Data Association

“Data association is the process of associating uncertain measurements to known tracks.”

- **Problem types**
  - Track creation, maintenance, and deletion
  - Single or multiple sensors
  - Target detection
  - False alarm model and rates
  - Single or multiple targets

- **Approaches**
  - **Bayesian:** compute a full (or approx.) distribution in DA space from priors, posterior beliefs, and observations
  - **Non-Bayesian:** compute a maximum likelihood estimate from the possible set of DA solutions
Data Association

Overall procedure:

- **Make observations** (= measurements).
  Measurements can be raw data (e.g. processed radar signals) or the output of some target detector (e.g. people detector)

- **Predict the measurements** from the predicted tracks. This yields an area in sensor space where to expect an observation. The area is called **validation gate** and is used to narrow the search

- **Check if a measurement lies in the gate.**
  If yes, then it is a valid candidate for a pairing/match
Data Association

What makes this a difficult problem

- **Multiple targets**
- **False alarms**
- **Detection uncertainty** (occlusions, sensor failures, ...)
- **Ambiguities** (several measurements in the gate)
Measurement Prediction

- Measurement and measurement cov. prediction
  - This is typically a frame transformation into sensor space
    \[
    \hat{z}(k) = H(k)\hat{x}(k|k - 1)
    \]
    \[
    \hat{R}(k) = H(k)\hat{P}(k|k - 1)H^T(k)
    \]
  - If only the position of the target is observed (typical case), the measurement matrix is
    \[
    z = [x \ y]^T \quad H = \begin{bmatrix} 1 & 0 & 0 & \cdots \end{bmatrix}
    \]
  - Note: One can also observe
    - Velocity (Doppler radar)
    - Acceleration (accelerometers)
Validation Gate

- Assume that measurements are distributed according to a Gaussian, centered at the measurement prediction $\hat{z}(k)$ with covariance $\hat{S}(k)$

$$p(z(k)) = \mathcal{N}(z(k); \hat{z}(k), \hat{S}(k))$$

This is the **measurement likelihood model**

- Let further

$$d = \sqrt{(x - \mu)^T C^{-1} (x - \mu)}$$

be the **Mahalanobis distance** between $x$ and $\mu$
**Validation Gate**

- Then, the measurements will be in the area
  \[
  \mathcal{V}(k, \gamma) = \{ z : (z - \bar{z})^T \hat{S}^{-1}(z - \bar{z}) \leq \gamma \}
  = \{ z : d^2 \leq \gamma \}
  \]
  with a probability defined by the gate threshold \( \gamma \) (omitting indices \( k \))

- This area is called **validation gate**

- The threshold is obtained from the inverse \( \chi^2 \) cumulative distribution at a **significance level** \( \alpha \)

- \( \chi^2 = \text{“chi square”} \)
Validation Gate

- The **shape** of the validation gate is a hyper-ellipsoid (an ellipse in 2d)
- This follows from setting

\[
c = \frac{1}{(2\pi)^{k/2} |S|^{1/2}} \exp \left( -\frac{1}{2} (z - \tilde{z})^T S^{-1} (z - \tilde{z}) \right)
\]

which gives

\[
c' = (z - \tilde{z})^T S^{-1} (z - \tilde{z})
\]

- The gate is defined by an **iso-probability contour** obtained when intersecting a Gaussian with a hyper-plane.
Validation Gate

Why a $\chi^2$ distribution?

