similar to the target separation metric, decreasing distance between platforms leads to greater complexity of the scenario. The crux of DF relies on obtaining information from several sensors so as to obtain complementary data which will lead to a wider awareness of the situation and a more comprehensive picture. An example of the use of complementary data is the use of two video cameras positioned at different angles. The two cameras provide complementary information as each camera sees a different perspective of the same object due to the angle each camera is viewing. With the two sets of information put together, it may be possible to construct the 3-D picture of the object [1].

However, if the platforms are located in close proximity to one another, the data received would not provide improved insight about the scenario. In the worst case, if all the platforms are clustered together, the data obtained from the sensors on all the platforms would be very similar and perhaps only be as good as data from a single sensor. As such, the platform separation metric is scored in the same manner as the target separation metric with (2), (3) and (4) modified appropriately.

3.2.3 Target-Platform Separation

Target-platform separation quantifies the difficulty of obtaining accurate target updates due to the distance between the platforms and targets. Each sensor has a maximum operating range, above which the sensor would not be able to locate the target. However, within the range, the difficulty of locating a target is assumed to vary linearly with sensor-target distance since the ease of sensor coverage decreases with increasing distance between the platform and target.

As in determining the target and platform separation metrics, the scenario is sampled at fixed time intervals. At each sampling instant, we calculate the distance between a platform k and a target j. We then calculate the average distance between platform k and target j.

$$\text{average distance}_{j,k} = \frac{\sum\limits_{t \in \text{time samples} \mid \text{distance}_{j,k}(t) < r} \text{dist}_{j,k}(t)}{n} \quad (5)$$

where r is the maximum operating range, $\text{dist}_{j,k}(t)$ is the distance between a target j and a platform k at a time instant t, and n is the number of target-platform distances which are less than r.

In (5), if the target-platform distance is within the maximum operating range of the sensor, we use the target-platform distance for in the calculation of average $\text{distance}_{j,k}$. On the other hand, if the target-platform distance is greater than the maximum operating range, we ignore the reading. The purpose of this is to prevent over-estimating the level of difficulty of the scenario. Targets outside the maximum operating range of a sensor are not detectable by that particular sensor. Thus, they do not contribute to the complexity of the scenario. As such, we calculate an average distance of a target j to a platform k by taking the average of all the platform-target distances that are within the maximum operating range.

Next, we calculate the average distance between a target j and all the available platforms. This enables us to calculate the mean distance of a target from all the platforms throughout the entire run-time of the scenario.

$$\text{target platform dist}_j = \frac{\sum_{k \in \text{platforms}} \text{average distance}_{j,k}}{N_k} \qquad (6)$$

where N_k is the total number of platforms in the scenario.

The target-platform score for a target j is then obtained by measuring its target-platform distance on a scale.

$$\text{target platform score}_j = \begin{cases} 0 & \text{if target platform dist}_j > r \\ \dfrac{\text{target platform dist}_j}{r} & \text{if target - platform dist}_j \leq r \end{cases} \qquad (7)$$

where r is the maximum sensor range.

The target-platform separation metric is then calculated by adding up all the target-platform scores of all the targets in the scenario.

$$\text{target platform score} = \sum_{j \in \text{targets}} \text{target platform score}_j \qquad (8)$$

3.2.4 Sensor Report Complexity

The aim of sensor report (SR) complexity is to measure how difficult it is to reconstruct continuous tracks from often sporadic, discontinuous sensor reports. While the previously discussed quality complexity metrics define the location of the targets and platforms, the SR complexity metric measures the maneuverability of the targets in the scenario. This metric is calculated in two stages – report-to-report variation and correlation complexity.

3.2.4.1 Report-To-Report Variation

Report-to-report variation details the change between successive updates on the same target. In the most ideal case, these variations between successive updates originate purely from target motion and maneuverability. However, in reality these variations are due to both true target maneuverability as well as sensor errors and imperfections.

