# Merging probabilistic data of multiple targets detected by multiple sensors


James Spillings [1]

*General Dynamics UK Ltd, Bryn Brithdir, Oakdale Business Park, Oakdale, South Wales, NP12 4AA, United Kingdom*

*and*

Antonios Tsourdos[2], Peter Silson[3] and Brian White[4]

*Department of Informatics & Sensors, Cranfield Defence & Security, Cranfield University, Defence Academy of the United Kingdom, Shrivenham, Swindon SN6 8LA, United Kingdom*



**The aim of this work is to present an extension to current data fusion techniques and associated results for the implementation of a collaborative multi-platform, multi-target detection system. A multi-sighting data fusion algorithm has been simulated. The ground-based platforms have been assumed autonomous, with fully operational guidance systems. The attached sensors have associated errors that are controlled through the simulation. The target sightings and errors are translated into estimates with associated covariances in both the major and minor axes represented by 3-dimensional Gaussian distributions. Data merging is performed in two stages using the Jointly Gaussian Probability Density Function (JGPDF) with global alignment and minimal acceptable distance calculations. Empirically data highlights that, when using this approach a better estimate of the target's location can be obtained when more observations are made, along with distinguishing between multiple targets.**


## I. Introduction

The his paper aims to explore the issues surrounding localising detected targets within a know region from data gathered by multiple platforms. The problem is addressed by using a platform, having a known map to perform self-localisation, to detect and localise a target with respect to itself. The technique of interest for localisation is Simultaneous Localisation and Map building (SLAM) while research will be conducted into data fusion techniques for merging of the resulting target acquisition data. The main objectives of this work are to:

- gain a theoretical understanding of SLAM and the surrounding issues,
- compare and contrast the estimation techniques employed within SLAM,
- perform a short study into appropriate sensor suites and fusion techniques and
- develop a practically feasible solution to the described SLAM problem.

## II. System Design & Considerations

We will first introduce the various aspects of sensor system design, the surrounding issues and the considerations needed when looking at such systems. The initial focus will be on the employment of sensors and why they play an important part in modern warfare. Following on from that the focus will be towards the sensor system design issues and considerations. This section endeavours to give the reader a short precise on some of the main issues

[1] Engineer, Department Name, General Dynamics UK Ltd.
[2] Head of Autonomous Systems, Department of Informatics & Sensors, AIAA Member.
[3] Assistant Professor, Department of Informatics & Sensors.
[4] Professor Emeritus, Department of Informatics & Sensors, AIAA Member..

surrounding the design of systems such as the one being designed here. The aim is to then lead this section into the next section which is the start of the design phase.

### A. Why Use Sensors?

In today's world sensors are used in abundance for a variety of tasks from asset tracking to modern warfare. Sensors are generally designed to operate in specific manner to satisfy a specific requirement, such as night vision goggles. These have specific detectors 'tuned' to be sensitive to certain frequencies. The goggles can only detect these frequencies unless they have special filters to enable them to switch between detector cells. Producing sensors that have very specific parameters allow them to perform in the most optimal manner. Again various limitations apply, especially where the environment is concerned. Many of the various constraints and limitations placed on sensors can seriously affect the performance. Along side this many factors such as cost can affect the systems employing these sensors. For example a laser has very good range and azimuth resolution, so why not always use a laser? A laser is extremely expensive and energy-inefficient requiring a large amount of power. As well as this as the beam travels through the atmosphere, the atmosphere heats up and can cause the beam to move or wobble. Admittedly this is only a major concern when dealing with very long-range applications but is still a consideration that has to be taken into account. Radar system resolutions can change dependant on the application. Very long range Radar systems can have resolutions exceeding 100 meters and the typical police Radar speed guns can have resolutions in the 10's of centimetres range. The trade off between range and resolution becomes obvious. It is due to the above reasons that many systems have started to employ multiple different sensor types thus allowing a system to be multi-spectral. The primary reason for systems utilising sensors in different spectrum is to allow objects be detected in the best possible manner dependant on the current conditions. Radar systems are highly effective against metal and similar material based objects. These will reflect the majority of the transmitted signal giving good indication of the target position, speed and other attributes. Infrared systems work best when heat is being emitted from the object of interest. If the object is a tank then both infrared and Radar can be used. However should there be a lot of mist then the infrared system would not be very useful. This introduces the concept of counter measures and the various options and workarounds to such counter measures. The use of different sensors within one system allows greater spectral coverage meaning objects will find it harder to become disguised. When it is possible to use all the sensors the system will be able to use the estimates to refine the data received. This can help to obtain more accurate information on pose and possible target type. This project is concerned with the use of the information obtained about targets from multiple sensor types based on disparate sensor platforms.

### B. System Design Issues

There are many MoD projects coming to the fore concerning information gathering and exploitation using sensor based systems. Programmes such as the Future Rapid Effects System (FRES), the Airborne STand-Off Radar (ASTOR), Watchkeeper and many others aim to gather information from sensors, either collaborative or disparate, fuse or associate the information with the end aim of producing increased knowledge on the environment. As an example, FRES is aimed at providing rapidly deployable medium weight armoured vehicles. A requirement of the system is to be able to observe the battlefield through the use of sensors, transmit the gathered data and then perform some data fusion to aid in knowledge improvement to support force movements and possible future attacks. It is envisaged that many of these vehicles will operate together although can be assumed disparate. In the case of ASTOR and Watchkeeper it is envisaged that the sensor data from the systems will be used in conjunction as an enhancement to the separate forms of data. It is evident that the data from the two systems is not only disparate data but potentially in varying formats; this brings its own challenges. From this it becomes clear that there is potentially a large amount of data becoming available from various sources with an aspiration and a need to do something with the data to make it 'better'. Data obtained from the sensing platforms could have been obtained from a variety of sensors at differing times subsequently introducing a varied amount of errors. The main issues that arise from sensor based systems is the choice of sensor. The customer can have aspirations for systems to perform many tasks that are simply not viable. Sensors are primarily required to detect range, bearing, elevation, target speed and where possible target type. Other information can be derived if required. From these requirements and the scenario the sensors are to employed within will depict the type of sensor and its attributes. Where the complete system is concerned the main points for concern are:

- Sensor resolution
- Meteorological conditions
- Operational requirements
- Collated data transmission requirements (data, bandwidth, timeliness of data)

- Power requirements
- Sensor size and weight and
- System payload.

The following sub-sections aim to describe some of these issues with attention paid to possible trade-off situations; examples are given where necessary.

*1. Resolution*

A sensors resolution is the smallest change it can detect in the quantity that it is measuring. When considering a Radar system a signal is transmitted and a subsequent reflection received. The received signal can provide data on the sited object. In a pulsed Radar system the pulse length depicts the resolution. The lower the resolution the larger the area per single capture, hence why fighter jets fly in close formation. Resolution, distance and power are three main components that need to be traded to get a system to perform as required. The more power the longer the range of the system; however the longer range means a larger pulse and subsequently a lower resolution.The diagram below, Figure II-1, shows the detected target is at location (x,y) however due to the resolution of the sensor the target could feasibly be at any location within the bounded oval. The better the resolution in the various directions the smaller the oval becomes, reducing the possible location options. Depending on the sensor being employed the resolutions will vary. Some sensors may give the user an idea of the target position however the errors may be considered too high for the target to be of use ate the given range. If the target is a landmark that is being used to provide localisation data then the resolution of the sensor will need to be much better than say a sensor designed to track targets beyond the horizon.