- Let $X_i$ be a set of $k$ i.i.d. standard normally distributed random variables, $X_i \sim \mathcal{N}(x; 0, 1)$. Then, the variable $Q$

$$Q = \sum_{i=1}^{k} X_i^2$$

follows a $\chi^2$ distribution with $k$ “degrees of freedom”

- We will now show that the Mahalanobis distance is a sum of squared standard normally distributed RVs.
Validation Gate in 1D

- Assume 1D measurements and $\mu = \hat{z}(k)$, $\sigma^2 = \hat{S}(k)$
- The Mahalanobis distance is then

$$d^2 = (z - \mu)^T (\sigma^2)^{-1} (z - \mu) = \frac{(z - \mu)^2}{\sigma^2}$$

- By changing variables, $y = (z - \mu)/\sigma$, we have

$$y \sim \mathcal{N}(0, 1)$$

- Thus, $d^2 = y^2$ and is $\chi^2$ distributed with 1 degree of freedom
Validation Gate in ND

- Assume ND measurements and \( \mu = \hat{z}(k), \Sigma = \hat{S}(k) \)
- The Mahalanobis distance is then
  \[
  d^2 = (z - \mu)^T \Sigma^{-1} (z - \mu)
  \]
- By changing variables, \( y = C^{-1}(z - \mu), \Sigma = CC^T \) we have \( y \sim \mathcal{N}(0, I) \) and therefore
  \[
  d^2 = y^T I^{-1} y \quad \Rightarrow \quad d^2 = \sum_{i=1}^{k} y_i^2
  \]
  which is \( \chi^2 \) distributed with \( k \) degrees of freedom.
  \( (C \text{ is obtained from a Cholesky decomposition}) \)
Validation Gate

Where does the threshold \( \gamma \) come from?

- \( \gamma \), often denoted \( \chi^2_{k,\alpha} \), is taken from the inverse \( \chi^2 \) cumulative distribution at a level \( \alpha \) and \( k \) d.o.f.s

- The values are typically given in tables, e.g. in most statistics books (or by the Matlab function \( \text{chi2inv} \))

- Given the level \( \alpha \), we can now understand the interpretation of the validation gate:

The validation gate is a region of acceptance such that \( 100(1 - \alpha)\% \) of true measurements are rejected

- Typical values for \( \alpha \) are 0.95 or 0.99
Validation Gate

Euclidian distance

Takes into account:
✓ Position
✗ Uncertainty
✗ Correlations

→ It seems that i-a and j-b belong together
Validation Gate

**Mahalanobis distance** with **diagonal** covariance matrices

Takes into account:
- ✓ Position
- ✓ Uncertainty
- ✗ Correlations

→ Now, i-b is “closer” than j-b
Validation Gate

Mahalanobis distance

Takes into account:
✓ Position
✓ Uncertainty
✓ Correlations

→ It’s actually i-b and j-a that belong together!
False Alarms

- False alarms are **false positives**
- They can come from sensor imperfections or detector failures
- They raise the two questions:
  
  What is actually inside my **validation gate**?
  - The real measurement or
  - a false alarm?

How to **model false alarms**?
- Uniform over sensor space
- Independent across time
False Alarm Model

- Assume (temporarily) that the sensor field of view \( V \) is discretized into \( N \) discrete cells, \( c_i, \ i = 1, \ldots, N \).
- In each cell, false alarms occur with probability \( P_F \).
- Assume independence across cells.
- The occurrence of false alarms is a Bernoulli process (flipping an unfair coin) with probability \( p = P_F \).
- Then, the number of false alarms \( m_F \) follows a Binomial distribution:

\[
P(K = m_F) = \binom{N}{m_F} p^{m_F} (1 - p)^{N-m_F}
\]

with expected value \( Np \).
False Alarm Model

- Let the spatial density $\lambda$ be the number of false alarms over space
  \[ \lambda = \frac{Np}{V} \]  
  [occurrences per m$^2$]

- Let now $N \rightarrow \infty$, that is, we reduce the cell size until the continuous case. Then the Binomial becomes a Poisson distribution with
  \[ \mu_F(m_F) = e^{-\lambda V} \frac{(\lambda V)^{m_F}}{m_F!} \]

- The measurement likelihood of false alarms is assumed to be uniform,
  \[ p(z|z \text{ is a false alarm}) = \frac{1}{V} \]
Single Target Data Association

Let us consider

- A **single** target to track
- Track already initialized
- Detection probability < 1
- False alarm probability > 0

Data association approaches

Non-Bayesian:

- Nearest neighbor (NN)
- Track splitting filter

Bayesian:

- Probabilistic Data Association Filter (PDAF)
Single Target DA: NN

Nearest Neighbor filter \((\text{NN})\)

1. Compute Mahalanobis distance to all measurements
2. Accept the closest measurement
3. Update the track as if it were the correct measurement

- Problem: with some probability the selected measurement is not the correct one. This can lead to filter divergence (covariances collapse regardless)

- Conservative NN variant:
  Do not associate in case of ambiguities
Single Target DA: PDAF

Probabilistic Data Association filter (PDAF)

- Integrates all measurements in the validation gate
  - Conditioning the update on

\[ \theta_i(k) = \begin{cases} 
    z_i(k) \text{ is the correct measurement} & i = 1, \ldots, m(k) \\
    \text{no correct measurement is present} & i = 0 
\end{cases} \]

- With probability \( \beta_i \triangleq P(\theta_i|Z^k) \) for the Poisson case

\[ \beta_i(k) = \begin{cases} 
    \frac{e_i}{b + \sum_{j=1}^{m_F} e_j} & i = 1, \ldots, m(k) \\
    \frac{b}{b + \sum_{j=1}^{m_F}} & i = 0 
\end{cases} \]

\[ e_i = \mu_F (m(k) - 1) \cdot P_D P_G \cdot P_G^{-1} N(\nu_i(k); 0, \hat{S}(k)) \]

\[ b = \mu_F (m(k))(1 - P_D P_G) \]
Single Target DA: PDAF

- Uses all the measurements in the validation area
  - Conditioning the update on

\[
\theta_i(k) = \begin{cases} 
  z_i(k) \text{ is the correct measurement} & i = 1, \ldots, m(k) \\
  \text{no correct measurement is present} & i = 0
\end{cases}
\]

- With probability \( \beta_i \triangleq P(\theta_i | Z^k) \) for the Poisson case

\[
\beta_i(k) = \begin{cases} 
  \frac{e_i}{b + \sum_{j=1}^{m(k)}} & i = 1, \ldots, m(k) \\
  \frac{b}{b + \sum_{j=1}^{m(k)}} & i = 0
\end{cases}
\]

\[
e_i = \mu_F(m(k) - 1) \cdot P_D P_G \cdot P_G^{-1} \mathcal{N}(\nu_i(k); 0, \hat{S}(k))
\]

\[
b = \mu_F(m(k))(1 - P_D P_G)
\]
Single Target DA: PDAF

- State update

\[ \hat{x}(k|k) = \hat{x}(k|k - 1) + K(k)\nu(k) \]

- With the combined innovation

\[ \nu(k) = \sum_{i=1}^{N} \beta_i(k)\nu_i(k) \]

- Covariance update

\[ P(k|k) = \beta_0(k)P(k|k - 1) + (1 - \beta_0(k))P(k|k) + \tilde{P}(k) \]

- With the spread of innovations

\[ \tilde{P}(k) = K(k) \left[ \sum_{i=1}^{N} \beta_i(k)\nu_i(k)\nu_i(k)^T - \nu(k)\nu(k)^T \right] K(k)^T \]
Single Target DA: Summary

- **Nearest Neighbor filter** (NN)
  - Simple to implement
  - Can integrate wrong measurements (false alarms), and thus, produce overconfident estimates
  - Good if prediction and measurement models are accurate

- **Probabilistic Data Association filter** (PDAF)
  - A bit more involved to implement
  - Provide conservative estimates
  - Good in presence of high clutter and noisy models
Multi-Target Data Association

- Multiple targets to track
  - Tracks already initialized
  - Detection probability < 1
  - False alarm probability > 0

- Non Bayesian approaches
  - Nearest neighbor
  - Interpretation tree
  - Joint compatibility (JCBB)

- Bayesian approaches
  - JPDAF
  - MHT
  - MCMC
Multi-Target DA: NN

- Build the assignment matrix $A = [d_{ij}^2]$

  $$d_{ij}^2 = \nu_{ij}(k)^T S_j^{-1}(k) \nu_{ij}(k)$$

- Iterate
  - Find the minimum cost assignment in $A$
  - Remove the row and column of that assignment