Updates on a certain ground truth target j may be received at arbitrary times $T_{k,j}$ where k denotes the kth update received. Target updates are reported by the sensors at their respective sensor update frequencies, which may differ from sensor to sensor. While the calculation of previous metrics requires the sampling of the scenario at regular time intervals, this method of calculating report-to-report variation does not require that each update be received at fixed time intervals.

We define the instantaneous report-to-report variation of a target j at time $T_{k,j}$ as the change in its position over the time interval between $T_{k,j}$ and $T_{k-1,j}$.

$$\text{instantaneous report - to - report variation}_{k,j} = \frac{\text{dist}_j(T_{k,j}, T_{k-1,j})}{T_{k,j} - T_{k-1,j}} \qquad (9)$$

where $\text{dist}_j(T_{k,j}, T_{k-1,j})$ refers to the distance between the positions of target j at times $T_{k,j}$ and $T_{k-1,j}$. From (9), we can see that calculation of the instantaneous report-to-report variation can only proceed after the second report is received so that a time interval between successive reports can be defined.

For each target in the scenario, we then calculate the target report variation $\text{trv}_j(T_{k,j})$, which is the average of the instantaneous report-to-report variation calculated over a sliding time window. The sliding time window is defined as the last n number of time instances in which updates were received. This calculates a moving average of instantaneous report-to-report variations of a target j over the most recent n updates, namely $T_{k-n+1,j}$ to $T_{k,j}$. If the number of updates received so far is less than n, we calculate the target report variation based on the updates received so far.

$$\text{trv}_j(T_{k,j}) = \begin{cases} \dfrac{\sum_{i=k-n+1}^{k} \text{instantaneous report - to - report variation}_{i,j}}{n} & \text{if } k > n \\ \dfrac{\sum_{i=2}^{k} \text{instantaneous report - to - report variation}_{i,j}}{k+1} & \text{if } k \leq n \end{cases} \qquad (10)$$

where n is a user defined parameter which states the number of time intervals desired in the sliding time window.

The advantage of this method is that it allows us to generate the maneuverability trend of a target j. By using the most recent n updates, we capture the variation in the most recent movements of the target. This in turn provides us with a good prediction of the subsequent movement of the target since we have the knowledge of the target's most recent kinematic history. Using a fixed value for target maneuverability would discount the fact that the target may perform movements of widely varying maneuverability over the whole duration of the scenario. Hence, there would be difficulty in choosing an appropriate fixed value.

3.2.4.2 Correlation Complexity

As target updates are received in real time, the DF system does not possess any prior knowledge of which target each update originated from. Hence, there exists difficulty in correlating the newly received updates to existing fused tracks. This is especially the case if targets are close to one another or if the targets exhibit unpredictable and irregular movements. The correlation complexity metric aims to give us an idea of how difficult this association process is. Although this metric gives us the average number of targets that a target update is associated with, the actual number is dependent on the data association algorithm used by the DF system. However, since we are trying to classify the difficulty of the scenario without taking any other factors into account, we do not use the association algorithm of the DF system. At time T_k, a set of new updates $R_j(T_k)$ is received by the DF system through the various sensors reports. Firstly, we calculate a confusion set $F_j(k)$ for each target j. This confusion set consists of all the updates i from $R_j(T_k)$ which are within the target report-to-report variation of the updated target j.

$$F_j(k) = \{ i \mid \text{dist}(R_j(T_{k-1,j}), R_j(T_k)) < \text{trv}_j(T_{k-1,j}) \times (T_{k,j} - T_{k-1,j}) \} \qquad (11)$$

$F_j(k)$ is the set of newly received updates i in which the distance between the newly received update and the most recent update of target j is less than the instantaneous report-to-report variation. Thus, this provides us with a subset of updates from $F_j(k)$ which reside within the possible sphere of movement of target j. Possible

difficulties in correlating updates with existing tracks arise when there are either none or more than one updates in each confusion set. In either of these cases, it is not clear which track the update should be related with, leading to difficulty in data association.

