A new approach to distributed data fusion

de Waard, H.W.

Citation for published version (APA):
Chapter 2

Basic methods for Data Association

In this chapter elementary concepts concerning sensor systems, target tracking and measurement processing are introduced. State estimation (extended first order Kalman filter, IMM filter) and gating are described. After the introduction of the data association problem, four of the most commonly used data association methods are presented: global nearest neighbor (GNN), the all-neighbors data association approach (PDA, JPDA) and multiple hypothesis tracking (MHT). At the end of the chapter a performance comparison between the different approaches is made, followed by the summary and conclusions section.

2.1 Targets and sensor systems

Earlier, in section 1.2, a sensor has been defined as a measuring device which processes the signal received from the target to derive one or more characteristic properties of a target. Typical sensor systems, such as radar, infrared (IR), and video, report measurements from diverse sources: targets of interest, background noise sources such as clutter, or internal error sources such as thermal noise. In this section a number of methods to obtain information about a target is introduced. Use has been made of [120], [25] and [88].

A radar system is an electromagnetic system for the detection and location of objects. The word radar is a contraction of the words radio detection and ranging [120]. Functionally, a radar system consists of a transmitting antenna emitting electromagnetic radiation, a receiving antenna, an energy-detecting device or receiver and a processing component (Fig. 2.1). A fraction of the transmitted signal is bounced off the reflecting target, and reradiated in all directions.

It is the energy reradiated in the direction of the radar which is of prime interest. The receiving antenna collects the returned energy and delivers it to the receiver, where it is processed to detect the presence of the target and to extract its location, velocity etc. relative to the radar.
CHAPTER 2. BASIC METHODS FOR DATA ASSOCIATION

The passive nature of infrared search and tracking sensors make them eminently suitable for military tracking systems since their operation does not betray the presence of the platform which carries them [25]. An infrared sensor has excellent angular measurement capabilities. The energy an infrared sensor detects is either produced by the target itself or reflected off the target from other sources, such as the sky and earth. The primary sources of aircraft or rocket target energy include hot exhaust gases, exposed hot engine components, for example tailpipes, and aerodynamically heated aircraft surfaces. Functionally, an infrared or video sensor consists of a sensor optics part, a receiver and a processing component (Fig. 2.2). Infrared performance is degraded in smog, smoke and dust and is limited in range.

Video sensors operate in the visible portion of the electromagnetic spectrum.
The visual spectrum image is primarily produced by objects reflecting the electromagnetic energy falling on them. The performance of video is limited in smog, smoke, dust and nighttime operation and limited in range.

2.2 Target tracking and measurement processing

A typical multitarget tracking system contains the functions shown in Fig. 2.3. A data set can contain measurements related to objects of interest received during a sensor sweep of a certain observation volume or area. After receiving a new data set, the measurements it contains are taken successively and the already existing tracks are predicted to the associated measurement times. For this prediction step specific dynamic target models for the different tracks are assumed. The gating function determines for each measurement the tracks for which the measurement is a likely update candidate. Gating is performed primarily to reduce the number of unnecessary computations during the association process [25]. The data association function partitions the measurement data into sets of measurements, related to by the same source. The result of the data association is used to update existing tracks. For measurements that cannot be assigned with existing tracks new, tentative tracks are initiated. A tentative track becomes confirmed when the number and quality of the measurements included in the track satisfy certain confirmation criteria. Tracks which do not fulfil preset similarity criteria are deleted.

The multitarget tracking systems considered here will receive measurements from scanning sensors, which means that the sensor will observe a target only for a limited amount of time. Special measures are taken to minimize the occurrence of measurements which do not originate from the targets of interest by removing returns which possibly correspond with extraneous sources like terrain features, sea waves, rain, etc., using appropriate dynamical thresholds on the signal strength [25]. To simplify the association process, the assumption

![Figure 2.3: A typical multitarget tracker.](image-url)
is made that during a certain scan a sensor can receive at most one measurement from a given target.

The search volume of the track-while-scan (TWS) sensor can be divided into radar resolution cells which depend on the radar characteristics. Assume for a pulsed TWS radar that during the detection process \( k \) cells are used and \( n \) pulses are transmitted. If a reflected signal exceeds a certain pre-set threshold a detection decision is taken. Detections originating from the same radar cell are assumed to correspond with the same target. Sensor processing may not be able to resolve closely spaced targets in the same sensor resolution cell. The consequence is that a single measurement may actually be a composite measurement that was formed from two or more targets. Koch [76] proposes a Multiple Hypothesis Tracking approach with explicit handling of resolution conflicts for two targets which has been applied successfully to real data.

### 2.3 State estimation

In this section the state estimation process is discussed. Assume that the dynamical target model is given by \( \vec{s}_{k+1} = \Phi \vec{s}_k \) in which \( \vec{s}_k \) describes the actual state vector of the target at time \( t_k \), where \( k \) represents the scan number. Normally, the state vector contains the actual position and velocity information of the target of interest, but it may also contain acceleration information for the target. Assume that the state vector is given by \( \vec{s}_k = [x(t_k), y(t_k), v_x(t_k), v_y(t_k)]^T \) and the dynamical model is defined by

\[
\Phi = \begin{pmatrix}
1 & 0 & \delta t & 0 \\
0 & 1 & 0 & \delta t \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}
\]

(2.1)

where the time interval is defined by \( \delta t = t_{k+1} - t_k \). If at \( t_0 \) the initial state of the target is \( \vec{s}_0 \), then the behavior of the target in the state space is defined in a deterministic way.

The real world is more complicated. The state vector is often not accessible to direct measurement but has to be estimated from available measurements. Variations in the target trajectory due to unknown target behavior and measurement errors have to be accounted for. In stochastic filtering both the measurement errors of the sensors and the trajectory disturbances are considered as random errors (stochastic processes).

In this thesis it is assumed that the filter used is based on the extended version of the Kalman filter [73]. The first order Kalman filter procedure combines all information acquired up to and including the latest measurement and computes an optimal estimate for the target state vector at the time of the measurement. If all conditions for the application of the Kalman filter are satisfied, the different state vector estimates \( \hat{s}_k \) are optimal in the sense that the sum of the variances of the errors in the estimates is minimized. When the estimation error distribution is assumed to be Gaussian with zero mean, the distribution is completely specified by the estimation error covariance matrix \( \hat{P}_k \), which is produced by the Kalman filter.

The dynamical behavior of the target is modelled by \( \vec{s}_{k+1} = f(\vec{s}_k) + \vec{w}_k \), \( f(\vec{s}_k) \) is the nonlinear state transition function, \([\vec{w}_k, k = 1, ..., n]\) is a sequence of
2.3. STATE ESTIMATION

Figure 2.4: The structure of the recursive Kalman filter.

uncorrelated variables and \( n \) is the number of measurements to be processed. Each variable \( \tilde{\nu}_k \) is Gaussian distributed with zero mean and covariance matrix \( Q_k \). Such a sequence of variables is called a white noise sequence. The measurement process is modelled by the nonlinear relationship

\[
\tilde{z}_k = h(\tilde{s}_k) + \tilde{\nu}_k
\]  

(2.2)

where \( h(\tilde{s}_k) \) represents the ideal (noiseless) connection between the measurement and the state vector at the time of measurement \( t_k \). Each variable \( \tilde{\nu}_k \) is Gaussian distributed with zero mean and covariance matrix \( R_k \). The sequence of variables \([\tilde{\nu}_k, k = 1, \ldots, n]\) is uncorrelated and also forms a white noise sequence. Furthermore, the variables \([\tilde{\nu}_k, k = 1, \ldots, n]\) are not correlated with the variables \([\tilde{w}_k, k = 1, \ldots, n]\).