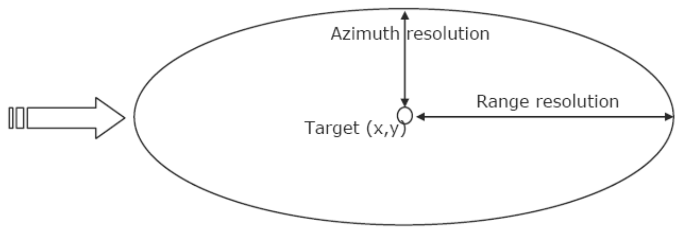


**Figure II-1 Range & Azimuth resolution**

*2. Data & Bandwidth*

Bandwidth and data capture are closely linked. Any system that is concerned with data capture must be able to either store and deal with the data entering the system or have sufficient bandwidth to transmit the data to the desired location in a timely manner. If the data capture and manipulation algorithms are collocated then there are no concerns for bandwidth however if the data is required on a disparate platform so that fusion or association can take place then bandwidth becomes a limiting factor. If we look at current Closed Circuit Television (CCTV) within city centres there is a vast amount of information being collated. The average 3Mpixel digital camera might capture 6 frames per second creating a frame size of ~700Kb. This system would create in the order of 4.2Mb of imagery every second that would need to be transmitted to the CCTV control centre. It can be seen that such a system would require large amounts of bandwidth in order to transmit the data in real-time. This type of system has the benefits of being able to utilise a fixed wired infrastructure where bandwidth limitations are not considered a huge issue. In contrast to the fixed system where bandwidth issues are of low priority wireless systems, similar to that employed in military environments show that bandwidth is potentially the largest affecting factor. The main issue arises from the lack of line-of-sight communications and the subsequent need to use the High Frequency (HF) spectrum, although the Very and Ultra High Frequency (V\UHF) spectrums are also employed where possible. The through-put of the system can range from the very low bits per second range up to ~300Kb\s for UHF, however the UHF radios have a low range and are generally used to provide a back bone to the main VHF and HF radios; bandwidth at this level are around the 16Kb mark for VHF and 300b mark for HF. When designing a system where large volumes of data are being captured the designer has three main options available:

- To stream the data as it is captured. This ensures that all the captured data can be used for analysis,
- To perform some analysis on the data in an attempt to remove redundant data and only transmit that which is felt necessary,
- Or to analyse the data in full and only transmit the results.

All options have their merits and the final decision is ultimately arrived through complete analysis of the key system and user requirements.

*3. Power Consumption, Size & Weight*

Three influencing factors of any sensor-based system design are the size, weight and power consumption of the sensors. Modern battlefields have platforms ranging from Un-manned Air Vehicles (UAVs) with small payloads right up to tanks with much greater capabilities for housing sensors requiring large amounts of power and weight. The task the sensor is going to be employed for will affect the aspects of the sensor. For a large long range radar system, for instance, there is a requirement for large amounts of power and also a large receiver antenna. This type of sensor would not be installed on a UAV due to power and payload limitations. Looking at small Infra-red (IR) sensors used to detect a target require small amounts of power and have a small footprint, hence they are ideal for use in missiles. Modern advances in battery technologies have allowed systems to operate longer. Mobile phones have much greater standby times due to Lithium Ion technology. Component sizes are also reducing and with this generally a weight decreases occurs. Many technological advances are useful in modern civilian life however are more important within military systems. Such systems are required to be ruggedised so they can withstand high velocity impacts and shock treatment and must continue to operate. The technological advancements in component sizes can allow these military modules to become smaller and bear less of an effect on the personnel carrying them. All of the above areas need to be taken into consideration by the system design engineer. In this project the requirements are for a group of platforms to be able to scan a region of interest in a given pattern recording all target detections so to enable merging of the collated data with the other platforms. The following section introduces the problem in more depth with a look at the requirements of the complete system, the selection of sensors and algorithms and finally running various tests on the completed system in a simulated environment.

## III. The Problem Space & Solution

Previous sections presented a variety of techniques and technologies surrounding the sensor and data fusion domain. They also presented the high-level system design work required when implementing such a system and the considerations that attention must be paid to in order to ensure the system can handle all the eventualities it is required to. The following sub sections present the problem space the project is in. It then goes on to provide a description of the overall solution to the problem described. All of the testing and results will be presented in the following section; the application code will be available in the appendices.

### C. The Problem Space

This sub section focuses on the problem space from a high-level perspective focussing down onto the area this project aims to offer a solution to. The section looks at the motivation behind the solution and the individual problem at hand. The final subsection will consider techniques employed by other systems, as discussed in previous sections, and highlights the contributions made to the algorithm of choice in an attempt to improve the overall algorithm.

#### *4. Motivation*

Recent events have highlighted the ever-changing face of modern warfare and the need for strategical urban warfare techniques to be developed and improved upon. The increase in asymmetric warfare shows that the current blue forces have particular weaknesses exploited by the enemy, due to an increased knowledge of the urban battlespace, putting lives in greater danger. As such the requirement for 'better' information is growing and this information is much harder to obtain. The battles of WWI and WWII were fought on open planes where visibility was good and stand-off assets could be employed. In scenarios such as Iraq, Kuwait and to some extent Afghanistan the warfare is very different where visibility of targets is poor due to built-up, urban areas. As such the requirement to get into the areas to find the targets is high and comes with many associated risks, namely attack from snipers and non-identifiable red force personnel. In order to make sense of the situation 'disposable' assets are employed. Assets such as UAVs and UGVs carrying sensors enabling them to detect, track, identify and sometimes destroy targets. Developing an SA picture with high confidence is now becoming a high priority, as can be seen by the various defence agencies putting large amounts of capitol into the intelligence, surveillance and reconnaissance domains. Although the capability to detect, identify, track, and destroy targets exists, reliably locating targets using multiple assets still has problems. As more unmanned technologies are employed the requirement for them to interact passing data on things such as maps, locations and targets is increasing. This data needs to be merged in order to provide more accurate data so more informed decisions can be made.

#### *5. The Problem*

Considering multiple sightings of multiple targets is an error filled process that leads to difficult localisation of the targets being interrogated. This type of problem is common place in the battle space of today where disparate sensors are used to survey an area of interest potentially detecting many targets. As such, many sightings of many targets can lead to potential false targets being created and confusion thus arises. The underlying issue when attempting to resolve targets into one or more targets is caused when a single sensors' resolution is of a subsequently low quality that more than one target can occupy the sensed area. As well as this all sensors have varying intrinsic errors in both azimuth and range leading to the inevitable incorrect positioning of a target or targets. These two issues alone cause a level confusion with targeting. This project is looking at the problem of a multiple platforms sighting multiple targets in an area of interest. The platforms will not have any knowledge of the targets, neither positions nor number, and will have to locate the targets to the best of their ability based on the information available. The aspiration is that by fusing the sightings of all the targets then a more confident red force picture can be developed.

*6. Focus and Contribution*

Many systems that are currently employed to perform target tracking, data fusion and other such tasks generally require either a few sightings of a target in order to confidently identify that it is one target and subsequently initialise a track of the target or for there to be one target, such as in robot football. The last example was mentioned as Stroupe et al [46] employed a technique to identify a target from two disparate platforms and provide a confident estimate of the target location. The complete system was used to identify a ball by two collaborating robots. The algorithm here aims to extend on this system, originally introduced by Duffin[5]. By extending the algorithm it is hoped that the system will be able to, confidently, locate multiple targets using readings from multiple sightings from a platform. Multiple disparate platforms should then be able to supply their data and provide an even more confident estimate of the target locations. Although this work primarily focuses on a mobile platform, static target scenario the aim to is to create an algorithm that can be either extended itself to handle a completely dynamic environment and also accommodate tracking where required or to be used alongside other algorithms to create a best of breed approach to the overall sensor to shooter problem.