Next, we calculate the number of confusion sets that each newly received update i falls into.

$$NF_i(k) = | \{ j | i \in F_j(k) \} | \qquad (12)$$

The number of confusion sets which a report i is a member of corresponds to the number of targets in which the update i could be associated with. As a result, the number of confusion sets an update i falls into ranges from 0 to the total number of targets. The larger the number, the greater the complexity since there are more choices of targets for a update to be correlated to, thus creating a greater probability of associating the update to the wrong target.

Over the course of the scenario run-time, a total of m updates would be received by the DF system. As a result, we can calculate the correlation complexity metric by taking the average number of confusion sets which a report will fall into. This is done by averaging the correlation complexity over the m updates.

$$\text{correlation complexity metric} = \frac{\sum_{i \in updates} NF_i(k)}{m} \qquad (13)$$

where m is the total number of updates received by the DF system throughout the entire duration of the scenario run-time.

For example, let us assume that there are 5 targets (A, B, C, D, E) in a given scenario and that the only set of updates obtained occurs at a certain time T_k. A total of 8 reports (r1, r2,..., r8) are received from the various sensors.

Step 1: Create confusion sets for each target from (11)
$F_A(k) = \{r1, r3, r6, r8\}$
$F_B(k) = \{r2, r3, r5\}$
$F_C(k) = \{r1, r4, r5, r6, r8\}$
$F_D(k) = \{r3, r5\}$
$F_E(k) = \{r2, r6\}$

Step 2: Calculate the number of confusion sets that each update falls into from (12)
$NF_{r1}(k) = 2$
$NF_{r2}(k) = 2$
$NF_{r3}(k) = 3$
$NF_{r4}(k) = 1$
$NF_{r5}(k) = 3$
$NF_{r6}(k) = 2$
$NF_{r7}(k) = 0$
$NF_{r8}(k) = 2$

Assuming that the only set of updates obtained from the scenario occurred at *time T_k*, we can now calculate the correlation complexity metric which is the average of the values obtained in step 2 from (13).

Correlation complexity metric = (2 + 2 + 3 + 1 + 3 + 2 + 0 + 2) / 8 = 1.875

Of course, for any given a *scenario*, we would have updates generated every time any sensor sends a report.

We now show the case in calculating the correlation complexity metric where reports are generated from T_{k-1} to T_{k+1}. Each time instant may yield a different number of updates. For instance, in comparison to the 8 updates received at T_k, only 4 updates were received at T_{k-1}.

T_{k-1}	T_{k+1}
$NF_{r1}(k-1) = 3$	$NF_{r1}(k+1) = 2$
$NF_{r2}(k-1) = 4$	$NF_{r2}(k+1) = 3$
$NF_{r3}(k-1) = 1$	$NF_{r3}(k+1) = 0$
$NF_{r4}(k-1) = 3$	$NF_{r4}(k+1) = 1$
	$NF_{r5}(k+1) = 4$

Correlation complexity metric = 36 / 17
= 2.118

4   Performance Assessment

In contrast to complexity assessment which determines the difficulty of the input scenario, performance assessment is concerned with the quality of the output track picture. However, it is also possible to separate performance assessment into two main parts. Quantity performance describes the quality of the global track picture in terms of the number of reconstructed tracks without taking the integrity of the tracks into account. Accuracy of the reconstructed tracks is then measured by the quality performance metric.

4.1 Quantity Performance

The quantity performance of a DF system depends on the number of tracks seen in the global track picture. Given the targets maneuvering in the scenario, one method of gauging the output track picture is simply to count the number of tracks which are detected by the DF system as originating from targets.

4.1.1   Global Track Picture

On a macroscopic scale, an approximate estimate of the reconstructed track picture quality can be made by comparing the number of reconstructed tracks in the global track picture with the number of true tracks made by the targets in the scenario. In an ideal track picture reconstruction, the number of tracks produced by the DF system would correspond to the number of targets. In truth, we may see the occurrence of false tracks or deficiencies of missing tracks. False tracks are generated when the sensors provide reports from non-target objects. On the other hand, missing tracks result when sensors fail to detect true targets or when the DF system fails to ascertain the presence of a target due to errors in the DF process.