The extended first order Kalman filter is recursive and characterized by two phases (Fig. 2.4). In the first phase the estimated state vector \( \hat{s}_k \) and the error covariance matrix \( \hat{P}_k \) from the previous update are predicted to the time of the measurement to be processed. For this prediction the following set of equations is used

\[
\hat{s}_{k+1} = f_k(\hat{s}_k)
\]

(2.3)

\[
\hat{P}_{k+1} = F_{\hat{s}_k} \hat{P}_k (F_{\hat{s}_k})^T + Q_k
\]

(2.4)

where \( \hat{P}_{k+1} \) is the predicted error covariance matrix and \( F_{\hat{s}_k} \) is the Jacobian of \( f(\tilde{s}_k) \), determined for the earlier estimated state vector \( \hat{s}_k \) [30].

In the second phase the predicted state vector \( \hat{s}_{k+1} \), calculated during the first phase using eq. 2.3, is used to determine the innovation \( \tilde{z}_{k+1} - h(\hat{s}_{k+1}) \). The innovation is used to determine the new state vector estimate \( \hat{s}_{k+1} \). Furthermore the new estimate \( \hat{P}_{k+1} \) for the error covariance matrix of the state vector is determined. The update equations to determine the new estimates are given by

\[
\hat{s}_{k+1} = \hat{s}_{k+1} + K_{k+1} \times (\tilde{z}_{k+1} - h(\hat{s}_{k+1}))
\]

(2.5)

\[
\hat{P}_{k+1} = \hat{P}_{k+1} - K_{k+1} H_{\hat{s}_{k+1}} \hat{P}_{k+1}
\]

(2.6)

where \( H_{\hat{s}_{k+1}} \) in eq. 2.6 is the Jacobian of \( h(\hat{s}_{k+1}) \) for the predicted state vector
The Kalman gain \( K_{k+1} \) can be determined by
\[
K_{k+1} = \hat{P}_{k+1} R_{k+1}^{-1} (H s_{k+1} \hat{P}_{k+1} (H s_{k+1}) + R_{k+1})^{-1}
\]

Normally, the times of the measurements to be processed differ. To use the Kalman prediction equations, it is necessary to use a target behavior model\(^1\). The fundamental problem is that normally there is not enough information available to determine the exact behavior of the target at each moment in time. This means that an approach is necessary which can generate different hypotheses for the dynamical behavior of a target which are ranked according to a defined ranking factor.

The basic idea of all multiple model approaches, as applied to tracking manoeuvring targets, is that manoeuvres are typically abrupt deviations from basically straight-line target motion [25]. Because this process is difficult to model with a single manoeuvre model, multiple models representing different potential target manoeuvres are run in parallel and continuously evaluated using filter innovation histories. The output of the filtering process typically is either a probability-weighted composite of the individual filters or the output of the filter with the highest probability. A very efficient implementation of the multiple model approach is based on the interacting multiple models (IMM) algorithm, first proposed by Blom [29]. The unique feature of the IMM approach is the manner in which the state estimates and the error covariance matrices from the multiple models are combined according to a Markov model\(^2\) for the transition between target manoeuvre states. Here a short description of the IMM filter is given, based on [25] and [1]. It is assumed that the total number of target (manoeuvre) models is \( n \), which results in a bank of \( n \) parallel filters. Due to the nature of the IMM filter multiple tracks per target are maintained. For the discussion, it is assumed that extended first order Kalman filtering is used and that gating and data association have been carried out. Thus, measurements to be processed have already been assigned to the tracks of interest.

The process flow of an IMM filter is shown in Fig. 2.5. Using the predicted state vectors and covariance matrices and the probabilities determined by the probability update process, the update process carries out updates for the different maintained tracks. Using the updated information, the probability update function calculates updated dynamical model probabilities. During the update step, the calculated probability for the dynamical target model is added to the updated state vector and error covariance matrix information produced by the corresponding filter. The updated information is used by the merge and mixing processes. During the merge stage, for each dynamical model the state vectors and covariance matrices are merged into just one state vector and covariance matrix. Using the assumed Markov transition properties between models, new filtered state estimates, error covariance matrices and corresponding model probabilities are computed for each model via the mixing process. A detailed mathematical description of the IMM filter can be found in [25] and [1].

---

\(^1\)When the prediction time interval increases, the different terms in the covariance matrix representing the accuracy also increase.

\(^2\)Loosely speaking, a model is Markov if, given the state \( x_{k-1} \), no earlier states \( x_j, t_j < t_{k-1} \), have influence on the state \( x_k \), i.e. if \( t_1 < t_2 < \ldots < t_{k-1} < t_k \), then the conditional probability of \( x_k \), conditioned on \( x_{k-1}, x_{k-1} \), is given by \( P_{x_k|x_{k-1},x_1} (x_k | x_{k-1}, x_1) = P_{x_k|x_{k-1}} (x_k | x_{k-1}) \) [6].
2.4 Gating

Assume that estimates for the target state vector $\hat{s}(t_k)$ and the corresponding error covariance matrix $\hat{P}(t_k)$ are available for each tracked target. It is further assumed that the measurement $\vec{z}_k$ produced at time $t_k$ is modeled by a random variable with a Gaussian distribution function. The measurement probability density function is defined by

$$f_{m}(\vec{z}_k) = \frac{1}{\sqrt{(2\pi)^n |B|}} e^{-\frac{1}{2} [\vec{z}_k - h(\hat{s}(t_k))]^T B^{-1} [\vec{z}_k - h(\hat{s}(t_k))]} \quad (2.8)$$

where $\vec{z}_k$ is the measurement vector, $h(\cdot)$ is the transformation from Cartesian to polar coordinates, $n$ is the dimension of the measurement vector and $B$ is the covariance matrix $B = H_k P_k (H_k)^T + R_k$. $R_k$ is the measurement noise covariance matrix and $H_k$ is the Jacobian of $h(\cdot)$ taken at the predicted state vector $\hat{s}(t_k)$.

Samples drawn from a Gaussian distributed population tend to fall in a single cloud or cluster. The center of the cluster is determined by the mean vector $h(\hat{s}(t_k))$ and the shape of the cluster is determined by the covariance matrix $B$.

Formally seen, $f_{m}^{m}$ in (2.8) is not a measurement density function but the conditional density function $f_{m}^{m}(\vec{z}_k \mid \vec{z}_1 \cap \ldots \cap \vec{z}_{k-1})$ which is conditioned on the earlier processed measurement set $\{\vec{z}_1, \ldots, \vec{z}_{k-1}\}$.

Figure 2.5: The principle of the IMM filter.
The quantity \[ (\vec{z}_k - h(\vec{s}(t_k)))^T B^{-1} (\vec{z}_k - h(\vec{s}(t_k))) = r^2, \]
called the Mahalanobis distance, determines a hyperellipsoid with the property that all points have the same constant density value \([56]\). The variable \(r^2\) is \(\chi^2\)-distributed.

![Figure 2.6: An elliptical gate is positioned at the predicted target position.](image)

Gating is a technique for eliminating unlikely measurement-to-track pairings. A correlation gate is the volume in the measurement space within which likely measurements, originating from the object corresponding with the track hypothesis, are expected. The first possibility is to apply a hyperellipsoid gate, using the Mahalanobis distance

\[ (\vec{z}_k - h(\vec{s}(t_k)))^T B^{-1} (\vec{z}_k - h(\vec{s}(t_k))) \leq r^2 \]  \hspace{1cm} (2.9)

Using the dimension of the measurement vector and accepting a certain risk of making a wrong decision, the factor \(r^2\) can be read from the \(\chi^2\)-distribution table. A measurement falling within the gate is assumed to be a likely association candidate for the track under consideration. In terms of cpu-time the use of the elliptical gate is rather expensive when the calculation has to be carried out for each possible target-measurement combination, however unlikely. To reduce the use of computation time a pre-gating method is necessary which removes the very unlikely measurements before application of the elliptical gate.