- Check if assignment are in the validation regions
  - Unassociated tracks can be used for track deletion
  - Unassociated measurements can be used for track creation

- Problem: It’s not a global minimum
- Conservative NN variant: no association in case of ambiguities
Multi-Target DA: Global NN

- Build the assignment matrix \[ A = [d_{ij}^2] \]
  \[ d_{ij}^2 = \nu_{ij}(k)^T S_j^{-1}(k) \nu_{ij}(k) \]
- Solve the linear assignment problem
  \[ \min \sum_{i} d_{ij}^2 \cdot x_{ij} \quad x_{ij} \in \{0, 1\} \]
  \[ \sum_{i} x_{ij} = 1 \quad \sum_{j} x_{ij} = 1 \]
  - **Hungarian** method (blow up to square matrix)
  - **Munkres** algorithm for rectangular matrices
  - Finds *global* cost minimum!
- Check if assignments are in the validation gate
- Performs DA jointly!
Assignment Matrix Example

- **Rectangular**

\[
A = \begin{bmatrix}
    d_{11}^2 & d_{12}^2 & d_{13}^2 & d_{14}^2 & d_{15}^2 & d_{16}^2 & d_{17}^2 & d_{18}^2 \\
    d_{21}^2 & d_{22}^2 & d_{23}^2 & d_{24}^2 & d_{25}^2 & d_{26}^2 & d_{27}^2 & d_{28}^2 \\
\end{bmatrix}
\]

- **Square**

\[
A = \begin{bmatrix}
    d_{11}^2 & d_{12}^2 & d_{13}^2 & d_{14}^2 & d_{15}^2 & d_{16}^2 & d_{17}^2 & d_{18}^2 \\
    d_{21}^2 & d_{22}^2 & d_{23}^2 & d_{24}^2 & d_{25}^2 & d_{26}^2 & d_{27}^2 & d_{28}^2 \\
    p_F & p_F & p_F & p_F & p_F & p_F & p_F & p_F \\
    p_F & p_F & p_F & p_F & p_F & p_F & p_F & p_F \\
    p_F & p_F & p_F & p_F & p_F & p_F & p_F & p_F \\
    p_F & p_F & p_F & p_F & p_F & p_F & p_F & p_F \\
    p_F & p_F & p_F & p_F & p_F & p_F & p_F & p_F \\
    p_F & p_F & p_F & p_F & p_F & p_F & p_F & p_F \\
\end{bmatrix}
\]

- **Entries**
  - \( d_{ij}^2 = \nu_{ij}(k)^T S_j^{-1}(k) \nu_{ij}(k) \)
  - False alarm probability
GNN vs. Interpretation Tree

- A solution to GNN in presence of constraints
  - Introduced in [Grimson 87], used in feature-based SLAM

- Main idea: consider all possible association among measurements and tracks
  - The association are built over a tree representation
  - A Depth-first visit on the tree is performed
  - Constraints are used to prune the tree

- Worst case: exponential complexity
  - The whole number of associations is $(\#T + 1)^\#m$
Interpretation Tree: SLAM

Interpretation tree

\[ S_{h_2} = \{ \{l_1, g_3\}, \{l_2, g_7\}, \{l_3, g_2\} \} \]
Interpretation Tree: SLAM

\[ S_h = \{ \{ l_1, g_4 \}, \{ l_2, g_8 \}, \{ l_3, \ast \} \} \]
GNN Comparison

Assignment solver

- Pros
  - Fast: polynomial
  - Libraries available
  - Extension to k-best

- Cons
  - Only linear constraints
  - Blow-up of cost matrix

Interpretation tree

- Pros
  - General constraints
  - Extensive search
  - Provide k-best

- Cons
  - Slow: exponential

Interpretation trees are good when several constraints are available, in other cases assignment solver are preferable
Joint Compatibility