The extent to which a reconstructed global track picture mirrors the true scenario can thus be calculated by adding up the number of false and missing tracks as follows:

$$\text{global track picture metric} = \text{no. of false tracks} + \text{no. of missing tracks} \qquad (14)$$

In this method of calculating the global track picture metric, every false and missing track is assumed to be of equal significance. However, this can be modified in the

case where greater importance has to be placed on certain tracks. Each track could be multiplied by a weight factor which relates the degree of importance of track. The global track picture metric is thus given by:

$$\text{global track picture metric} = \sum_{f \in \text{false track}} \left(\text{false track}_f \times \text{weight factor}_f \right) + \sum_{m \in \text{missing track}} \left(\text{missing track}_m \times \text{weight factor}_m \right) \quad (15)$$

4.2 Quality Performance

Details and errors about each reconstructed track are captured in this metric. In this case, we consider both the updates used to reconstruct target tracks and the final target trajectory mapped out by the DF system.

4.2.1 Correlation Accuracy

Due to the myriad of possible movements of targets, as well as the inherent impossibility of perfectly correlating every received report, data association errors may be committed by the DF system. Correlation accuracy can thus be measured by the track update error percentage, which is the percentage of wrongly associated reports to a given track.

Of all the reports used to update the track of a certain target, we calculate the percentage of false reports which, in actual fact, did not originate from the target. Hence, track update error percentage tells us the impurity of a target track. An example is shown in Figure 2.

$$\text{track update error percentage}_j = \frac{\text{no. of false reports}}{\text{total no. of reports related to target } j} \quad (16)$$
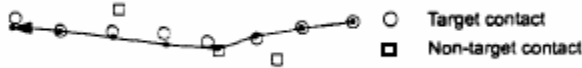

Figure 2: Track update error percentage of 1/8 (0.125) due to the association of one false target contact [3].

To calculate the correlation accuracy metric, we then sum up the track update error percentages from all the targets.

$$\text{correlation accuracy metric} = \sum_{j \in \text{targets}} \text{track update error percentage}_j \quad (17)$$

It is important to note that it would be difficult to calculate the correlation accuracy metric if the scenario that was reconstructed was not a simulated scenario but a real experiment. This is due to the unavailability of totally accurate ground truth in actual operations, leading to difficulties in determining false reports. This problem can be overcome by using simulated scenarios in which the actual ground truth is known perfectly.

4.2.2 Individual Track Quality

Ideally, sensor reports on the targets in the scenario would be received would be infinitely fast and totally accurate. However, this ideal DF system is clearly impossible to implement. As such, even though the DF system may reconstruct the track of a target, it is extremely unlikely to be able to do so perfectly unless the target was stationary or traveling in a very simple manner such as in a straight line or purely circular motion.

Individual track quality metric thus attempts to determine how closely the reconstructed track models the actual path of the target.

To estimate the position of a target at a certain time instant, a DF system first predicts the target position from the positional and kinematic histories of the target. Following that, the DF system fuses the target updates received from the sensors with its own prediction to yield an estimated target position. This represents the DF system's best evaluation of a target's position and may differ from the true target position. The positional error ε_i between the estimated and true target positions can thus be calculated. Figure 3 shows the estimated track positions and true positions of a target for several time instants.

$$\varepsilon_i = | z_{i,\text{estimated}} - z_{i,\text{true}} | \quad (18)$$

where $z_{i,\text{estimated}}$ and $z_{i,\text{true}}$ are the estimated and true target positions of the target at time i.
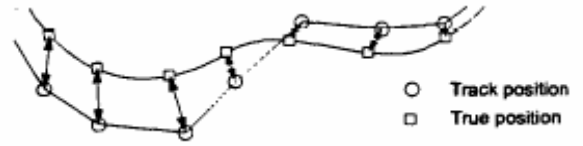

Figure 3: True and estimated positions of a target at several time samples [3]

The root mean square (rms) positional error is then obtained. This rms error represents the average deviation of the reconstructed track from the true track of a target.

$$\text{rms } \varepsilon = \sqrt{\frac{\sum_{i=1}^{n} \varepsilon_i^2}{n}} \quad (19)$$

where n is total number of track updates.