The proposed pre-gating method is based on first defining a correlation ellipse (ellipsoid), defined by eq. 2.9 and a required \(r^2\)-value, for each track in the measurement space. The next step is to approximate this ellipsoid by a hypercube of the correct dimension. A relatively fast method to approximate a correlation ellipsoid in a \(n\)-dimensional measurement space is given in \([48]\). Due to the fact that the volume covered by the approximation is guaranteed to be somewhat larger than the volume of the ellipsoid, measurements which do not fall within the cube certainly do not fall within the ellipsoid. In Fig. 2.7 the principle of the approximation method is illustrated, using as example a three-dimensional correlation ellipsoid.
2.5 Out-of-sequence processing

It is common practice to distribute measurements in batches. In a configuration with multiple sensors, this means that there is a significant probability, that a platform receives an earlier measurement from a certain target after a later one has been already processed. This is the out-of-sequence measurement situation [16]. But even in the case that individual measurements are distributed, out-of-sequence measurements are possible. In a data fusion application where the sensors or platforms are non-collocated, two types of delay may occur. If the bandwidth of available communication channels is exceeded, the transmission of measurements can be significantly delayed, which may result in out-of-sequence measurements.

The second type of delay may occur when a platform has to process a large number of measurements on limited processing resources. If processing results are distributed to other platforms, out-of-sequence measurements possibly result (see chapter 7 e.g.).

With respect to data association, there is a fundamental problem related to the processing of out-of-sequence measurements. Assume that a delayed measurement with measurement time \( t_l \) is received at time \( t_k \). Due to pruning actions performed in the time interval \( t_k - t_l \), the set of track hypotheses at time \( t_k \) may differ significantly from the set of track hypotheses existing at time \( t_l \). Theoretically, it is possible that the track hypothesis to which the measurement would be assigned if it was received at time \( t_l \), does not exist anymore. The probability that this situation occurs, increases when the involved time interval increases. The only way to prevent this situation from occurring, is to refrain from pruning actions. Practically seen, such a kind of approach is not very realistic.
2.5.1 Filtering Out-of-sequence measurements

A short overview is given of filtering approaches found in literature. Blackman [25] identifies at least four potential approaches to solve the out-of-sequence problem. The processing of an out-of-sequence measurement is considered as a filtering problem, where a delayed measurement, measured at time $t_l$, has to be included in a track hypothesis that has just been updated with a measurement, produced at a later time $t_k$.

The simplest approach is to just ignore the delayed measurement. In the second approach the state estimate is retrodicted to the time of the delayed measurement ($t_l$) and the innovation is formed. The innovation $\tilde{y}$ is defined as the difference between the measurement and the predicted measurement:

$$\tilde{y}_l = \tilde{z}_l - h(\tilde{s}_l) \quad (2.10)$$

where $h(\tilde{s}_l)$ represents the ideal (noiseless) connection between the measurement and the state vector at time $t_l$ (section 2.3). The innovation $\tilde{y}_l$ is used to update the state estimate in the same manner that a current measurement would be used. Process noise effects are ignored and the filter error covariance matrix remains referenced to the current time. The gain and subsequent covariance update are also computed as if the measurement was not delayed.

In a third approach, proposed by Bar-Shalom and Li [16], the state vector and the covariance matrix are computed for the time of the delayed measurement. In this approach an extended first order Kalman filter is used (section 2.3). Again, process noise effects are ignored. After determination of the innovation and the corresponding covariance matrix, the filter gain is calculated. Using the standard equations, the newly updated state and its covariance matrix are determined, which now incorporates the delayed measurement.

The fourth method discussed here is based on the approach given in [13]. This optimal method accounts for the effects of the process noise that entered the system during the time interval from $t_l$ to $t_k$. Again, an extended first order Kalman filter is used. Generally, a number of updates will have been performed before the delayed measurement is received. The first step is to transform the set of measurements received in the time interval $[t_l, t_k]$ to an equivalent measurement or tracklet [53]. The equivalent measurement is calculated using eq. 7.15 (section 7.2.3) and the corresponding covariance matrix using eq. 7.33 (section 7.2.5). This means that the same problems may occur with the calculation of the equivalent measurement covariance matrix, as will be identified in section 7.3.3.3. Using the equivalent measurement, the state vector and error covariance matrix are retrodicted to the time of the delayed measurement. Using modified update equations, the actual updated state and covariance matrix are determined, incorporating the delayed measurement.

Three filtering approaches have been discussed, which propose an (optimal) algorithm to process the delayed measurement. The first method does not process the measurement, but discards it. A disadvantage of the second and third method is, that process noise effects are ignored. A disadvantage of the fourth method is, that due to the chosen tracklet calculation method numerical problems may easily occur.

---

4 Retrodict: to utilize present information or ideas to infer or explain a past event or state of affairs (Definition from the Merriam-Webster Online Dictionary).

5 Tracklets will be extensively discussed in chapter 7.
2.5. OUT-OF-SEQUENCE PROCESSING

2.5.2 Alternative approach to process out-of-sequence measurements

The objective of this section is to define an alternative approach to process out-of-sequence measurements, which takes into account process noise effects and which is numerical stable. The delayed measurement is received at time \( t_k = t_l + \Delta t \). We are only interested in out-of-sequence measurements, for which the delay \( \Delta t \) does not exceed a pre-defined time interval \( \Delta t_d \), which has a value significantly larger than zero. Measurements which have a larger delay are not taken into consideration. It is assumed that the state vectors and error covariance matrices for a track hypothesis are saved during the time interval \([t_l - \Delta t_d, t_k]\). When a track hypothesis is pruned (see also section 2.6.3), also the related saved state vectors and error covariance matrices are deleted from memory. The most recent saved state vectors and error covariance matrices of the different tracks with a time \( t_l \leq t_k \) are predicted to measurement time \( t_l \), and correlations are carried out to determine hypotheses suitable for association of the measurement. For each of the candidate track hypotheses, the delayed measurement \( \vec{z}_l \) is predicted to the time \( t_k \) (section 2.5.2.1), when it is actually received. The predicted measurement \( \vec{z}_k \) is then processed as if it were generated at time \( t_k \), using for example a Kalman filter.

2.5.2.1 Measurement prediction

Assume that a certain track hypothesis, with predicted state vector \( \vec{s}_l \) and error covariance matrix \( \vec{P}_l \), is selected as source for the delayed measurement. Using the track information, the probability density function for the measurement \( \vec{z}_l \), valid at time \( t_l \), is defined by the multivariate normal probability density function \( \mathcal{N}(\vec{\mu}_l, \vec{B}_l) \), where \( \vec{\mu}_l = \mathbb{E}(\vec{z}) = h(\vec{s}_l) \), \( \vec{B}_l = H_{\vec{s}_l} \times \vec{P}_l \times H_{\vec{s}_l}^T + R_l \) and \( \vec{R}_l \) is the measurement covariance matrix valid at time \( t_l \). Furthermore, \( H_{\vec{s}_l} \) is the Jacobian of \( h(\vec{s}_l) \), where \( h(\cdots) \) represents the ideal (noiseless) connection between the measurement and the state vector at the measurement time \( t_l \) (section 2.3). The function \( \mathcal{N}(\vec{\mu}_l, \vec{B}_l) \) represents the predicted distribution of the measurement related to the target of interest. The idea behind the new method is to predict this distribution function to the time that the delayed measurement is actually received.