## D. The solution

This subsection will first look at the platforms and sensors to be employed including the use of the sensors and the movements of the platforms. The main part of the subsection will focus on the algorithm being used to merge the data.

*7. Platforms and Sensors*

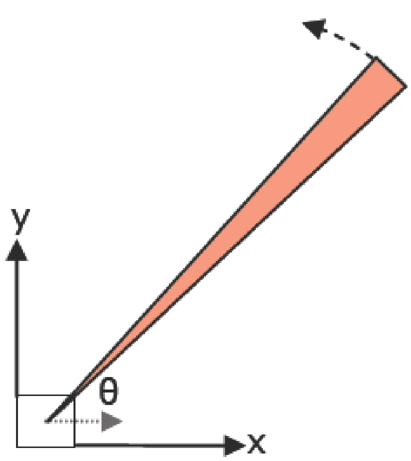

**Figure III-1 Platform sensor sweep**

The aim, while developing the solution, is to consider the algorithm that will be used, the platforms that will be expected to be used in the problem and subsequently the sensors that can be deployed with those platforms. Having analysed the issues surrounding the choice of platforms and sensors in previous sections the main requirement here is for a land-based platform that has a limited payload. As such a UGV with the capabilities on board to freely navigate a structured environment has been chosen. The environment will be something akin to an American city, based on a grid road system and the targets will be different to that of the everyday expected environment. This will allow the main focus of the work to be towards the merging algorithm that will be used. The sensor suite, for ease within the early stages will be based on a laser; this provides a single beam with a generous amount of accuracy. As will be seen, the actual sensor suite and accuracy of the sensors does not affect the overall implementation of the algorithm. The platforms are to be used to survey a given area of interest (AOI) and to detect various unknown objects. At each step (one step is considered to be a point in time when a detection is made) the platforms will merge the individual results with all of the individual merged results being merged at the end to create a set of results equal to the number of targets in the scenario. It must be noted that the merging of the results in this manner can occur at any point in the calculation; the result will remain the same. The final output of the association work is expected to provide a more accurate location of each target. As the platform will only have one sensor type on board the sensor will perform a circular sweep of the AOI recording the angle of detection and the range of the object detected, see Figure III-1. The system will then store the angle of the sensor and the range reading at the moment an object is detected. This along with the platforms location at detection time will provide the relevant data for association to occur. The platform will utilise a global positioning system (GPS) module along with an inertial navigational system (INS) module to maintain an accurate

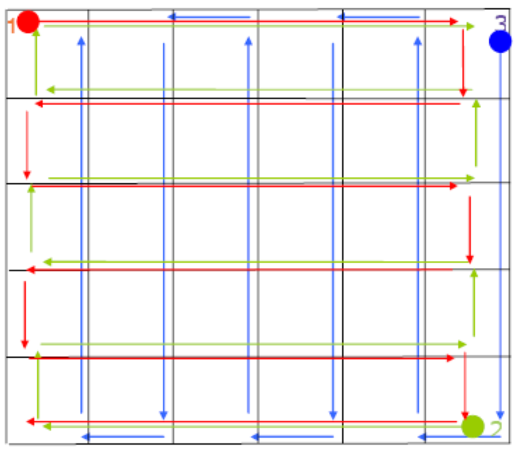


**Figure III-2 Platform pre-planned routes**

real-time fix of its position. As well this each platform will have a pre-planned route to follow allowing it to comb the desired area of interest (AOI). In order maintain a level of realism the scenario will use three platforms. Each platform will follow a different route in order to obtain detections of targets from various angles. The routes, as can be seen in Figure III-2, provide comprehensive cover of the AOI and should allow a rich number of results to be obtained from the sensors. As can be seen in the diagram in Figure III-2, the platforms will start at different locations (the start points are identified by the coloured dots, the corresponding end point is the same coloured cross). The varying start locations should be different and varied where possible to ensure that the targets or red force components do not suspect a collaborative mission is occurring, amongst other reasons.

*8. The Algorithm*

Previous sections have discussed the various methods used when performing data association and merging. The technique being employed here uses the Jointly Gaussian Probability Density Function (JGPDF); this is concerned with the probability of a target being where the sensor says it is. This technique was first introduced by Duffin[5] and is akin to the techniques employed in Kalman Filter algorithms. The Gaussian distribution represents every location as a probability, assigning a probability that the target is in a specific location. As such a bell curve of the probabilities is created across the entire locale.

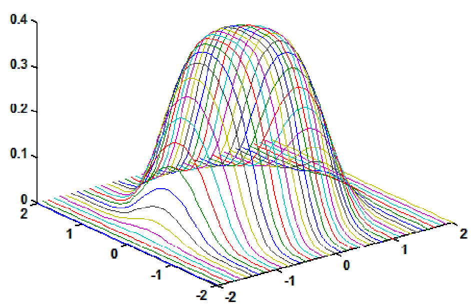


**Figure III-3 Probability Density Function**

As can be seen in the example above, **Error! No bookmark name given.**, the bell has an even distribution positioned about the location (0, 0). The curve very quickly tends to zero and as such can be more simply represented as a two-dimensional Gaussian distribution. The mean of the distribution represents the targets estimated location. The estimates of the uncertainty, in range and azimuth, in the observation are represented by the standard deviations (SD) along the major and minor axis. The range and azimuth resolutions of the sensors employed are known and as such are modifiable to enable the effects of degraded resolutions to be explored. The diagram below, Figure III-4, shows the parameters of the Gaussian distribution that are pertinent to the merging of multiple target sightings.

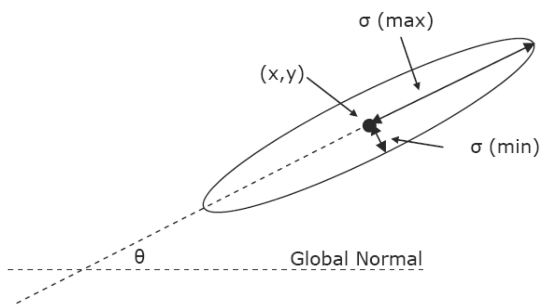


**Figure III-4 Two-dimensional Gaussian parameters**

The diagram above, Figure III-4, highlights that the target sighting can occur at any angle from a global normal; represented by θ. Due to the nature of the technique being used to merge the sightings, all of the parameters must be relative to a global reference frame, shown in the diagram as the global normal. As such the sightings must be rotated to this global normal. The overall algorithm is split into two sections. The first section concentrates on whether a merge should take place; the second section looks at the merge process. The following subsections look at each part individually and introduce the mathematics behind the algorithm.

*9. Deciding to merge*

There are systems that use the JGPDF[46] however it is always assumed that there is one target and the sightings are such that a merge can always be made. In this scenario there are no guarantees on the number of targets. Prior to a merge occurring there needs to be a process in place that can identify if the targets are potentially the same. The following diagrams describe the most likely situations that will be encountered and whether the merge process should then be started or not. The decision made as to whether to merge or not is a rule-based approach. The minimum required conditions for a merge to take place are:

IF <distance between targets> is less than <x>
AND <targetA> is within one SD of <targetB>
THEN merge.

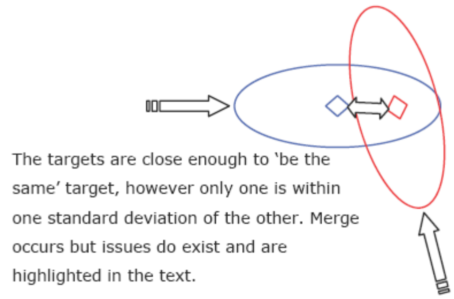


**Figure III-5 Minimum merge conditions**

The above pseudo code is shown visually in the diagram in Figure III-5. Although this provides the basis and other merge \ not merge scenarios stem from this one, it is not in itself definitive. This is caused by the variation in the intrinsic errors in different types of sensor and this issue is not in the scope of this project; it could be taken through into future work. The following diagram,
, is the most conclusive scenario that can occur to ensure a merge will occur. There is a very high level of confidence that the two targets are one target and the resulting merge should allow a target location with greater confidence to be produced. This is due to both targets being within one SD of each other and within the permitted range.