A track error threshold is also defined by the user of the evaluation tool. This value represents the maximum tolerable error of a reconstructed track. The rms error of a track is then scored on the following scale.

$$\text{track error score} = \begin{cases} 1 & \text{if rms } \varepsilon \geq \text{track error threshold} \\ \dfrac{\text{rms } \varepsilon}{\text{track error threshold}} & \text{if rms } \varepsilon < \text{track error threshold} \end{cases} \quad (20)$$

Finally, the track quality metric is obtained by summing up all the track error scores from all the reconstructed tracks.

$$\text{track quality metric} = \sum \text{track error score} \quad (21)$$

5 Test Scenarios And Results

Having designed the DF system evaluation methodology and metrics, tests were carried out to assess the evaluation tool. These tests were carried out with the use of a scenario generator which models the movements of platforms and targets. Also, it provides sensor reports and fuses the information to obtain a global track picture. However, the distinct advantage of using this scenario generator is that knowledge of actual ground truth is available and we know which target every update originates from. This is important in judging the accuracy of the data association process.

To test the effectiveness of the DF system evaluation tool, several conditions which impact the performance of the fusion system are identified. These conditions and

their respective impacts on DF system performance are listed in Table 2. Following which, scenarios testing each condition are generated and assessed by the evaluation tool. Finally, the results are compared to those of a default scenario so ascertain if the conclusions match our hypotheses.

Conditions to vary for analysis	Expected impact on data fusion performance
User-defined Inputs	Decreased performance with more stringent distance and error thresholds.
Target Separation	Decreased performance with decreased target separation distance.
Target Maneuverability	Decreased performance with increasing maneuverability due to sudden changes in velocity and direction.
Number of Targets	Decreased performance due to increased number of updates and data association errors.

Table 2: Conditions affecting DF system performance and the corresponding expected impacts.

Figure 4 depicts the computer-generated scenario A with 4 platforms (P1, P2, P3 and P4) and 5 targets (T1, T2,…, T5). Scenario A serves as the default scenario and is the basis of comparison with other scenarios. A close up of part of one of the target trajectories is shown in Figure 5.
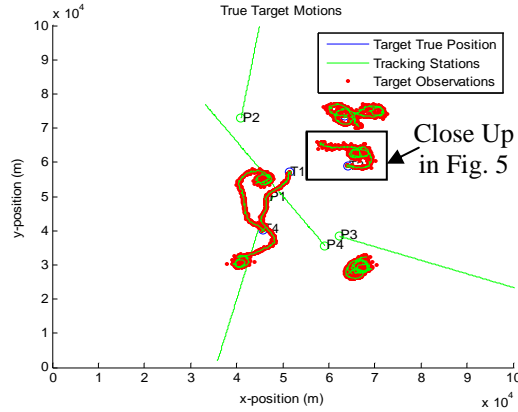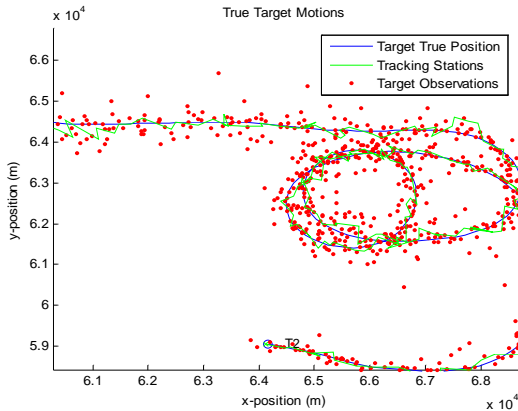

Figure 4: Default scenario A.

Figure 5: Close-up of part of a target trajectory

As can be seen in figure 5, the smooth line represents the actual path of the target, while the jagged line with numerous kinks represents the reconstructed track of the target after DF. The target updates provided by the sensors on the platforms are shown by dots.