The next step is to predict the measurement \( \vec{z}_l \) to time \( t_k \). The relation between the measurement vector and the target state vector at time \( t_l \) is given by

\[
\vec{z}_l = h(\vec{s}_l) + \vec{v}_l
\]  

where \( \vec{v}_l \) is the measurement noise. The evolution of the target state vector during the time interval \( \Delta t = t_k - t_l \) is given by the nonlinear model

\[
\vec{s}_k = f(\vec{s}_l) + \Gamma \times \vec{w}_l
\]

where \( f(\vec{s}_l) \) represents the nonlinear state transition function and \( \vec{s}_l = [x_l, y_l, z_l, \dot{x}_l, \dot{y}_l, \dot{z}_l]^T \). With respect to section 2.3, \( \vec{w}_l \) is replaced by \( \Gamma \times \vec{w}_l \) [14]. The process noise contributes to the velocity, which means that the factor...
Γ is defined by
\[
\Gamma = \begin{pmatrix}
\Delta t & 0 & 0 \\
0 & \Delta t & 0 \\
0 & 0 & \Delta t \\
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{pmatrix}
\] (2.13)

It is easily shown that at time \( t_k \) the predicted measurement distribution is given by
\[
\tilde{\mu}_k = h(\bar{s}_k)
\]
\[
B_k = H\bar{s}_k \times [F\bar{s}_k \times \bar{P}_l \times F^T + \Gamma \times Q_l \times \Gamma^T] \times H^T + R_k
\] (2.14)

where \( Q_l = E(\bar{w}_l \times \bar{w}_l^T) \) and \( F\bar{s}_k \) is the Jacobian of \( f(\bar{s}_l) \), determined for the predicted state vector at time \( t_k \).

Generally, a multivariate normal probability density function can be written as a function of independent and identically distributed standard normal probability density functions, which are denoted by \( \mathcal{N}(0, 1) \) [80, 23]. Suppose \( \bar{Z} = (Z_1, \ldots, Z_m)^T \), where the probability density function of each independent random variable \( Z_i \) \( (i \in [1, m]) \) is given by \( \mathcal{N}(0, 1) \). The mean value of \( \bar{Z} \) is defined by \( E(\bar{Z}) = (0, \ldots, 0)^T \) and the covariance matrix is defined by \( E(\bar{Z} \times \bar{Z}^T) \).

Because the different random variables are independent \( E(\bar{Z} \times \bar{Z}^T) = I_m \), where \( I_m \) is the \( m \times m \) unit matrix. The multivariate normal probability density function \( \mathcal{N}(\hat{\mu}, \Sigma) \), \( \hat{\mu} = E(\bar{Z}) \), is also known as the \( m \)-variate standard normal probability distribution function. Assume that the random variable \( \bar{x} \) has the probability density function \( \mathcal{N}(\hat{\mu}, \Sigma) \), where \( \Sigma \) is a positive definite matrix. This means that all of its characteristic roots (eigenvalues) are positive [74]. The dimension of \( \bar{x} \) is \( n \). Define \( \bar{x} = \hat{\mu} + B \times \bar{Z} \), where the dimension of \( B \) is \( n \times m \).

Substitution in the covariance matrix of \( \bar{x} \) produces
\[
\Sigma = E(\bar{x} - \hat{\mu}) \times (\bar{x} - \hat{\mu})^T = E(\bar{Z} - \bar{\mu}) \times (\bar{Z} - \bar{\mu})^T
\]
\[
= B \times E(\bar{Z} \times \bar{Z}^T) \times B^T = B \times B^T
\] (2.15)

A proof is necessary to show that for each relevant covariance matrix \( \Sigma \) there is at least a matrix \( B \), which fulfills eq. 2.15. There exists always a real orthogonal \( n \times n \) matrix \( P \) such that [74]
\[
\Sigma = P \times diag(\lambda_1, \ldots, \lambda_n) \times P^T
\]
\[
= P \times diag((\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n}) \times P^T \times P \times diag((\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n}) \times P^T
\]
\[
= B \times B^T
\] (2.16)

where the \( \lambda_i \) \( (i \in [1, n]) \) are the characteristic roots or eigenvalues for the symmetrical matrix \( \Sigma \). An orthogonal matrix has the properties that \( P^T = P^{-1} \) and that \( |P| = \pm 1 \). Furthermore, \( B \) is defined by
\[
B = P \times diag((\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n}) \times P^T
\] (2.17)
2.5. **OUT-OF-SEQUENCE PROCESSING**

and has dimension \( n \times n \). This matrix is also called the square root of the matrix \( \Sigma \). The inverse of the matrix \( B \) is given by

\[
B^{-1} = P \times \text{diag}(\sqrt{\frac{1}{\lambda_1}}, \ldots, \sqrt{\frac{1}{\lambda_n}}) \times P^T
\]  

(2.18)

Our next objective is to transform the variate \( \vec{z}_l \) from the multivariate normal probability density function \( N(\vec{\mu}_l, B_l) \) to a corresponding variate \( \vec{z}_k \) from the multivariate normal probability density function \( N(\vec{\mu}_k, B_k) \). Using the multivariate standard normal distribution \( N(\hat{Z}, I_n) \) and the square roots \( B_{ls} \) and \( B_{ks} \) for the covariance matrices \( B_l \) and \( B_k \), it is possible to write

\[
\vec{z}_l = \vec{\mu}_l + B_{ls} \times \hat{Z} \\
\vec{z}_k = \vec{\mu}_k + B_{ks} \times \hat{Z}
\]  

(2.19)

where \( \hat{Z} \) is the corresponding variate from the multivariate standard normal distribution \( N(\hat{Z}, I_n) \). Due to the fact that \( B_{ls}^{-1} \) does exist, it is possible to write

\[
\hat{Z} = B_{ls}^{-1} \times (\vec{z}_l - \vec{\mu}_l)
\]  

(2.20)

Substitution in eq. 2.19 finally produces the measurement

\[
\vec{z}_k = \vec{\mu}_k + B_{ks} \times B_{ls}^{-1} \times (\vec{z}_l - \vec{\mu}_l)
\]  

(2.21)

which is predicted to time \( t_k \).

### 2.5.2.2 Simulation results

For the test, the scenario shown in Fig. 2.8 has been used. Three non-collocated

![Figure 2.8: The scenario used to test the algorithm, proposed in section 2.5.2.1.](image)
radars, represented by dots, are observing a part of the airspace. It is assumed that each radar has a maximum detection range of 200 km. The assumed measurement accuracy, expressed in standard deviations, is 0.03 km in range and 0.3 degrees in both bearing and elevation. The scan time of each sensor is 5 sec. The process noise standard deviation is set to 30 m/s. The target appears closely to the rightmost sensor in Fig. 2.8 (the blue dot) and flies at an altitude of 1000 m. with a velocity of 500 m/s in the direction of the y-axis. To test the algorithm, developed on the basis of section 2.5.2.1, a Monte Carlo approach is taken. Statistical information has been collected over a total number of 100 runs to collect statistical information.

Fig. 2.9 and 2.10 show the obtained accuracy results, given as function of delay. The accuracies are expressed by standard deviations and are indicated by brackets in the figures. The original measurement tracking without delay is compared with the predicted measurement tracking (section 2.5.2.1). The measurements of the rightmost sensor are delayed. The average accuracy for original measurement tracking is expressed by the covariance matrix $M_m$ and for predicted measurement tracking by the covariance matrix $M_{pr}$, which both are estimated using the equation

$$M_j = \frac{1}{N} \sum_{i=1}^{N} \delta_i^j \times (\delta_i^j)^T$$

where $j \in \{m, pr\}$ and $N$ is the number of updates used. At time $t_i$ the measured difference vector is given by $\delta_i^j = \hat{\vec{s}}_i^j - \hat{\vec{s}}_i^o$, where $\hat{\vec{s}}_i^j$ is the estimated track state vector and $\hat{\vec{s}}_i^o = [x, y, z, v_x, v_y, v_z]^T$ is the actual state vector of the real
To compare the predicted measurement tracking accuracy with the original measurement tracking accuracy, the ratio between the obtained average predicted measurement accuracy volume ($pr$) and the average measurement accuracy volume ($m$) is used, which is defined by (appendix E)

\[ V_r = \sqrt{\frac{M_{pr}}{M_m}} \]  

(2.23)

Furthermore, a comparison is made with the approach in which the delayed measurements are discarded.