- Individual compatibility (e.g. independent tracks)
  - One measurement integration influences only one track
  - One measurement per track assumption
  - Typical of target tracking

- Joint compatibility (e.g. correlated tracks)
  - One measurement integration influences several tracks
  - Multiple measurement per track association
  - Typical of localization and SLAM
Joint Compatibility

- Given the joint hypothesis $\mathcal{H} = \{j_1, \ldots, j_m\}$
  - the k-th measurement is associated with track $j_k = t$

- And the joint measurement function

  $$z_{\mathcal{H}} = h_{\mathcal{H}}(x) + w$$

  $$h_{\mathcal{H}} = \begin{bmatrix} h_{i,j_1} \\ \vdots \\ h_{i,j_m} \end{bmatrix}$$

- The hypothesis is jointly compatible if

  $$D_{\mathcal{H}}^2 = (z_{\mathcal{H}} - h_{\mathcal{H}}(\hat{x}))^T C_{\mathcal{H}}^{-1} (z_{\mathcal{H}} - h_{\mathcal{H}}(\hat{x})) \leq \chi^2_{d, \alpha}$$

  $$C_{\mathcal{H}} = H_{\mathcal{H}} \hat{P} H_{\mathcal{H}}^T + R_{\mathcal{H}}$$
Joint Compatibility – JCBB

- Joint compatibility branch and bound
- Initialize with empty hypothesis and first obs.

- For all tracks
  - If is individually and jointly compatible assign and recursively call JCBB
  - Otherwise consider it a false alarm

procedure JCBB (H, i):
-- H : current hypothesis
-- i : observation to be matched
if i > m
  if pairings(H) > pairings(Best)
    Best = H
  fi
else
  for j in {1...n}
    if unary(i, j)
      and joint_compatibility([H j])
        JCBB([H j], i + 1)
      fi
  rof
  if pairings(H) + m - i > pairings(Best)
    JCBB([H 0], i + 1)
  fi
fi

[Neira et al.’03]
Multi-Target DA: MHT

- Reason about the associations of sequences measurements with tracks and false alarm
- Evaluate the probability of association hypotheses
- Optimal Bayesian solution

- Algorithm
  - State and measurement prediction
  - Hypotheses generation
  - Hypotheses probability evaluation
  - State update
  - Hypotheses management (i.e. pruning, elimination, creation)

- Exponential complexity of the full solution
  - Pruning strategies
  - K-best hypotheses
MHT: Hypothesis Generation

- A hypothesis $\Theta_i^k = \{\Theta_{p(i)}^{k-1}, \Theta_{c(i)}(k)\}$ at time $k$ is a history of assignment sets $\theta = \{\theta_{obs}, \theta_{track}\}$ to time $k$
  - $\theta_{obs} = \{z_1, \ldots, z_{m(k)}\}$ is a set of measurement associations, where a measurement is either associated to track $z_i = t$, treated as a new track $z_i = n$, or as a false alarm $z_i = f$
  - $\theta_{track} = \{l_1, \ldots, l_{t(k)}\}$ is a set of track label, where a track can be matched $l_i = m$, occluded $l_i = o$, or deleted $l_i = d$

- Hypotheses are generated recursively in a tree-based structure
  - Unlikely branch are avoided by validation gating
  - Exponential growth of the trees
  - Only a subset of hypotheses are generated in practice
MHT: Hypothesis Generation

\[ \Theta^{k-1}_{p(i)} \]
MHT: Hypothesis Generation

Considering $z_1$
MHT: Hypothesis Generation

Considering $z_2$
MHT: Hypothesis Generation

Considering $t_1$

\[ \Theta^{k-1}_{p(i)} \]

\[ z_1 = f \quad z_1 = n \quad z_1 = t_2 \quad z_1 = t_1 \]

\[ z_2 = f \quad z_2 = n \]

\[ t_1 = d \quad t_1 = o \]

\[ t_2 \]

\[ z_2 = f \quad z_2 = n \]

