Target Tracking
Le 5: Multi-Target Tracking: multi-hypothesis tracking

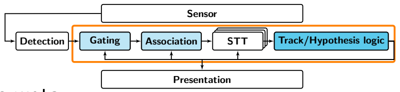
Gustaf Hendeby and Rickard Karlsson

Div. Automatic Control
Dept. Electrical Engineering
gustaf.hendeby@liu.se,
rickard.g.karlsson@liu.se

LINKÖPING UNIVERSITY

LINKÖPING UNIVERSITY

Summary: lecture 4

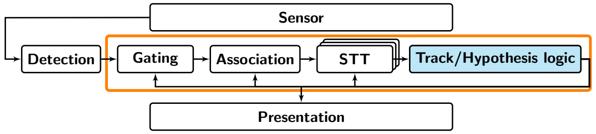


- Extended previous methods to several targets.
- Methods for gating, clustering, and association were presented, yielding the validation and association matrix.
- SHT: *One* measurement association hypothesis is used
  - GNN: A hard decision; choose the most likely association hypothesis.
    The association problem can be solved with many of-the-shelf algorithms, e.g., auction, after constructing the association (cost) matrix.
  - JPDA: A soft decision; marginalize all possible associations.
    How to combine the possible measurements depends on the association matrix.

References on Multiple Target Tracking Topics

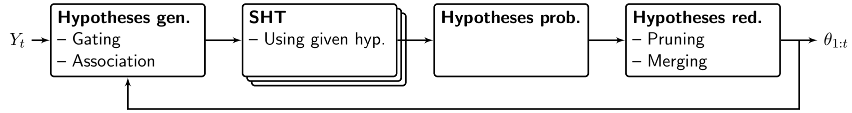
- S. Blackman. Multiple hypothesis tracking for multiple target tracking.
  IEEE Transactions on Aerospace and Electronic Systems, 19(1):5–18, 2004 (MHT overview)
- D. Reid. An algorithm for tracking multiple targets.
  IEEE Transactions on Automatic Control, 24(6):843–854, Dec. 1979 (MHT concept)
- C. Chong, S. Mori, and D. Reid. Forty years of multiple hypothesis tracking — A review of key developments.
  In *21st International Conference on Information Fusion*, Cambridge, UK, July 2018.
  URL https://www.researchgate.net/publication/327483668
- Y. Bar-Shalom, S. S. Blackman, and R. J. Fitzgerald. Dimensionless score function for multiple hypothesis tracking.
  IEEE Transactions on Aerospace and Electronic Systems, 43(1):392–400, Jan. 2007 (MTT, MHT)
- B.-N. Vo, M. Mallick, Y. Bar-Shalom, S. Coraluppi, R. Osborne, III, R. Mahler, and B.-T. Vo. *Multitarget Tracking.*
  Wiley Encyclopedia of Electrical and Electronics Engineering, 2015.
  URL https://www.researchgate.net/publication/283623828_Multitarget_Tracking (MTT, MHT)
- J. Williams. Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and association-based MeMBer.
  IEEE Transactions on Aerospace and Electronic Systems, 51(3):1664–1687, July 2015 (MHT and RFS, see later lecture)

Multi-Hypothesis Tracking



LINKÖPING UNIVERSITY

System Overview



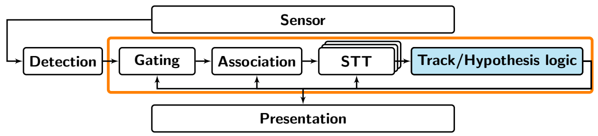
An MHT can conceptually be seen as:

- Generating all possible association hypotheses.
- Run an SHT for each potential association.
- Compute the probability of the different options.
- Reduce the number of hypothesis to make the algorithms manageable.