For this specific scenario, the conclusion is that for delays less than 0.4 sec. the predicted measurement tracking is almost as accurate as measurement tracking without delay. For this time interval the predicted measurement tracking produces nearly optimal results. For larger delay times, the processed delayed measurements do not improve the tracking results. The conclusion is that delayed measurements with $\Delta t > 0.4$ sec. can be discarded. For this scenario therefore $\Delta t_d = 0.4$ sec. (section 2.5.2).

\footnote{Due to the fact that it is a simulation experiment, at each update time the actual state vector for the real object is known. It is common practice to use this information to evaluate the tracking performance [15].}
2.6 Data Association problem

One of the major difficulties in the application of multitarget tracking involves the problem of associating measurements with the appropriate tracks, especially when there are missing measurements, unknown targets and false alarms (clutter). In section 1.3 it has been stated that the data association problem is NP-hard for a number of sensors $\geq 3$. For the different basic architectures which are traditionally used to realize task force data fusion systems (section 1.4), problems with platform processing can be expected when a large number of measurements have to be processed.

The multitarget tracker, introduced in section 2.2, is envisioned to operate in future environments with high false alarm rates and against complex scenarios containing tight formations. A scanning radar system (section 2.1) is used which makes complete scans over $360^\circ$. During the scan a large number of measurements is collected for which the source is not known. The set of measurements has to be processed by the tracker. It is assumed that each target in the coverage of the scanning radar can produce a maximum of one measurement during a radar scan.

Assume that the multitarget tracker has produced a number of tracks, using radar measurements collected during earlier scans. To determine which measurements are likely candidates to be produced by a certain target, a gate or association ellipse (ellipsoid) is positioned at the predicted state vector of the target in the state space [26]. Consider again Fig. 2.3. The association function receives the output of the gating function and has to partition the sensor measurements into tracks. In cases where a single measurement is within the gate of a single track, the assignment can be immediately made. For closely spaced targets, it is more likely that conflict situations such as those shown in Fig. 2.11 will occur. In this figure association gates, shown as a circle in the $(x, y)$-plane, are shown for two tracks. Only one measurement can be assigned to each of

![Gating example](image)

Figure 2.11: Gating example [25].
most likely distribution of the measurements \( \{M_1, ..., M_6\} \) to the set of targets \( \{T_1, T_2\} \), new targets and false alarms. The association problem can be considered as an optimization problem in which some defined object function has to be minimized. Consider the example given in Fig. 2.12 where a number of hypotheses concerning the origin of the measurements is given after receiving three scans of sensor measurements. The un-connected measurements do not originate from the same source and are mutually independent, representing false alarms or new targets. Theoretically, it is possible to generate all possible data association hypotheses. The next step is to define appropriate probability density functions that the source of a measurement is an already known target, a new target or a false alarm. It is now possible to define a conditional probability function for a data association hypothesis, which expresses the likelihood of the hypothesis given all earlier processed measurements. The hypothesis with the highest probability value is considered to represent the best data association hypothesis. In the next subsections data association methods are discussed.

### 2.6.1 The global nearest neighbor approach

In this section the global nearest neighbor (GNN) method as introduced by Blackman [26] is discussed. The GNN method considers all possible measurement-to-track pairings and generates the most likely assignment hypothesis [25] for the different measurements in a data set. The assignment hypothesis is used to make irrevocable assignments which cannot be influenced by data collected in later scans.

We follow the notation as introduced in [107]. GNN produces only a single data association hypothesis \( \Omega^k \) which is based on the cumulative data set \( Z^k \) up to scan \( k \). The actual data set to be processed is \( Z(k) \), \( \Omega^{k-1} \) represents the data association hypothesis for the cumulative data set \( Z^{k-1} \) and \( \psi_h \) is the data association hypothesis for the current scan \( k \). Using the measurement
CHAPTER 2. BASIC METHODS FOR DATA ASSOCIATION

probability density function defined by equation 2.8 the conditional probability density function of the actual data set $z(k)$, given the compound event $(\Omega^{k-1} \cap \psi_h \cap Z^{k-1})$ \footnote{For example, the event $\psi_h$ is the set containing as primary events the different possible data associations hypotheses for the current scan \cite{71}.}, can be written as

$$f(Z(k) \mid \Omega^{k-1} \cap \psi_h \cap Z^{k-1}) = \Pi_{(i,j) \in \psi_h} \frac{1}{\sqrt{(2\pi)^n | B |}} e^{-\frac{d_{ij}^2}{2}}$$  \hspace{1cm} (2.24)$$

where

$$d_{ij}^2 = [\vec{z}_i - h(\vec{s}_j)]^T B^{-1} [\vec{z}_i - h(\vec{s}_j)]$$  \hspace{1cm} (2.25)$$

The factor $d_{ij}$ indicates how well measurement $\vec{z}_i$ ($\vec{z}_i \in Z(k)$) matches with track $j$ and $\vec{s}_j$ is the state vector of track $j$ predicted to the time of measurement $\vec{z}_i$.

The objective of GNN is to determine the optimal assignment $\psi_h$ by maximizing $f(Z(k) \mid \Omega^{k-1} \cap \psi_h \cap Z^{k-1})$ using the data of the last scan. The data association hypothesis at time $t_k$ is produced by extending the hypothesis $\Omega^{k-1}$ valid at time $t_{k-1}$ with the assignment hypothesis $\psi_h$, formally written as $\Omega^k = (\Omega^{k-1} \cap \psi_h)$. Due to the fact that only the current scan’s data is used to determine an optimal solution, in general the real, but unknown assignment solution for all received measurements will not be found.

2.6.2 The all-neighbors data association approach

An in-depth description of both all-neighbors approaches, introduced in this section, can be found in \cite{14}. The different examples are taken from \cite{25}. Here only the basic principles are explained and it is assumed that a single sensor is collecting measurements. In the all-neighbors approach, after each scan multiple hypotheses are formed and subsequently combined into a single hypothesis. The result is that the track estimate for a given track may contain contributions from more than one measurement. The essence is that all hypotheses are taken into account in the track.

First the probabilistic data association (PDA) method is considered, which is based on the assumption that a single target $i$ has to be tracked in clutter. In the example shown in Fig. 2.13 several measurements are falling within the correlation gate of the target. The set of validated measurements is indicated by $Z = \{\vec{z}_j(t_{k+1})\}_{j=1}^N$, where $N$ is the number of measurements falling within the gate. The origin of the measurements could be the target or clutter. Given that $N$ measurements fall within the gate, $N+1$ hypotheses can be formed. The first one, denoted $H_0$, corresponds with the case that none of the measurements originates from the target and has a probability $p_{0i}$. The probability of hypothesis $H_j$ that measurement $j$ originates from the target is given by $p_{ji}$. Explicit expressions to determine the different probabilities can be found in \cite{25}.

Using the $N$ measurements and the probabilities that they originate from the target, a composite state estimate for the track is determined, using the extended first order Kalman filter introduced in section 2.3. Assume that the $(k+1)^{\text{th}}$ update has to be determined. The innovation $\hat{y}_{k+1}^n$ to be used in the filter is a weighted sum of the innovations associated with the different
measurements:

\[
\tilde{y}_{k+1} = \sum_{j=1}^{N} p_{ji} \times (\tilde{z}_j - h(s^i_{k+1})) \tag{2.26}
\]

where \(\tilde{z}_j\) is the measurement vector of the \(j^{th}\) measurement and \(s^i_{k+1}\) is the state vector of target \(i\) predicted to time \(t_{k+1}\), given by equation 2.3. To determine the new state estimate the extended first order Kalman filter update equation 2.5 is used:

\[
s^i_{k+1} = f(s^i_{k+1}) + K_{k+1} \times \tilde{y}_{k+1} \tag{2.27}
\]

where the gain \(K_{k+1}\) is computed using equation 2.7.