The scenario below, shows how one target can be within one SD of the other, however the reverse isn't true. This scenario may occur when a sensor with a markedly poorer overall resolution than another sensor sights the same target from a different angle; typically when the sensors encroach on being perpendicular.

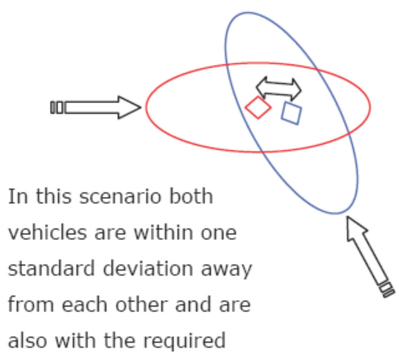


**Figure III-6 Ideal merge conditions**

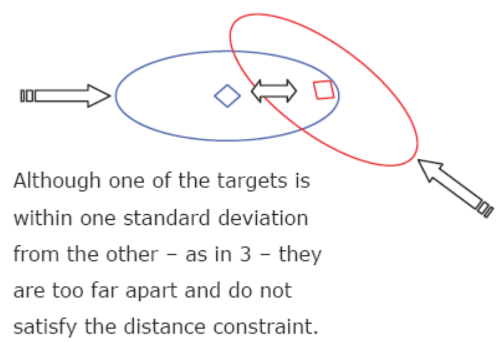


**Figure III-7 Merge distance criteria not met**

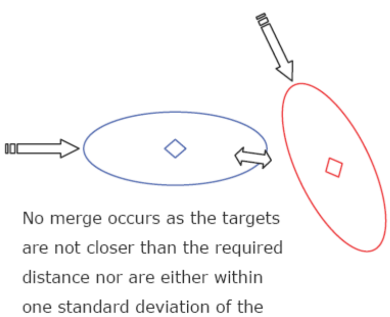


**Figure III-8 No conditions met so no merge occurs**

It is due to this type of scenario that the distance test has been introduced. The final scenario, Figure III-7, shows when two sightings are clearly of two separate targets that are in no way related. There will always be a debate as to what is an acceptable distance for two sightings to be apart before they can be considered separate targets. As such this distance can be altered to enable the emergent behaviours to be seen. For this project a reasonable distance based on the expected size of the target being approximately 5m x 2m (something the size of a tank) you can expect a distance of 30m to be considered reasonable. For the simulated work being done here the sensor errors and merge-ranges will be different though representative. Diagrammatically it is easy to see if a point is within one SD of another, however mathematically this is a little more complex. Ascertaining the distance between two points is achieved using basic algebra. In order to perform the SD check at least on target (the one the check will be performed on) needs to be rotated to a global normal. The same rotation can then be performed on all points that are part of the current SD check. Once this is done all objects will have a common reference. The diagram in Figure III-9 shows how the two reference frames differ. The x-axis is the global reference frame with $\theta^A$ and $\theta^B$ being the angles of rotation from the local reference frames to the global reference frame. The locations of targets A and B are denoted by (h,k) and (a,b) respectively.

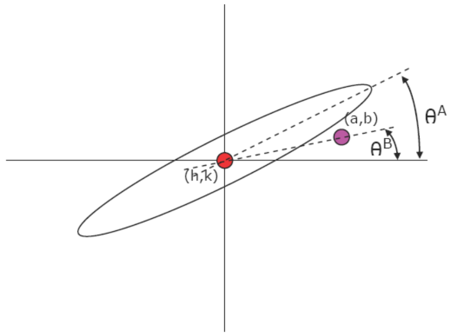


**Figure III-9 No common reference frame**

The first part of the algorithm is to ascertain the angle target B is from the global reference frame, angle $\theta^B$. This is done using the following formula

$$\cos(B) = (a - h) / r \qquad \text{(III.1)}$$

This allows the angle of rotation to be found. The second part of the algorithm is to utilise the above result and then perform the transform to calculate the new coordinates of the second target, (c,d) (as can be seen in Figure III-9)

$$c = h + \left(r \times \cos\left(\theta^B - \theta^A\right)\right) \qquad \text{(III.2)}$$

$$d = k + \left(r \times \sin\left(\theta^B - \theta^A\right)\right) \qquad \text{(III.3)}$$

Now the new location of the target is known and the both targets are in the same coordinate frame. The final part of this algorithm is to check if target B is within one SD of target A. If the result comes back as true then a merge can take place (assuming all other conditions are met), else the merge will not take place. The diagram above, Figure III-10, shows how the targets now sit with respect to each other. The final equation in the algorithm will show if target B is inside, outside or on the limit of one SD from target A.

$$\left(\Delta x^2 / \sigma_{maj}\right) + \left(\Delta y^2 / \sigma_{\min}\right) \qquad \text{(III.4)}$$

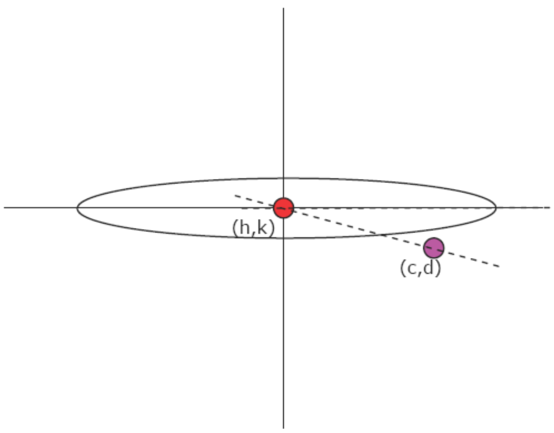


**Figure III-3 Targets transformed to global coordinate frame**

From the above equation the result will depict whether a merge will ensue:

Answer < 1 :: target B is within one SD of target A
Result → merge
Answer > 1 :: target B is not within one SD of target A
Result → no merge
Answer = 1 :: target B is on the circumference of the one SD limit of target A
Result → merge

In the latter scenario a merge will occur however further investigation would be required to see if there are any deeper implications to merging under these conditions. The second part of the overall algorithm is the merge section. The following subsection introduces the methodology behind the algorithm and introduces the mathematics involved in the merge process.