To determine the impact of the conditions discussed in Table 2, scenarios B, C and D were generated. They are depicted in Figures 6, 7 and 8 respectively. All the scenarios were then evaluated and the resulting assessment indexes compared with those of default scenario A. Table 3 summarizes the various scenario settings.
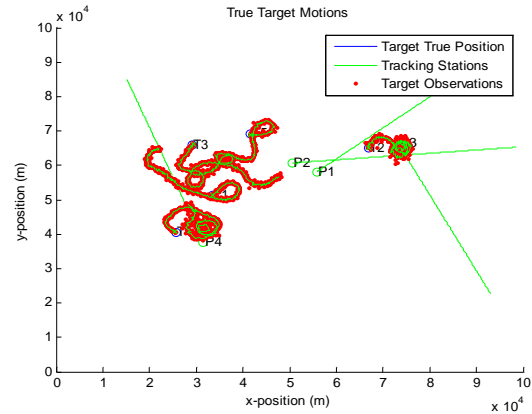

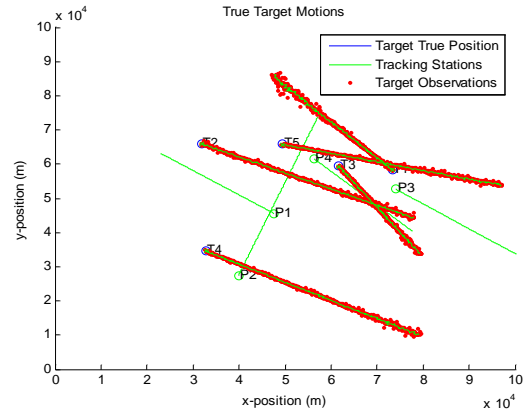Figure 6: Scenario B testing the effects of increased target separation.

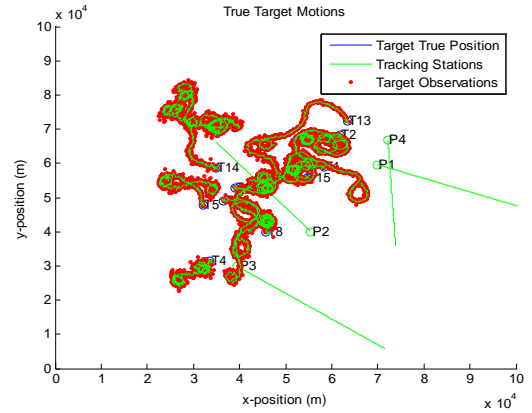Figure 7: Scenario C testing the effects of decreased target maneuverability.

Figure 8: Scenario D testing the effects of increased number of targets.

Scenario No.	A	A	B	C	D
No. of targets	5	5	5	5	15
Average target separation (m)	35000	35000	20000	35000	35000

Inter-target distance threshold (m)	20000	5000	20000	20000	20000
Inter-sensor distance threshold (m)	20000	5000	20000	20000	20000
Track-error threshold (m)	300	100	300	300	300
Maximum target turn rate (rad/s)	0.005	0.005	0.005	0.000	0.005
Quantity assessment index	0.2222	0.2222	0.2222	0.2222	0.1513
Quality assessment index	0.9360	1.7540	0.8327	0.8651	1.0331

Table 3: Table of scenario settings and evaluation results.

Scenario A is first evaluated using the settings provided in Table 3. We obtain a quantity assessment index of 0.2222 and a quality assessment index of 0.9360. For every scenario, 2 false tracks were assumed to be present. In actual fact, there are no false tracks as the number of reconstructed tracks corresponds perfectly to the number of targets. However, for purposes of comparing the quantity assessment index of different scenarios, an arbitrary number of 2 is chosen.

To determine the impact of user-define settings, scenario A is evaluated again using smaller distance and error thresholds. With a smaller margin of tolerable error, we would expect a poorer evaluation of the global track picture since it is graded against more stringent standards. This is clearly evident from the quality assessment index of 1.7540 obtained using the smaller user-defined thresholds.

As can be seen from Figure 4 and 6, the targets in scenario B are more densely cluttered together, contributing to decreased target separation distance. Since the same scenario generator with the same DF capability was used to reconstruct the global track picture, the quality assessment index for Scenario B would be smaller since the complexity of the input scenario was more difficult. The quality assessment index of 0.8327 compared with 0.9360 confirms this hypothesis.