The estimate for the error covariance matrix \(\hat{P}_{k+1}^i\) is determined by:

\[
\hat{P}_{k+1}^i = P^0_{k+1} + \delta P \tag{2.28}
\]

The probability that none of the measurements does correspond with target \(i\) is given by \(p_0\), which means that the probability that one of the measurements corresponds with target \(i\) is given by \(1 - p_0\). The factor \(P^0_{k+1}\) is determined by a weighted sum associated with the possibilities that measurements corresponds with the target or clutter, and is defined by:

\[
P^0_{k+1} = p_0 \hat{P}_{k+1} + (1 - p_0)(\hat{P}^*_{k+1}) \tag{2.29}
\]

where \(\hat{P}_{k+1}^i\) is the predicted error covariance, given by equation 2.4. The contribution \(\hat{P}^*_{k+1}\) is the standard Kalman filter covariance matrix, as computed from eq. 2.6. The factor \(\delta P\) is an increment to reflect the effect of uncertain association on the error covariance matrix and is determined by:

\[
\delta P = K_{k+1} \sum_{j=1}^{N} p_{ji} \times (\tilde{z}_j - h(s^i_{k+1}))(\tilde{z}_j - h(s^i_{k+1}))^T - \tilde{y}_{k+1}(\tilde{y}_{k+1})^T | K_{k+1}^T \tag{2.30}
\]

PDA is based on the assumption that a single target has to be tracked. The method can easily be extended to the multitarget tracking case, resulting in
the joint probability data association (JPDA) method. The situation is now more complicated because the association probabilities have to be computed using all measurements and all tracks. The estimate of the state and the error covariance is again calculated by the equations 2.26-2.30. The difference is the determination of the different probabilities $p_{ji}$ which express the probability that measurement $j$ is assigned to track $i$. In section 2.6, already the subject of the generation of all possible data association hypotheses has been introduced. A discussion of the generation of the different data association hypotheses and the calculation of the hypothesis probabilities is given in section 2.6.3. Consider the typical type of conflict situation, shown in Fig. 2.14. Two measurements fall in the correlation gate of both track $T_1$ and $T_2$. All data association hypotheses are given in table 2.1. Several hypotheses are excluded from the table, because not all involved measurements correlate with both tracks. In the second column

<table>
<thead>
<tr>
<th>Hypothesis Number</th>
<th>Track 1</th>
<th>Track 2</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0.011</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0.086</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0</td>
<td>0.053</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0</td>
<td>0.019</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>2</td>
<td>0.041</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>2</td>
<td>0.306</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>2</td>
<td>0.086</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>3</td>
<td>0.032</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>3</td>
<td>0.239</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>3</td>
<td>0.145</td>
</tr>
</tbody>
</table>

Figure 2.14: Typical conflict situation suitable for JPDA.
the measurements assigned to track 1 and track 2 are given. A 0 means that no measurement has been assigned to the specific track. Suppose that it is necessary to know the probability that measurement 2 is assigned to target 1. Inspection of the different hypotheses tells us that this combination exists in hypothesis $H_3$ and $H_{10}$. Hypothesis $H_3$ has a probability of 0.053 and hypothesis $H_{10}$ has a probability of 0.145. The probability $p_{21}$ is now given as $p_{21} = P(H_3) + P(H_{10}) = 0.198$. All relevant probabilities can be determined in the same way.

JPDA has been used in many applications. For complex scenarios with closely spaced targets, JPDA shows a tendency to coalesce the tracks. This is due to the fact that JPDA forms composite measurements. This can be understood by examination of the example given in Fig. 2.15. The measurements shown in the overlap region of both gates contribute to the state vector estimates of both targets. When this occurs often and the measurements are randomly distributed, this may have as effect that the state vector estimates are moving towards the centroid of both targets and do not represent accurate estimates of the real target positions.

Assume that the constraint is fulfilled that only a maximum of one measurement is produced by each target. A certain unknown assignment exists which provides a correct explanation for the measurements received. If the data association process finds the corresponding hypothesis, optimal tracking results can be expected. Due to the fact that PDA and JPDA average over all possible measurement-track associations, the produced updated track estimates may contain contributions from more than one measurement. Statistically seen the tracks produced will probably be sub-optimal.
2.6.3 Multiple Hypothesis Tracking

Multiple hypothesis tracking (MHT) is a deferred decision logic in which alternative data association hypotheses are formed whenever there are measurement-to-track conflict situations such as shown for example in Fig. 2.12 and Fig. 2.15. Rather then combining these hypotheses, as in the JPDA method, the hypotheses are propagated in anticipation of subsequent data that may resolve the uncertainty [25]. Each hypothesis contains track estimates for the track hypotheses contained in the hypothesis. To obtain improved smoothed track estimates based on all the measurement data, a backward sweep over a fixed time interval could be made to refine the track estimates, the different measurement-track associations and the probability of the hypothesis [30].

The original MHT method, known as Reid’s algorithm, was first presented in [107]. The basic approach in the paper is to generate a set of data association hypotheses to account for all possible origins of every measurement. The algorithm developed by Reid contains the desirable features of multiple-scan correlation, clustering and recursiveness. Multiple-scan correlation is the capability to use later measurements to aid prior associations (correlations) of measurements with targets. Clustering is the process to divide the entire set of global hypotheses in sets of local hypotheses (clusters) that do not interact with each other. Instead of one large tracking problem, a number of smaller tracking problems is solved independently which significantly reduces the necessary amount of computer storage and computation time. The main reason for this reduction is that the association problem to be solved for each independent problem is (much) smaller 8. Finally, it is desirable for an algorithm to be recursive so that the previous data sets do not have to be reprocessed whenever a new data set is received.

In this section a single sensor is assumed which provides information from which the number of targets within the area of coverage of the sensor can be inferred. Radar is an example of this type of sensor. The sensor generates a data set which contains one or more measurements. It is assumed that no more than one measurement per data set can originate from a single target.

We discuss the measurement-oriented MHT approach of [107]. The algorithm maintains the hypothesis structure from scan to scan and updates the hypotheses as data is received. At each scan, a set of hypotheses will be carried over from the previous scan. Each hypothesis is composed of tracks in such a way that no track shares measurements with other tracks in the hypothesis. After receiving a new data set each hypothesis is expanded into a new set of hypotheses by considering all possible measurement-to-track pairings for the tracks within the hypothesis. During this process, the compatibility constraint that tracks within a hypothesis do not share measurements has to be maintained [25]. The hypotheses are generated by the simple, but inefficient mechanism of enumerating all possible associations. An elegant and very efficient alternative is the $k$–best MHT approach, proposed in ([39],[38]), which uses Murty’s algorithm [87] to generate only the $k$–best hypotheses.

To determine an appropriate weighting factor to rank the hypotheses, we will follow Reid [107]. At a certain moment the $k^{th}$ data set $Z(k) = \{\bar{z}_m(k), m = 1,2, ..., M_k\}$ is received from which the data association hypotheses correspond-

---

8An upper limit for the asymptotic complexity of the enumerative (k-best) MHT algorithm is derived in appendix M.
2.6. DATA ASSOCIATION PROBLEM

ing with the cumulative set of measurements \( Z^k = \{ Z(1), \ldots, Z(k) \} \) up through data set \( k \) must be generated. \( M_k \) is the number of measurements in data set \( k \). The set of hypotheses generated from the \( k - 1 \) earlier data sets is defined by \( \Omega_{k-1} = \{ \Omega_i^{k-1}, i = 1, \ldots, I_{k-1} \} \), where \( I_{k-1} = | \Omega^{k-1} | \) is the number of hypotheses in the set. The data association hypotheses are generated by extending each existing hypothesis, resulting from the \( k - 1 \) data sets, with a compatible association for each measurement in the current data set. The hypothesis \( \Omega_i^k \) can be considered as the hypothesis formed from the hypothesis \( \Omega_i^{k-1} \), which was formed before the information \( Z(k) \) became available, and the association hypothesis \( \psi_h \) for the current data set. This is formally written as \( \Omega_i^k = \{ \Omega_i^{k-1} \cap \psi_h \} \).