Multiple Hypothesis Tracking (MHT)

- MHT: consider multiple associations hypotheses over time, *i.e.*, difficult decisions are postponed until more data is available.
- MHT took off with the seminal paper (Reid, 1979).
- There were MHT solutions before Reid's, but not as efficient.
- Integrated track initialization.
- Two principal implementations:
  - Hypotheses-oriented (HO-MHT)
  - Track-oriented (TO-MHT)
- TO-MHT was at some point considered more efficient, but HO-MHT can now be quite efficiently implemented too.

The Conceptual MHT Principle



LINKÖPING UNIVERSITY

Conceptual MHT: basic idea

Idea

Generate all possible hypotheses, and then prune to avoid combinatorial hypotheses growth.

- Described by Reid (1979).
- Intuitive hypothesis based *brute force* implementation.
- Between consecutive time instants, different association hypotheses are kept in memory.
- Hypothesis limiting techniques:
  - Prune low probability hypotheses.
  - N-scan pruning.
  - Merge similar hypotheses.
- Ensures measurement-track consistency!

Conceptual MHT: efficient implementation

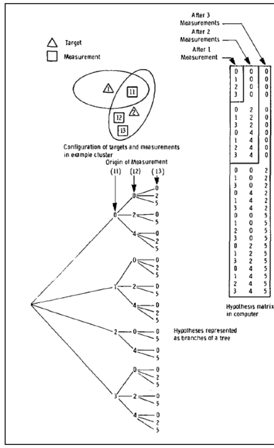


Image: Reid (1979)

- Reid (1979): list with hypothesis.
- One target generates only at most one measurement.
- Gating to remove unlikely combinations.
- Clustering could be used to split the problem in simpler ones.
- Example:
  - Two prior tracks (1 and 2)
  - Three new tracks (3,4, and 5)
  - Measurements denoted 11, 12, 13.
  - Alg: Measurement loop outside hypothesis loop

Hypothesis Probabilities (from last lecture on SHT)

Consider association hypotesis θ_t in measurement scan Y_t.

$$p(\theta_t|Y_t) \propto (\beta_{\text{FA}})^{m_t^{\text{FA}}}(\beta_{\text{NT}})^{m_t^{\text{NT}}}\Big[\prod_{j\in\mathcal{J}} P_{\text{D}} p_{t|t-1}^{(j)}\big(y_t^{(\theta_t^{-1}(j))}\big)\Big]\Big[\prod_{j\in\bar{\mathcal{J}}}(1-P_{\text{D}}P_{\text{G}})\Big],$$

where

- Measurement to track association at time t: θ_t
- $\mathcal{J}$ is the set of indices of detected tracks (assigned).
- $\bar{\mathcal{J}}$ is the set of indices of non-detected tracks (not assigned).
- $\theta_t^{-1}(j)$ is the index of the measurement that is assigned to track $j \in \mathcal{J}$. ($\theta_t^{-1}(j) = \emptyset$ is shorthand for no measurement associated with track j.)
- All but the last factors are associated with a measurement.

Extended Notation to Handle MHT

- One measurement sequence: $y_{1:t} = \{y_1, y_2, \dots, y_t\}$.
- Measurements in a scan: $Y_t = \{y_t^{(1)}, y_t^{(2)}, \dots, y_t^{(m_t)}\}$
- $Y_{1:t} = \{Y_1, Y_2, \dots, Y_t\}$
- The set of measurement to track association at time t: θ_t
- Hypothesis i at time t : $\theta_t^{(i)}$.
- $\theta_{1:t}$ is the history of measurement to track associations.
- Between consecutive time instants, N_h different association hypotheses, $\{\theta_{1:t-1}^{(i)}\}_{i=1..N_h}$, are kept in memory.
- $\theta_{1:t}^{(ij)} = (\theta_{1:t-1}^{(i)}, \theta_t^{(j)})$

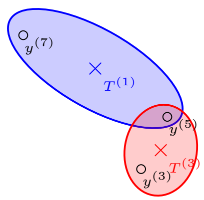
Generating Hypotheses

- Assume the hypotheses from time $t-1$, $\{\theta^{(i)}_{1:t-1}\}_i$.
- Form all possible new hypotheses,