\[ t_1 = d \quad t_1 = o \]

\[ t_1 = m \]
MHT: Hypothesis Generation

Considering $t_2$

$\Theta^{k-1}_{p(i)}$

$z_1 = f$

$z_1 = n$

$z_1 = t_1$

$z_2 = f$

$z_2 = n$

$z_2 = t_2$

$z_2 = t_1$

$t_1 = d$

$t_1 = o$

$t_2 = d$

$t_2 = o$

$t_1 = d$

$t_1 = o$

$t_2 = d$

$t_2 = o$

$t_1 = d$

$t_1 = o$

$t_2 = d$

$t_2 = o$

$t_1 = m$

$t_1 = o$

$t_1 = m$

$t_2 = m$

$t_2 = d$

$t_2 = o$

$t_2 = d$

$t_2 = d$

$t_2 = o$

$t_2 = o$

$t_2 = o$

$t_2 = o$

$t_2 = o$
MHT: Hypothesis Evaluation

- The probability of an hypothesis \( \Theta_i^k = \{\Theta_{p(i)}^{k-1}, \theta_{c(i)}(k)\} \) can be calculated using Bayes rules:

\[
P(\Theta_i^k | Z^k) = P(\Theta_{p(i)}^{k-1}, \theta_{c(i)}(k) | Z^k) = \frac{1}{\eta} \cdot p(Z(k) | \Theta_{p(i)}^{k-1}, \theta_{c(i)}(k), Z^{k-1}) \cdot P(\theta_{c(i)}(k) | \Theta_{p(i)}^{k-1}, Z^k) \cdot P(\Theta_{p(i)}^{k-1} | Z^{k-1})
\]

Likelihood \hspace{2cm} Assignment probability \hspace{2cm} Prior
MHT: Hypothesis Evaluation

- Likelihood
  
  \[ p(Z(k)|\Theta^{k-1}, \theta(k), Z^{k-1}) = \prod_{l=1}^{m(k)} p(z_l(k)|\Theta^{k-1}, \theta(k), Z^{k-1}) \]

  - Case 1: associated with track t
    
    \[ p(z_l(k)|\Theta^{k-1}, \theta(k), Z^{k-1}) = \mathcal{N}(z_l(k); \hat{z}_t(k|k-1), S_t(k)) \]

  - Case 2: false alarm
    
    \[ p(z_l(k)|\Theta^{k-1}, \theta(k), Z^{k-1}) = V^{-1} \]

  - Case 3: new track
    
    \[ p(z_l(k)|\Theta^{k-1}, \theta(k), Z^{k-1}) = V^{-1} \]
MHT: Hypothesis Evaluation

- Assignment probability

\[ P(\theta(k)|\Theta^{k-1}, Z^k) = P(\theta(k)|N_M, N_O, N_D, N_N, N_F) \cdot P(N_M, N_O, N_D, N_N, N_F|\theta(k), \Theta^{k-1}) \]

- \( P(N_M, N_O, N_D, N_N, N_F|\theta(k), \Theta^{k-1}) \) is the probability of having \( N_M \) matched tracks, \( N_O \) occluded tracks, \( N_D \) deleted tracks, \( N_F \) false alarm and \( N_N \) new tracks.

- \( P(\theta(k)|N_M, N_O, N_D, N_N, N_F) \) is the probability of a possible configuration \( \theta(k) \) given the number of events defined before.
MHT: Hypothesis Evaluation