To rank the different hypotheses, it is necessary to define a weighting factor for each hypothesis. The importance of the hypothesis \( \Omega_i^k \) is expressed by the conditional probability (density) function \( f_i^k = f(\Omega_i^k \mid Z^k) \). Formally it is now possible to write

\[
f(\Omega_i^k \cap Z^k) = f(\Omega_i^k \mid Z^k) = f(Z(k) \mid \Omega_i^{k-1} \cap \psi_h \cap Z^{k-1}) \times f(\Omega_i^{k-1} \cap Z^{k-1})
\]

The next step is to write

\[
f(\Omega_i^k \cap Z^k) = f(\Omega_i^k \mid Z^k) = f(Z(k) \mid \Omega_i^{k-1} \cap \psi_h \cap Z^{k-1}) \times f(\Omega_i^{k-1} \cap Z^{k-1}) \times f(Z(k) \mid Z^{k-1})
\]

and to substitute eq. 2.32 and 2.33 into eq. 2.31. The last step is to substitute \( \Omega_i^k = \{ \Omega_i^{k-1} \cap \psi_h \} \) and \( Z^k = \{ Z^{k-1} \cap Z(k) \} \). This produces the recursive equation

\[
f(\Omega_i^k \mid Z^k) = f(Z(k) \mid \Omega_i^{k-1} \cap \psi_h \cap Z^{k-1} \cap Z(k)) = \frac{f(Z(k) \mid \Omega_i^{k-1} \cap \psi_h \cap Z^{k-1}) \times f(\psi_h \mid \Omega_i^{k-1} \cap Z^{k-1}) \times f(\Omega_i^{k-1} \mid Z^{k-1})}{f(Z(k) \mid Z^{k-1})}
\]

which corresponds to equation 8 of Reid [107]. Due to the discrete nature of \( \Omega_i^k \) it is clear that \( f(\Omega_i^k \mid Z^k) \) is also a conditional probability.

Reid [107] has produced a result for 2.34 which can be implemented relatively easily. If the prior data association hypothesis is first multiplied by \( (1 - P_D)^{N_{\mathrm{TARGET}}} \), then as a branch is created for each measurement and its hypothesized origin, the conditional probability of the branch is found by either multiplying the prior probability by \( \beta_{\mathrm{FT}}, \beta_{\mathrm{NT}} \) or \( P_D f_m(\vec{z}_k) / (1 - P_D) \), where \( \beta_{\mathrm{NT}} \) is the density of new targets, \( \beta_{\mathrm{FT}} \) is the density of false alarms and \( f_m(\vec{z}_k) \), defined by eq. 2.8, is the measurement probability function for measurement \( \vec{z}_k \in Z(k) \). \( N_{\mathrm{TARGET}} \) is the number of previously known targets in hypothesis \( \Omega_i^{k-1} \) before processing the received data set. The probability that the sensor detects a target is given by \( P_D \).

The way in which the hypotheses are generated clearly contains the risk of a combinatorial explosion in the number of hypotheses. A number of heuristic

---

\(^9\)Each association hypothesis contains track hypothesis-measurement assignments and/or clutter-measurement assignments.
techniques has been developed to control the number of hypotheses ([107], [25]). However, application of those techniques introduces the possibility that the correct hypothesis is removed from the solution space. The simplest technique is to prune all unlikely hypothesis with a probability \( P \leq P_{th} \), where \( P_{th} \) is a specified threshold. A refinement is to derive track probabilities from the surviving hypotheses and to delete the tracks with a probability lower than another specified threshold. This does not apply to tracks which have not been updated during a long time interval, suggesting that the corresponding targets have disappeared. To eliminate this kind of track, tracks that have not been updated for a period of at least a given time interval, are deleted.

Another way to reduce the number of hypotheses is to combine similar hypotheses [107]. Here two approaches are possible. In the first approach the set of hypotheses \( s \) which have the last \( N \) data sets in common are combined by keeping one of the hypotheses with the new probability \( \sum_s P \). The disadvantage of this method is that hypotheses that differentiate between measurements in earlier data sets are eliminated. An alternative way for binding hypotheses together is to combine those hypotheses which are similar. For two hypotheses to be similar, they must have the same number of tracks and for all targets in each hypothesis both the mean and the variance of each estimate must be sufficiently close. The mean covariance of the resulting estimate is a combination of the individual estimates. A method to determine if two tracks correspond with the same target is proposed in [11].

Within a scan, k-best MHT determines the k best hypotheses. Due to the fact that only the k-best hypotheses survive from scan to scan the number of hypotheses which have to be considered is significantly reduced. A disadvantage is that it cannot be guaranteed that the best possible hypothesis is found, reducing the algorithm to an efficient heuristic. Application of this algorithm reduces the multitarget tracking problem from a NP-hard problem to a problem for which polynomial-time algorithms are available (see also chapter 5).

An alternative approach is the track-oriented MHT method [78]. This approach constructs target trees, whose branches contain the track hypotheses having the root node as common ancestor (see also section 5.2, chapter 5). The branches are extended by assigning correlating measurements. The root node is defined as the root track and represents the appearance of a new target or a previously established track hypothesis. The main difference between Reid’s MHT method and the track-oriented MHT method is that the first approach maintains data association hypotheses from scan to scan and uses these hypotheses to directly spawn new hypotheses [25]. On the other hand, the track-oriented approach on each scan reconstructs the data association hypotheses from the track hypotheses, which are maintained in the different target trees.

### 2.7 Comparative studies

All data association methods, presented in this chapter, work well for easy cases. Beyond a certain level of difficulty all methods will lead to poor performance. A number of studies is known in which the performance of the association methods discussed in this chapter is compared. Here several of the more interesting results are presented.

Keuk [123] has generalized the application of MHT tracking for phased-
array radar applications. Single targets in a cluttered environment have been considered. Suitable target revisit and beam positioning algorithms have been derived and optimized regarding the energy spent for tracking. A performance comparison with a standard PDA algorithm (section 2.6.2) has been made. With stronger clutter interference the MHT method (section 2.6.3) requires about a factor 2 to 3 less energy than PDA. To present an impression of the results achieved, fig. 1 and 2 of [123] have been used to produce table 2.2 in

Table 2.2: Comparison MHT and PDA against a single target in clutter derived from [123] in mean number of tracks lost in 1000 runs. The mean number of clutter measurements in the association gate is given by $\bar{n}_f$.

<table>
<thead>
<tr>
<th>$V_0$</th>
<th>$\bar{n}_f$</th>
<th>MHT</th>
<th>PDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>.10</td>
<td>.00</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>.01</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>1</td>
<td>4.3</td>
<td></td>
</tr>
<tr>
<td>.10</td>
<td>1</td>
<td>8.5</td>
<td></td>
</tr>
<tr>
<td>.30</td>
<td>1</td>
<td>17.5</td>
<td></td>
</tr>
</tbody>
</table>

which the mean number of tracks lost per 1000 runs is given as function of the average number $\bar{n}_f$ of clutter measurements in the association gate during the necessary time-on-target for illumination. The required bearing track accuracy is determined by the dimensionless parameter $V_0$ $^{11}$, which expresses the track sharpness relative to the beamwidth of the pencil beam used by the phased array radar to track the target. Tracking in range has not been considered explicitly. The mean number of lost tracks has a pronounced dependence on $V_0$. The conclusion is that MHT is less sensitive to clutter interference.

Koch [76] assumes a 2D scanning radar and compares the tracking performance of MHT (section 2.6.3,2.3) with the JPDA-method (section 2.6.2) in a scenario with two targets flying a tight formation in dense false alarms environments. Both methods use an IMM filter. An illustrative example of the results is given in table 2.3. The results are collected during 1000 Monte Carlo runs. $P_D$ is the probability to detect the target when the radar scans the target, $\rho_F$ is the number of false alarms per $km^2$, TL represents the number of tracks lost during the runs, $\bar{n}_{hyp}$ is the mean number of hypotheses maintained in a scan and $\delta r$ is the mean track accuracy in [km] averaged over the two targets. The results are clear enough. MHT has a better track continuity for all situations considered. For the highest false alarm density values considered, the JPDA tracking could not produce useful results, while the MHT tracker performs (rather) well for all cases.