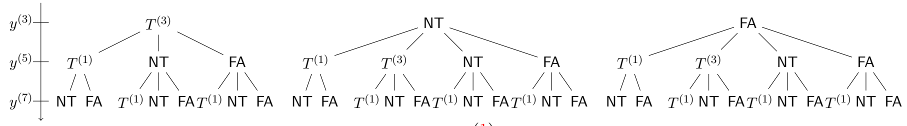
$$\theta^{(ij)}_{1:t} = (\theta^{(i)}_{1:t-1}, \theta^{(j)}_t),$$

with the obtained measurements, Y_t.
I.e., each measurement should be assigned either to an existing track, create a new track, or be considered a false detection.

Hypothesis Example



Hypothesis: $\theta^{(1)}_{1:t-1}$



All possible hypotheses derived from $\theta^{(1)}_{1:t-1}$ *and* $Y_t = \{y^{(3)}, y^{(5)}, y^{(7)}\}$

Hypothesis Probabilities

Now, let $\theta^{(ij)}_{1:t} = \{\theta^{(i)}_{1:t-1}, \theta^{(j)}_t\}$, then applying Baye's rule and $Y_{1:t} = \{Y_t, Y_{1:t-1}\}$

$$\begin{aligned} p(\theta^{(ij)}_{1:t}|Y_{1:t}) &\propto p(Y_t|\theta^{(ij)}_{1:t}, Y_{1:t-1})p(\theta^{(ij)}_{1:t}|Y_{1:t-1}) \\ &\propto p(Y_t|\theta^{(ij)}_{1:t}, Y_{1:t-1})p(\theta^{(j)}_t|\theta^{(i)}_{1:t-1}, Y_{1:t-1})p(\theta^{(i)}_{1:t-1}|Y_{1:t-1}) \\ &\propto \beta_{\mathrm{FA}}^{m_t^{\mathrm{FA}}}\beta_{\mathrm{NT}}^{m_t^{\mathrm{NT}}}\Big[\prod_{k\in\mathcal{J}^{(j)}} P_{\mathrm{D}} p^{(k)}_{t|t-1}(y_t^{((\theta_t^{(j)})^{-1}(k))})\Big]\Big[\prod_{k\in\bar{\mathcal{J}}^{(j)}}(1-P_{\mathrm{D}}P_{\mathrm{G}})\Big]p(\theta^{(i)}_{1:t-1}|Y_{1:t-1}) \end{aligned}$$

Hence, existing hypotheses probability is updated using the fundamental tracking formula.

Note

The sets $\mathcal{J}^{(j)}$ and $\bar{\mathcal{J}}^{(j)}$ depend on $\theta^{(i)}_{1:t-1}$! The number of targets and target estimates usually differ between hypotheses.

Complexity Reduction

The number of different hypotheses to consider grows exponentially over time, as has been illustrated, and quickly becomes intractable. Tricks and approximations are necessary to obtain a realistic problem.

Complexity reducing method:

- Clustering (as studied before, always fundamental).
- Pruning of low probability hypotheses.
- N-scan pruning.
- Merging of similar hypotheses.

Complexity Reduction: pruning

- **Delete hypotheses with low probability**
  Delete hypotheses with probability below a threshold, γ_p (*e.g.*, $\gamma_p = 0.1\,\%$):

  $$\text{Deletion Condition: } p(\theta^{(i)}_{1:t}) < \gamma_p$$

- **Keep only the most probable hypotheses**
  Keep the most probable hypotheses that together make up enough of the total probability mass, γ_c (*e.g.*, $\gamma_c = 99\,\%$):

  $$\text{Deletion Condition: } i > i_{\text{th}} = \arg\min_i \sum_{k=1}^{i} p(\theta^{(k)}_{1:t}) \geq \gamma_c,$$

  where $\theta^{(k)}_{1:t}$ has been ordered such that $p(\theta^{(k)}_{1:t}) \geq p(\theta^{(k+1)}_{1:t})$.