- **Assignment probability 1:** 
  \[ P(N_M, N_O, N_D, N_N, N_F | \theta(k), \Theta^{k-1}) \]
  - Assuming a multinomial distribution for track labels
    \[ P(N_M, N_O, N_D | \theta(k), \Theta^{k-1}) = \frac{N_T!}{N_M!N_O!N_D!} p_M^{N_M} p_O^{N_O} p_D^{N_D} \]
  - Assuming a Poisson distribution for new tracks
    \[ P(N_N | \theta(k), \Theta^{k-1}) = \frac{(V\lambda_N)^{N_N} e^{-V\lambda_N}}{N_N!} \]
  - Assuming a Poisson distribution for false alarm
    \[ P(N_F | \theta(k), \Theta^{k-1}) = \frac{(V\lambda_F)^{N_F} e^{-V\lambda_F}}{N_F!} \]
  - We obtain
    \[ P(\cdot) = \frac{N_T!(e^{-V\lambda_N})(e^{-V\lambda_F})}{N_N!N_F!N_M!N_O!N_D!} (V\lambda_N)^{N_N} (V\lambda_F)^{N_F} p_M^{N_M} p_O^{N_O} p_D^{N_D} \]
MHT: Hypothesis Evaluation

- Assignment probability 2: \( P(\theta(k)|N_M, N_O, N_D, N_N, N_F) \)
  - The possible choices of \( \binom{m(k)}{N_M} \) taken as matched tracks
    \[
    \binom{m(k)}{N_M} \text{Perm}(N_M, N_T) = \frac{m(k)!}{N_M!(m(k) - N_M)!} \frac{N_T!}{(N_T - N_M)!}
    \]
  - The combinations of \( \binom{m(k) - N_M}{N_N} \) \( \binom{m(k) - N_M - N_N}{N_F} = 1 \) taken as new tracks or false alarms
    \[
    \binom{m(k) - N_M}{N_N} \binom{m(k) - N_M - N_N}{N_F} = \frac{(m(k) - N_M)!}{N_N!(m(k) - N_M - N_N)!}
    \]
  - The combinations of \( \binom{N_T - N_M}{N_O} \) \( \binom{N_T - N_M - N_O}{N_D} = 1 \) taken as occluded or deleted
    \[
    \binom{N_T - N_M}{N_O} \binom{N_T - N_M - N_O}{N_D} = \frac{(N_T - N_M)!}{N_O!(N_T - N_M - N_O)!}
    \]
  - The probability is 1 over all the possible choices
    \[
    \left[ \frac{m(k)!}{N_M!(m(k) - N_M)!} \frac{N_T!}{(N_T - N_M)!} \frac{(m(k) - N_M)!}{N_N!(m(k) - N_M - N_N)!} \frac{(N_T - N_M)!}{N_O!(N_T - N_M - N_O)!} \right]^{-1}
    \]
MHT: Hypothesis Evaluation

- Assignment probability 2: \( P(\theta(k)|N_M, N_O, N_D, N_N, N_F) \)
  - The possible choices of \( m(k) \) taken as matched tracks
    \[
    \binom{m(k)}{N_M} \cdot \text{Perm}(N_M, N_T) = \frac{m(k)!}{N_M!(m(k) - N_M)! \cdot (N_T - N_M)!}
    \]
  - The combinations of \( m(k) - N_M \) taken as new tracks or false alarms
    \[
    \binom{m(k) - N_M}{N_N} \cdot \binom{m(k) - N_M - N_N}{N_F} = \frac{(m(k) - N_M)!}{N_N!(m(k) - N_M - N_N)!}
    \]
  - The combinations of \( N_T - N_M \) taken as occluded or deleted
    \[
    \binom{N_T - N_M}{N_O} \cdot \binom{N_T - N_M - N_O}{N_D} = \frac{(N_T - N_M)!}{N_O!(N_T - N_M - N_O)!}
    \]
  - The probability is 1 over all the possible choices

\[
\left[ \frac{m(k)!}{N_M!(m(k) - N_M)!} \cdot \frac{N_T!}{(N_T - N_M)!} \cdot \frac{(m(k) - N_M)!}{N_N!} \cdot \frac{(N_T - N_M - N_N)!}{N_D!} \cdot \frac{(N_T - N_M)!}{N_O!} \cdot \frac{(N_T - N_M - N_O)!}{N_F!} \right]^{-1}
\]
MHT: Hypothesis Evaluation