*10. The merge process*

The approach is based on the collation and merge of sightings from multiple disparate platforms in order to obtain a better estimate of the target location. As the sensing platforms plough the AOI they will gather more data on the targets and as such the final number sightings to merge should provide a substantially enhanced estimate. This process is a bi-variate case and uses the following equation as the basis:

$$f\left(\vec{x} \mid \vec{\mu}, C\right) = \frac{1}{2\pi\sqrt{|C_L|}} \exp\left(-0.5\left(\vec{x} - \vec{\mu}\right)^T C_L^{-1}\left(\vec{x} - \vec{\mu}\right)\right) \quad \text{(III.5)}$$

where:

$$\text{mean} = \mu = (x, y) \text{ and covariance} = C_L = \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}, \text{[Subscript L denotes sensor]} \quad \text{(III.6)}$$

The aim is to determine the mean, standard deviation and angle in order to provide a best estimate of the target position. The merging of the distributions is a relatively simple process. Smith and Cheeseman[47] introduced the methods being used here in their paper on representing and estimating special uncertainty. In the covariance, $C_L$, shown above, $\boldsymbol{\rho}$ is the correlation coefficient. As the two estimates are uncorrelated this value is zero and as such this results in the following diagonal covariance matrix.

$$C_L = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad \text{(III.7)}$$

As the covariance matrix is relative to the sensing platform, a rotation of the matrix to a global reference frame needs to take place. The rotation of X in equation 1.5 gives the following:

$$C = RC_L R^T \quad \text{(III.8)}$$

where:

$$R = \begin{pmatrix} \cos(\Theta) & \sin(\Theta) \\ -\sin(\Theta) & \cos(\Theta) \end{pmatrix} \quad \text{(III.9)}$$

Not forgetting the main principles behind the algorithm being used, namely the Kalman technique, as such the Kalman gain is the next part to be calculated. This is done as follows:

$$K = C_A * [C_A + C_B]^{-1} \quad \text{(III.10)}$$

This factor is used to calculate the resulting merge covariance where subscripts A and B represent the individual sensor. The merged covariance is calculated as follows:

$$C_M = C_A - K * C_A \quad \text{(III.11)}$$

Once the merged covariance is found the final estimated mean needs to be calculated. This is done using the following equation:

$$X_M = X_A + K * (X_B - X_A) \quad \text{(III.12)}$$

As mentioned before there are three main parts to calculate; the estimated mean, standard deviation and resulting angle. The latter is the last to be calculated and is done so using the following:

$$\Theta_M = 0.5\tan^{-1}\left(\frac{2B}{A-D}\right) \qquad \text{(III.13)}$$

In the above equation, A, B and D are components of the resulting covariance matrix; the upper left, upper right and lower right, respectively. The resulting standard deviation is calculated by rotating the resultant covariance matrix to align with the final merged major and minor axes:

$$C_L^{'} = R^T C_M R \qquad \text{(III.14)}$$

The standard deviations in the major and minor axes are taken from $C'_L$. The derivations of all the algorithms, in part, can be found either in [47]. There are also standard derivations done on many mathematic sites. All of the above equations have been implemented using MatLab.

## IV. Simulation and Results

A number of scenarios have been created to see how the algorithm performs. The scenarios start off with a simple single sensor, single target situation working up to more complex situations. The aim to see if the algorithm can cope with multiple sensor estimates of multiple targets. The following subsections describe the scenarios and provide the simulation results and analysis. All scenarios are run with under the same conditions:

- Targets remain stationary
- Target locations remain the same; (5,7), (2,-4) and (-6,3) for targets 1, 2 and 3 respectively
- Sensor platforms move 5 points then perform a scan
- Sensor platforms follow a predetermined route. The sensor platform route numbers can be seen in Figure III-2.
- All sensors will have the same standard deviation (2 and 1 in the major and minor axes respectively) unless otherwise stated.

Where any of the conditions change the changes will be mentioned though these will only be in the last scenarios. The graphs that show the final results only contain the individual sightings, the individual merges (where multiple sightings can be merged) and the final merge (of all the individual merges that have occurred). There will be no graphical representation of the covariances although metrics will be provided where pertinent.

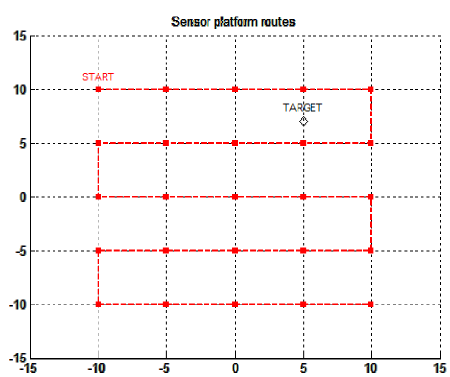


**Figure IV-1 Platform route**

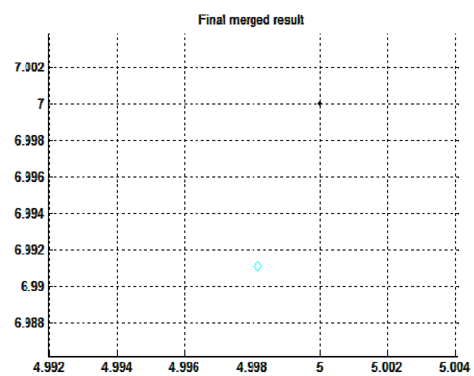


**Figure IV-2 Estimate**

### E. Scenario 1

*11. Test*

The first scenario looks at the most basic test case of one sensor and one target. The scenario aims to make sure the algorithm is working as expected. A sensor-based platform navigates a pre-determined route to find a target placed somewhere on that route. The graph above, Figure IV-1, shows the route the platform takes. It also shows where the target is relevant to the platform's route.

*12. Results*

The graph in Figure IV-2, shows the final estimate the platform has for the target; the cyan diamond. The estimated location was (5.0144, 6.9960) with a standard deviation of (0.4116, 0.2468), in the major and minor axes respectively. The estimate has the target well within one standard deviation of itself and as such is an acceptable estimate. The diagram in Figure IV-3 shows the individual estimates (in red), the merged estimates (yellow diamonds), the final estimate (cyan diamond) and the actual target position (black). As can be seen final estimate has produced an improvement on the individual estimates; in some cases a relatively vast improvement.

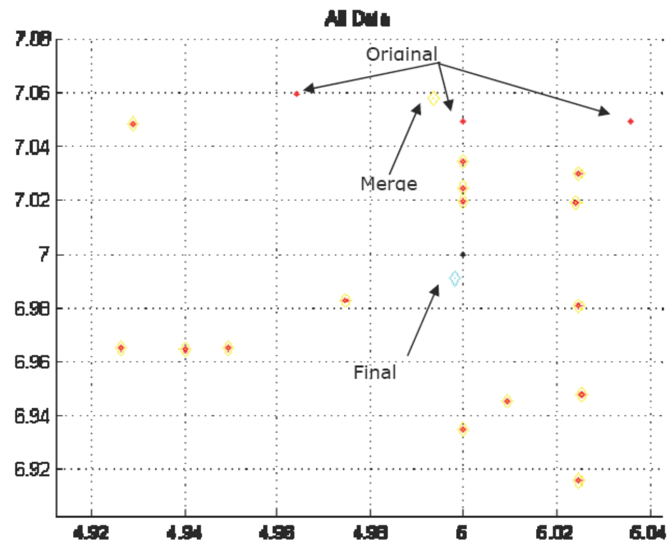


**Figure IV-3 Individual estimates and final estimate**

*13. Analysis*

This simple scenario indicates how the algorithm copes with one target in the area of interest. With no other distractions the algorithm performs as expected. Although there is a highly accurate estimate of the target's location, it would be expected that each sighting would only be 'merged with itself'. On close inspection it can be seen that some sightings were deemed to be close enough, within one standard deviation of each other, to be merged (three sightings identified at the top of the graph). The merge was probably aided by the major and minor covariance of the sightings. The major point to focus on in this example is that every point on the graph has contributed to the final merge as they were all deemed to be within the required distance. As such a very confident estimate has been provided.

## F. Scenario 2

*14. Test*

The next scenario looks to add more complexity to the area of interest by adding further targets. The aim of this scenario is to see how the algorithm handles multiple readings. The platform follows the same route as in.

*15. Results*

The diagram below, , shows all of the outputs from the simulation. At a glance it is clear to see that the algorithm has clusters of estimates around the target locations. This provides an initial indication that the algorithm has performed as expected though closer analysis of one target will need to be performed. The final estimates of the target locations are (-6.0102, 3.0122), (5.0068, 6.9904) and (1.9993, -4.0100) with associated standard deviations of (0.3067, 0.2291), (0.3150, 0.2418) and (0.3140, 0.2422) respectively.