De Feo et al. [43] have compared the performance of the methods JPDA combined with IMM-filtering (IMM-JPDA) (sections 2.3, 2.6.2 ), track-oriented MHT (section 2.6.3) and GNN (section 2.6) in a variety of radar tracking scenarios. In the track-oriented MHT method, N-scan pruning is achieved by tracing back N scans from the most likely track to establish a new root node [25]. The

$^{10}$Stronger means that the average number of clutter measurements received per dwell is 0.3 instead of for example 0.03.

$^{11}$Assuming a certain $\chi^2$-value, $V_0$ is defined as the ratio between the major axis of the predicted accuracy ellipsoid and the half-beam width of the used tracking pencil beam.
Table 2.3: Comparison MHT and JPDA tracking performance against two targets flying in tight formation derived from Koch [76] in mean number of tracks lost \( TL \), mean number of hypotheses \( \bar{n}_{hyp} \) maintained during a scan and the mean track accuracy \( \delta r \) in [km]. The results are averaged over 1000 Monte Carlo runs.

<table>
<thead>
<tr>
<th>( P_D )</th>
<th>( \rho_f )</th>
<th>( TL )</th>
<th>( \bar{n}_{hyp} )</th>
<th>( \delta r )</th>
<th>( TL )</th>
<th>( \delta r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>.95</td>
<td>.005</td>
<td>3</td>
<td>3.48</td>
<td>0.126</td>
<td>57</td>
<td>0.123</td>
</tr>
<tr>
<td>.1</td>
<td>.01</td>
<td>2</td>
<td>4.11</td>
<td>0.129</td>
<td>89</td>
<td>0.126</td>
</tr>
<tr>
<td>.1</td>
<td>.05</td>
<td>4</td>
<td>7.51</td>
<td>0.158</td>
<td>no result</td>
<td></td>
</tr>
<tr>
<td>.80</td>
<td>.002</td>
<td>1</td>
<td>4.63</td>
<td>0.160</td>
<td>96</td>
<td>0.155</td>
</tr>
<tr>
<td>.004</td>
<td>.04</td>
<td>3</td>
<td>5.34</td>
<td>0.165</td>
<td>131</td>
<td>0.159</td>
</tr>
<tr>
<td>.80</td>
<td>.0005</td>
<td>45</td>
<td>9.63</td>
<td>0.204</td>
<td>no result</td>
<td></td>
</tr>
<tr>
<td>.60</td>
<td>.005</td>
<td>3</td>
<td>5.30</td>
<td>0.275</td>
<td>139</td>
<td>0.313</td>
</tr>
<tr>
<td>.001</td>
<td>4</td>
<td>6.29</td>
<td>0.291</td>
<td>174</td>
<td>0.334</td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>26</td>
<td>9.75</td>
<td>0.334</td>
<td>no result</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IMM-JPDA method considers two or three different dynamical target models. For a number of easier cases all three methods have a comparable performance. For the more difficult cases considered, the results clearly show the advantages of MHT over the other two methods. As an example, consider the results for a scenario containing two targets joining and diverging. To produce the results 100 Monte Carlo runs have been made. Global nearest neighbor (GNN) reacts rather slowly on the change of course occurring when the targets join. The tendency is to maintain the course of both targets, so that the probability of a final hypothesis in which the targets cross is very high. The multi-scan algorithms MHT and IMM-JPDA can reproduce the manoeuvres much better and are often able to resolve the scenario successfully. In table 2.4 estimated target crossing probabilities are presented for two different values of the detection probability \( P_D \). For MHT two pruning policies are considered: 1-scan back and 2-scans back. Preliminary tests indicate that IMM-JPDA and one-backscan MHT require similar processing requirement, while these requirements halve for the GNN solution and roughly double for the two-backscan MHT.

<table>
<thead>
<tr>
<th></th>
<th>( P_D = 1 )</th>
<th>( P_D = 0.8 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMM-JPDA (2 models)</td>
<td>0.29</td>
<td>0.58</td>
</tr>
<tr>
<td>IMM-JPDA (3 models)</td>
<td>0.37</td>
<td>0.60</td>
</tr>
<tr>
<td>GNN</td>
<td>0.94</td>
<td>0.98</td>
</tr>
<tr>
<td>MHT (1-backscan)</td>
<td>0.12</td>
<td>0.31</td>
</tr>
<tr>
<td>MHT (2-backscans)</td>
<td>0.11</td>
<td>0.28</td>
</tr>
</tbody>
</table>
2.8 Summary and conclusions

This chapter started with discussing elementary concepts concerning sensor systems (section 2.1), target tracking and measurement processing (section 2.2). In section 2.2 the typical multitarget tracking system has been introduced.

Section 2.3 introduced the recursive extended first order Kalman filter which is commonly used for target state estimation. Due to the fact that the real target trajectory is unknown and has a high probability of deviating from a linear trajectory, an interacting multiple model (IMM) approach for radar tracking systems is often necessary.

Gating was introduced in section 2.4. The objective of gating is to associate measurements with the appropriate tracks under the condition that there can be missing reports, unknown targets and false reports. To overcome the high computational cost of the proposed gating method, a pre-gating method was proposed in this section. The pre-gating method is used to remove the most unlikely association possibilities before the expensive gating method is applied. The advantage is a significant reduction in the necessary computation time.

The problem of processing out-of-sequence measurements has been considered in section 2.5. After identifying a number of filtering approaches in literature to process out-of-sequence measurements, it was shown that each of the approaches had an undesirable shortcoming. In section 2.5.2, an alternative method for processing out-of-sequence measurements has been proposed, with none of the earlier identified shortcomings. This method is based on predicting the out-of-sequence measurement and its probability distribution function to the time that it was actually received. It has been shown in section 2.5.2.2, that this new approach produces near-optimal results for the time interval in which it is useful to process those measurements.

A major objective of the research described in this thesis is to carry out research to develop a new distributed data fusion architecture which has to operate in environments with high false alarm rates and complex scenarios with tight formations (section 1.6). With respect to filtering, it is clear that modeling the real target trajectory with a single model is often inappropriate, implying that an IMM approach or the possibility to switch between different models is necessary. The conclusion is, that for this kind of environments IMM-filtering has to be used.

After discussion of the data association problem in section 2.6, four of the most commonly used data association methods were introduced. Section 2.6.1 discussed the global nearest neighbor (GNN) approach, section 2.6.2 the all-neighbors data association approach (PDA, JPDA) and section 2.6.3 the Multiple Hypothesis Tracking (MHT) approach. A performance comparison of the discussed methods has been carried out in section 2.7, illustrated with example results. For a single target flying in (dense) clutter, the results presented in table 2.2 clearly show that the PDA averaging approach to produce tracks (which may contain contributions from more than one measurement) does not perform as well as MHT. For two targets flying in tight formation in a dense clutter environment, the results in table 2.4 show that this conclusion is also true with respect to JPDA. A possibility is to enhance JPDA with IMM filtering or to use GNN instead. For two targets which are joining and then diverging again, table 2.4 clearly shows that MHT with the possibility to switch between target models has the best performance. From the results, enumerated in the different
articles considered and supported by other literature [25], the conclusion can be
drawn that Multiple Hypothesis Tracking offers the best performance in com-
plicated scenarios under heavy clutter conditions. Based on preliminary results
given in [43] the conclusion can be drawn that in terms of computation time
GNN is the most efficient, JPDA is less efficient and MHT ranks third as least
efficient. Based on the presented results, it is concluded that MHT combined
with IMM-filtering is the preferred algorithm to produce a SIAP.