Scenario C was generated to gauge the effect of target maneuverability on the quality of the output. This was achieved by changing the target turn rates to 0 rad/s so that the targets would move only in straight paths. Since the assessment index comprises both the performance of the DF system and the complexity of the input scenario, one may expect the less complex scenario C to result in a larger quality assessment index. However, in actual fact, we see a smaller index of 0.8651 compared to 0.9360 of scenario A. This is because the increased accuracy of the reconstructed tracks due to the reduced target maneuverability far outweighs the contribution of the simpler scenario.

Lastly, scenario D highlights the effect of increasing numbers of targets. From Table 3, we see that this leads to a smaller quantity assessment index as well as a larger quality assessment index. Since we have 15 targets and 4 platforms in scenario D, the fact that only 2 false tracks were created indicates better performance when compared to scenario A in which 2 false tracks were created when there were only 5 targets and 4 platforms. However, due to the increased number track reconstruction errors, a larger quality assessment index of 1.0331 is obtained. Thus, the assessment indexes calculated provide a good estimate of the performance of the DF system.

6 Conclusions

From tests conducted above, we have demonstrated that the proposed method of evaluating DF systems enables us to measure the effectiveness of a DF system in the light of scenario complexity. As such, this allows us to use a wide spectrum of scenarios of varying complexity as inputs to a particular fusion system and yet be able to interpret the assessment metrics in relation to each other. Without considering the complexity of the scenarios, the assessment indexes of various scenarios could not be compared and there would difficulty in estimating the appropriate level of performance for any given scenario. Thus, we can use this assessment methodology to highlight particular deficiencies in the DF system in order to optimize system performance.

References

[1] G.W. Ng, *Intelligent Systems – Fusion, Tracking and Control*, Research Studies Press Ltd, Hertfordshire, UK, 2003.

[2] M.O. Hofmann, and S.M. Jameson, *Complexity and Performance Assessment for Data Fusion Systems*, Proc. 1998 IRIS National Symposium on Sensor and Data Fusion, Mar 1998.

[3] A. Theil, L.J.H.M. Kester, and E. Bosse, *On Measures of Performance to Assess Sensor Fusion Effectiveness*, Fusion 2000 - Proc. 3rd International Conference on Information Fusion, vol. II, Jul 2000.

[4] S. Rawat, *A Frame Work For Performance Evaluation Of Multi Target Tracking Systems*, Final Technical Report For AFOSR and AFFTC, Aug 2002.

[5] S. Rawat, J. Llinas and C.L. Bowman, *Design Of A Performance Evaluation Methodology For Data Fusion-based Multiple Target Tracking Systems: Part 2: Analysis Methodologies*, Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2003, Proc. SPIE, vol. 5099, pp. 139-151, 2003.

[6] S. Bhattacharya and R. Appavu Raj, *Performance Evaluation Of Multi-sensor Data Fusion Technique For Test Range Application*, Sadhana 2004, 29(2):237--247, April 2004.

[7] B.N. Suresh and K. Sivan, *Performance Evaluation Of Multi-sensor Data Fusion Systems In Launch Vehicles*, Sadhana 2004, 29(2):175--180-247, April 2004.

[8] P.B.W. Schwering, B.A. Baertlein, S.P. van den Broek and F. Cremer, *Evaluation Methodologies For Comparison Of Fusion Algorithms In Land Mine Detection*, Detection and Remediation Technologies for Mines and Minelike Targets VII, Proc. SPIE, vol. 4742, pp. 847-856, 2002.

[9] O.E. Drummond and R.L. Rothrock, *Performance Metrics For Multiple Target Multi Sensor Tracking*, Signal and Data Processing Of Small Targets 2000, Proc. SPIE, vol. 4048, pp. 521-531, 2000.

[10] O.E. Drummond, *Methodologies For Performance Evaluation Of Multiple Target Multi Sensor Tracking*, Signal and Data Processing Of Small Targets 1999, Proc. SPIE, vol. 3908, pp. 355-369, 1999.