To further analyse the results of the simulation the lower target (actual location (2, -4)) will be looked at in more detail. **Error! No bookmark name given.** shows all the estimates (red dots indicate sensor estimates and yellow diamonds indicate merged estimates). As can be seen there are a wide variety of estimates made during the simulation. The final merged estimate again is much closer to the actual target location, as expected.

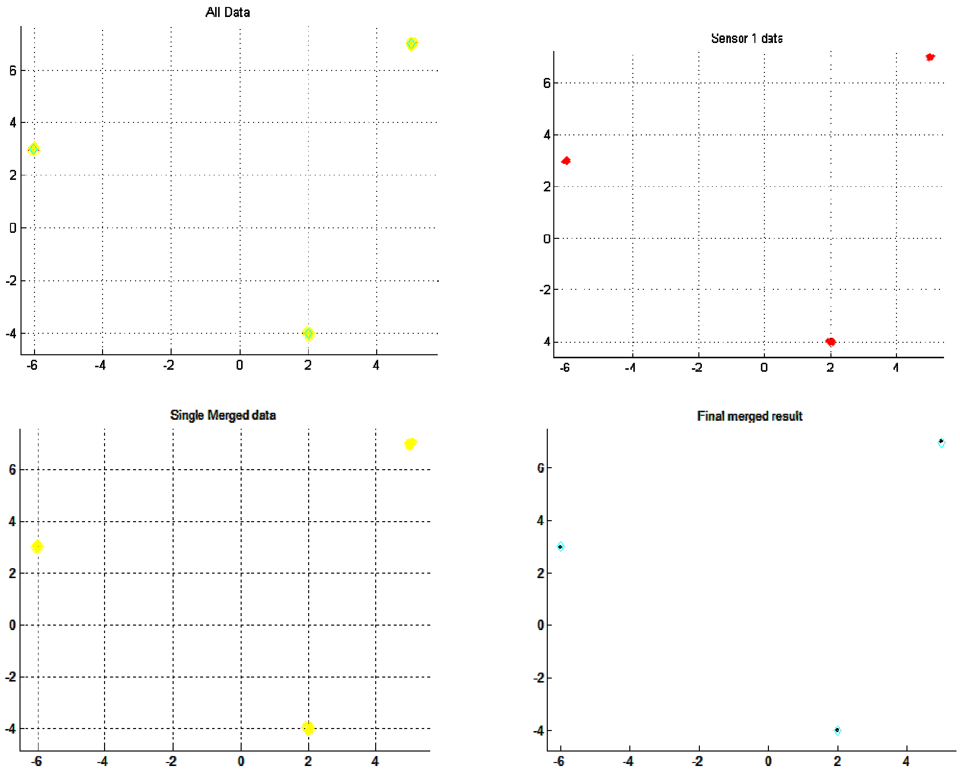


**Figure IV-4 High-level view of three targets, one sensor**

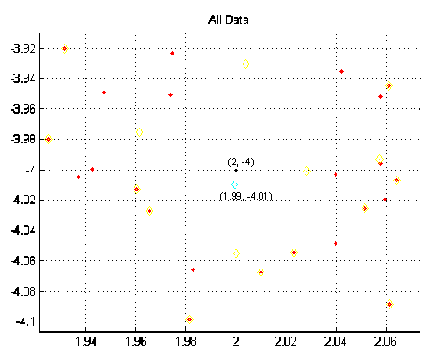


**Figure IV-5 All estimates**

*16. Analysis*

This scenario has proven that the algorithm is able to sense and resolve multiple targets in a given area of interest, whilst providing a sufficiently accurate estimate of each targets location. It can be seen in Figure IV-5 that the final estimate is closer to the target than any individual estimate. This helps to enforce the marked improvement the modification to the original algorithm has made. It must also be pointed out that more individual sightings have been merged allowing for greater confidence to be passed into the final merging process. The final estimates, although accurate do still have a large associated error in the major and minor axes. These errors are mainly due to only one sensor being employed; the errors of that sensor cannot be eliminated by any other sensor errors and as such are passed on directly to the resulting estimate.

## G. Scenario 3

*17. Test*

This scenario introduces more sensors to the area of interest. As with the first scenario this is to be made simple by only having one target. In real terms there will be three instances of the algorithm and as such there should be no reason for the algorithm to fail.

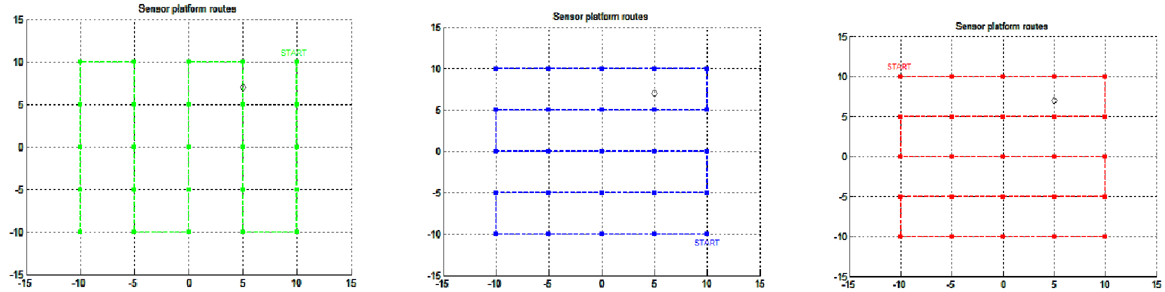


**Figure IV-6 Sensor-platform routes**

In  the routes of the platforms can be seen. The colours of the routes pertain to the colours shown in. As can be seen in this diagram all of the platforms returned good estimates of the targets location.

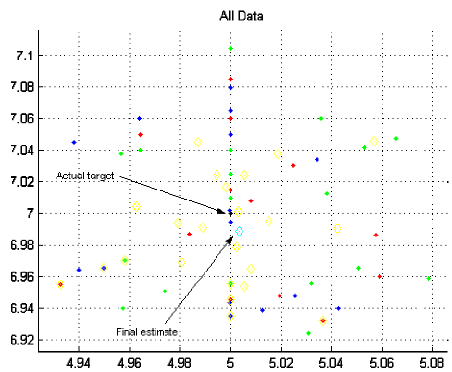


**Figure IV-7 Three sensor, single target estimate data**

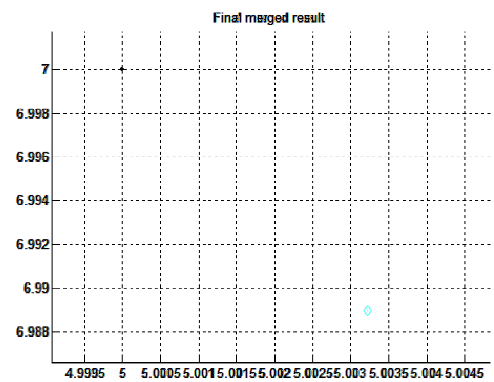


**Figure IV-8 Three sensor, single target final estimate**

*18. Results*

One aspect that can be seen more prevalently in this scenario is that the intermediate merged estimates, shown as yellow diamonds, are not collocated with as many individual estimates; as was obvious when only one sensor was employed. Where a merged estimate is collocated with a sighting it is due to the fact that only one sensor detected the target during that time step. The diagram below, , focuses down on the final merged result. The final estimate was (5.0032, 6.9890) with a standard deviation of (0.1754, 0.1784) in the major and minor axes respectively and is shown as the cyan diamond. In scenario one, where only one sensor was employed, the estimates were (5.0144, 6.9960). There is a marked improvement in the x coordinate and only a slight deterioration in the y coordinate. From this it can be deduced that the more sensors the better the estimate; or the more sightings of a target the better the estimate.