- Assignment probability 2:  
  \[ P(\theta(k)|N_M, N_O, N_D, N_N, N_F) \]
  - The possible choices of \( m(k) \) taken as matched tracks
    \[ \binom{m(k)}{N_M} \text{Perm}(N_M, N_T) = \frac{m(k)! N_T!}{N_M!(m(k) - N_M)! (N_T - N_M)!} \]
  - The combinations of \( N_T - N_M \) taken as new tracks or false alarms
    \[ \binom{N_T - N_M}{N_O} \binom{m(k) - N_M}{N_N} = \frac{(m(k) - N_M)!}{N_N!(m(k) - N_M - N_N)!} \]
  - The combinations of \( N_T - N_M \) taken as occluded or deleted
    \[ \binom{N_T - N_M}{N_O} \binom{N_T - N_M}{N_D} = \frac{(N_T - N_M)!}{N_O!(N_T - N_M - N_O)!} \]
  - The probability is 1 over all the possible choices
    \[ P(\theta(k)|N_M, N_O, N_D, N_N, N_F) = \frac{N_M!N_N!N_F!N_O!N_D!}{m(k)!N_T!} \]
MHT: Hypothesis Evaluation

- Assignment probability

\[ P(\theta(k)|\Theta^{k-1}, Z^k) = P(\theta(k)|N_M, N_O, N_D, N_N, N_F) \cdot P(N_M, N_O, N_D, N_N, N_F|\theta(k), \Theta^{k-1}) \]

- Combining everything together we have

\[
P(\theta(k)|\Theta^{k-1}, Z^k) = \frac{N_T!(e^{-V\lambda_N})(e^{-V\lambda_F})}{N_N!N_F!N_M!N_O!N_D!}(V\lambda_N)^{N_N}(V\lambda_F)^{N_F}p_M^{N_M}p_O^{N_O}p_D^{N_D}\frac{N_M!N_N!N_F!N_O!N_D!}{m(k)!N_T!}
\]
MHT: Hypothesis Evaluation

- Assignment probability

\[ P(\theta(k) | \Theta^{k-1}, Z^k) = P(\theta(k) \mid N_M, N_O, N_D, N_N, N_F) \cdot P(N_M, N_O, N_D, N_N, N_F | \theta(k), \Theta^{k-1}) \]

- Combining everything together we have

\[
P(\theta(k) | \Theta^{k-1}, Z^k) = \frac{N_T! (e^{-V\lambda_N})(e^{-V\lambda_F})}{N_M!N_O!N_D!} (V\lambda_N)^{N_N} (V\lambda_F)^{N_F} \frac{p_M^{N_M} p_O^{N_O} p_D^{N_D}}{m(k)!} \frac{N_M!N_N!N_D!}{N_T!} \]

- Simplifying the expression we obtain

\[
P(\theta(k) | \Theta^{k-1}, Z^k) = \frac{(e^{-V\lambda_N})(e^{-V\lambda_F})}{m(k)!} (V\lambda_N)^{N_N} (V\lambda_F)^{N_F} p_M^{N_M} p_O^{N_O} p_D^{N_D} \]
MHT Approximations

- Spatially disjoint hypothesis trees
  - Tracks can be partitioned in clusters
  - A separate tree is grown for each cluster

- K-Best hypothesis tree
  - Directly generate the k-best hypothesis
  - Generation and evaluation are integrated in a single step
  - Use Murty algorithm and a liner assignment solver

- N-Scan back pruning
  - Ambiguities are supposed to be resolved after N steps
  - Children at step k+N give the prob. of parents at step k
  - Keep only the most probable branch
MHT Example
MHT Example (Detail)
Multi-Target DA: Summary

- **Nearest Neighbor filters** (NN and GNN)
  - Simple to implement
  - NN: Good if tracks are well separated and not noisy
  - NN+GNN: No integration over time

- **Interpretation tree**
  - More involved to implement
  - Good in case of general constraints among associations

- **MHT**
  - Fully Bayesian
  - Most general framework for multiple targets
  - Complex and expensive
  - Only approximations are practically implemented