*19. Analysis*

Introducing more sensing platforms does not provide a more accurate estimate of the target locations. In fact looking at it can be seen that there are individual and intermediate merged estimates that provide a closer estimate to the actual target location. However a major factor of the final merged estimate is the associate major and minor covariances. This has been significantly reduced and is undoubtedly less than the individual sightings and merges (although no metric was recorded for this previous results would indicate this to be true). Another interesting aspect is the increased number of intermediate merges that has occurred. By looking closely at Figure IV-7 it can be seen that there are far more intermediate merges (indicated by the yellow diamonds) without collocated sightings. This again would suggest that the final merge will have the best possible data being passed to it in order to arrive at a more confident estimate for the target locations. By performing the intermediate merge step it effectively allows the individual sightings, those that are farthest from the actual target location, to be considered as a merged sighting as opposed to an individual sighting. By not performing this step could potentially result in a poorer estimate of the target locations.

## H. Scenario 4

*20. Test*

Scenario four looks to introduce more targets, in the same manner as scenario two. The platform routes are the same as those in scenario 3. It is expected that there should be no adverse affects to the algorithm and all targets should be located with high confidence.

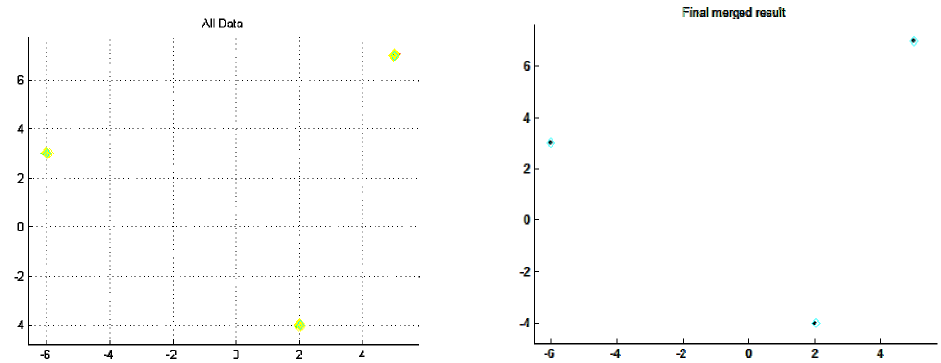


**Figure IV-9 High-level view of three targets, three sensors**

*21. Results*

The simulation results show that the target locations were successfully estimated. The graphs above, specifically the left most target, will be expanded further to allow analysis of an individual target to be made. The results shown in the diagram in Figure IV-11, show how randomly the individual estimates are distributed. The individual merged estimates, yellow diamonds, are generally located closer to the actual target indicating that the individual merging is providing a better estimate. The final estimate is, on this occasion, not the closest estimate however it is would be impossible to select an estimate and say it is better than the final estimate as the location of the target is not known. As such the final estimate is assumed to be the best.

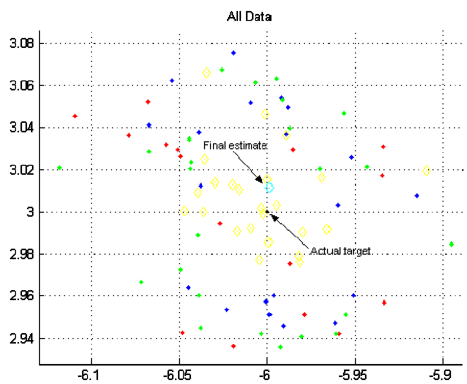


**Figure IV-11 Three sensor, three target estimate data**

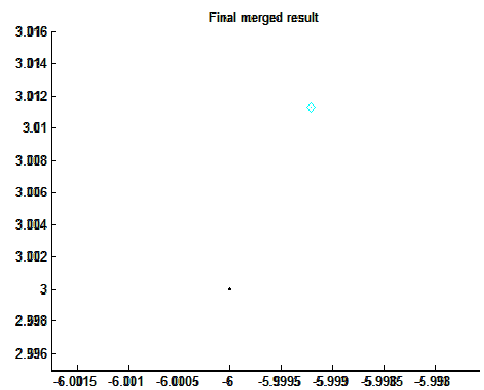


**Figure IV-10 three sensor, three target final estimate**

The final merged estimates for the three targets are (-5.9992, 3.0113), (2.0126, -3.9932) and (5.0078, 6.9875) with standard deviations of (0.1614, 0.1771), (0.1713, 0.1661) and (0.1801, 0.1735) respectively.

*22. Analysis*

In comparison to the single sensor results, which were (-6.0102, 3.0122), (5.0068, 6.9904) and (1.9993, -4.0100) with associated standard deviations of (0.3067, 0.2291), (0.3150, 0.2418) and (0.3140, 0.2422) respectively, it can be seen that although there is only an overall slight improvement on the estimated locations, the standard deviations have significantly reduced giving more confidence that the target is where the algorithm estimates it to be. It worth noting again that there is a vast spread of individual sightings around the graph. The intermediate merged estimates are much tighter packed about the actual target location. This again emphasises the importance of the intermediate merge step that is performed.

## I. Scenario 5

*23. Test*

Due to the nature of the algorithm and the random variables used to create the target sightings in the simulation each execution of the algorithm produced different results. Although this is closely representative of the real world it did mean that the repeatability of the tests was very low. In order to provide a fair representation of the estimates the algorithm was executed 50 times where the scenarios always had three targets; the first containing one platform, the second two and the final three platforms. By executing these scenarios it was hoped that an average estimate could be obtained.

*24. Results*

From the test the following results were obtained:

- One sensor:
  - Target location estimates (x, y):
    - (-6.0092, 3.0126)
    - (5.0079, 6.9991)
    - (1.9983, -4.0118)
  - Overall standard deviation estimate (maj, min):
    - (0.0839, 0.1842)

- Two sensors:
  - Target location estimates (x, y):
    - (-6.0088, 3.0187)
    - (1.9999, -3.9984)
    - (5.0084, 6.9991)
  - Overall standard deviation estimate (maj, min):
    - (0.0551, 0.1407)

- Three sensors:
  - Target location estimates (x, y):
    - (-6.0128, 3.0090)
    - (1.9975, -4.0007)
    - (5.0037, 6.9975)
  - Overall standard deviation estimate (maj, min):
    - (0.0428, 0.1208)

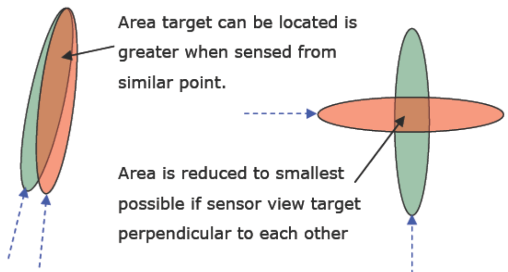


**Figure IV-12 Reducing possible target location by sensing the target from different angles**

*25. Analysis*

Comparing the estimated locations of the targets based on one, two and three platforms it can be seen that the estimated locations vary only very slightly. The main noticeable change is in the standard deviation. As the number of sensors increases so the error in the estimate decreases. This is due to the individual sightings that are then merged in the intermediate step, occurring at differing angles. As shown in the diagram in Figure IV-12, the estimated location improves when disparate sensors detect a target from different view points.

## J. Scenario 6

*26. Test*

The standard deviations vary from sensor to sensor. It is likely that location estimates from disparate sensors of varying types will be required to be merged. By increasing the standard deviation of the sensors, the location estimates produce by the algorithm can be analysed along with the resulting standard deviation.

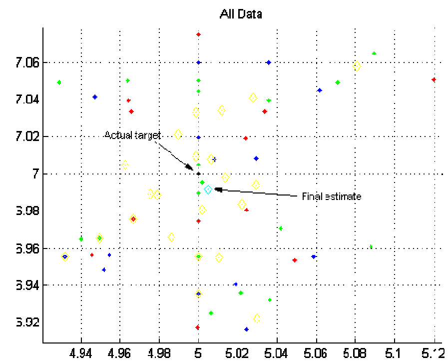


**Figure IV-13 Varying standard deviations**

*27. Results*

The scenario was run with three sensors and only one target; the final estimate is (5.0049, 6.9914) with a standard deviation of (0.6440, 0.6509). This scenario is similar to that of scenario 2 and as such the results are comparable. The increased standard deviations reflect directly onto the resultant standard deviation estimate.

*28. Analysis*

The noise on the sensors employed has been increased to see how the introduction of greater errors affects the overall effectiveness of the algorithm. As can be seen in Figure IV-13 the estimated target position is still acceptable, in fact varying very little from tests performed with 'better' sensors. There is a marked increase in the covariance in both the major and minor axes. This was expected due to the degraded sensor covariance.

## K. Scenario 7

*29. Test*

This scenario looks at how the algorithm handles targets that are within very close proximity to each other. The new target locations are (5, 7), (5, 5.5) and (6.5, 7). This aims to stress the distance checking aspect of the algorithm in order to highlight any potential pitfalls with the algorithm.

*30. Results*

The algorithm has managed to confidently provide an estimate of one target however it has failed to provide a suitable estimate for the other two targets. The simulation results are (5.2368, 6.1670) and (6.5426, 7.0052), with standard deviations of (0.1060, 0.1131) and (0.1560, 0.1762) respectively. The incorrect estimate will not provide confidence in the algorithm and as such will affect the end confidence in the other, correct estimate.

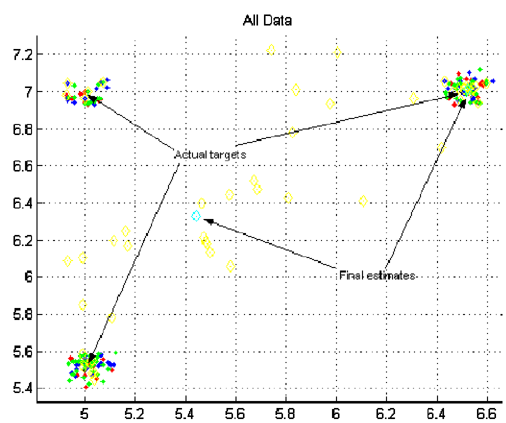


**Figure IV-14 Close targets estimates data**

*31. Analysis*

As can be seen in Figure IV-14 the individual target sightings are clustered about each target, as would be expected. However the intrinsic errors contained within each sighting mean that spurious intermediate merges take place. This coinciding with the minimum allowable distance acting on the limit means that the final estimated location of one of the targets comes out correct yet only one other estimate is provided for the other two targets.

## L. Overall Analysis

In general the algorithm has worked as expected and provides confident estimates of the target locations. When the errors in the sensors is increased the estimated locations are still accurate however, as can be expected, the covariance's increase; this is relative to the increase in the errors of the individual sensors. The distance measurement of the algorithm works in the expected manner until the targets become physically to close for them to be distinguishable.

## V. Conclusion and future work

An algorithm has been developed to allow confident estimates of stationary target locations to be obtained from multiple sensor-based platforms combing an area of interest. The use of the Joint Probability Density Function to locate a single target has been extended to multiple targets to be located. Through the use of distance measurements to check if targets are viably the same along with checks to ensure targets are within an agreeable one standard deviation of each other the merging process could occur. The extensive testing of the algorithm has proven that multiple sensor-based platforms can contribute to the successful location of targets providing high levels of accuracy as indicated by the resulting standard deviations. The algorithm does encounter some problems when the targets are with a close proximity to each other and as such fails to provide confident estimates of the targets within the area of interest. The algorithms employed are generally simple, yet adaptable. Preliminary work and testing has shown that the algorithm can be easily adapted to cope with scenarios including moving targets. The results from this work indicated that the targets appeared to be located with confidence. The algorithm, although applicable on its own would benefit from future developments. Suitable and applicable future work on the algorithm has been identified that will enhance the algorithm. The algorithm can be extended to include target future-position estimation and tracking; these two aspects go hand-in-hand. The algorithm could also benefit from being made more robust, especially when attempting to discern between two effectively co-located targets. As well as enhancements to the main algorithm, this work can become part of a more complex system where the platform has complete control of both its current and future location allowing it to be completely autonomous. This will allow the algorithm to operate in an environment where intrinsic errors within a system can be passed on to other aspects of the system. The algorithm will not only have to deal with the errors introduced by the sensor but will also have to consider the errors introduced by the platform self-localisation system.

## References


1. Sebastian Thrun, “Probabilistic Algorithms in Robotics”, AI Magazine, Vol. 21, N$^{o}$ 4, 2000.
2. Sebastian Thrun, “Robotic Mapping: A Survey”, Exploring Artificial Intelligence in the New Millenium, Morgan Kaufmann Publishers.
3. Tim Bailey, “Mobile Robot Localisation and Mapping in Extensive Outdoor Environments”, PhD Thesis, August 2002
4. P Swerling, "Studies of Target Detection by Pulsed Radar," Transactions on Information Theory, IRE (later IEEE), April 1960
5. Richard R Brooks & S S Iyengar, “Multi-Sensor Fusion – Fundamentals & Applications with Software”, Prentice Hall, 1998
6. Zohn R. Spletzer, “Sensor Fusion Techniques For Cooperative Localisation In Robot Teams”, 2003
7. George B Thomas Jr & Ross L Finney, “Calculus and Analytic Geometry (9th ed.)”, Addison Wesley 1996.
8. Michael Greenber, “Advanced Engineering Mathematics (2nd ed.)”, Prentice Hall 1998.
9. D.B. Rubin. “Using the SIR algorithm to simulate posterior distributions.” In M.H. Bernardo, K.M. an DeGroot, D.V. Lindley, and A.F.M. Smith, editors, *Bayesian Statistics 3*. Oxford University Press, Oxford, UK, 1988.
10. André Ayoun & Philippe Smets, “Data Association in Multi-Target Detection Using the Transferable Belief Model”, Int. J. Intell. Syst., Vol. 16, No. 10, pp. 11671182, Oct 2001
11. Waltz, E. and Llinas, J., “Multi-Sensor Data Fusion”, Artech House Inc., Boston, MA, 1990.
12. Sean A. Midwood, Ian N. Glenn and Murali Tummala, “A Multi Sensor Data Fusion Algorithm For The USCG’S Vessel Traffic Services System”, 1998, IEEE
13. J. Neira and J.D. Tardos, “Data association in stochastic mapping using the joint compatibility test”, IEEE Transactions on Robotics and Automation, 17(6):890-897, 2001.
14. J. Leonard, H. Durrant-Whyte, “Mobile robot localization by tracking geometric beacons”, IEEE Trans. on Robotics and Automation Vol. 7, pp. 89-97, June 1991.
15. Ashley W. Stroupe, Martin C. Martin, Tucker Balch, “Merging Probabilistic Observations for Mobile Distributed Sensing”, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, December 2000
16. Smith & Cheeseman. “On the Representation and Estimation of Spatial Uncertainty”, The International Journal of Robotics Research, 5(4):56-68, 1986.
17. Moravec & Elfes, “High resolution maps from wide angle sonar”, *Proceedings of the IEEE International Conference on Robotics and Automation* - 1985, pp. 116-121.

# Merging probabilistic data of multiple targets detected by multiple sensors

Spillings, J.

2009-11-11T17:54:28Z