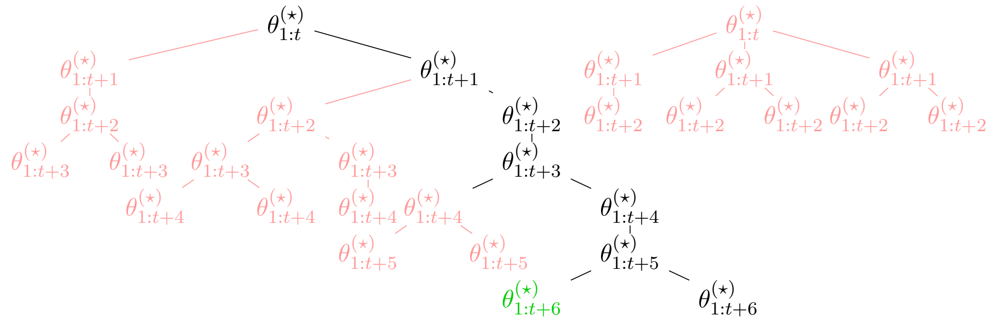
Make sure to renormalize the hypothesis probabilities after pruning.

Complexity Reduction: N-Scan Pruning

N-Scan Pruning $\Rightarrow$ Only keep the most likely node N steps back

We will look at an example with $N = 2$.

Complexity Reduction: N-Scan Pruning



Green: *most probable hypothesis* **Red:** *pruned hypotheses*

$N = 2$-scan pruning: Only keep the most likely node N steps back

Complexity Reduction: merging

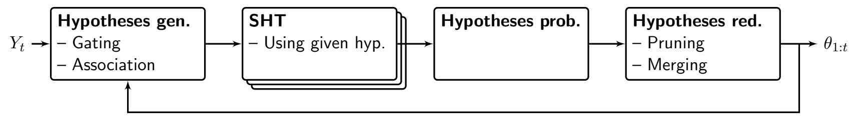
Reid's original paper suggests to check for hypothesis pairs with:

- the same number of targets (tracks)
- similar track estimates

If these conditions are satisfied:

- merge the hypotheses
- assign the new hypothesis the sum of the combined hypotheses' probability

Conceptual MHT: summary overview



An MHT can conceptually be seen as:

- Generate all possible association hypotheses.
- Run an SHT for each potential association.
- Compute the probability of the different options.
- Reduce the number of hypothesis to make the algorithms manageable.

Conceptual MHT: summary

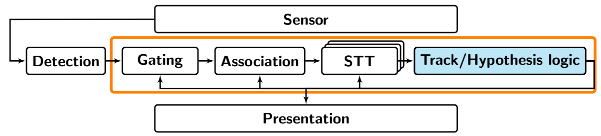
- Attractive method since each hypothesis is:
  - an alternative representation of reality
  - easily interpreted
- Drawback: generating all possible hypotheses only to discarding (most of) them is inefficient.
- Some hypotheses contain the same track; hence fewer unique tracks than hypotheses.

Extensions of the original MHT idea

HO-MHT More clever/efficient hypotheses generation: Cox and Miller (1995).

TO-MHT Track-oriented hypothesis handling.

Hypothesis-Oriented Multiple-Hypothesis Tracker



Hypothesis-Based MHT

- Proposed by Cox and Miller (1995).
- Only generate the best hypotheses, ignore hypotheses that will anyhow be deleted.
- Propagate the N_h-best hypotheses:
  - Generating as few unnecessary hypothesis as possible.
  - Use the k-best algorithm to find solutions to the assignment problem (Murty's alg).
- Regular hypothesis reduction techniques still apply.

Assignment Problem: k-best solutions

Murty's method

Given the assignment matrix $\mathcal{A}$:

- Find the best solution to the assignment problem (*e.g.*, Auction).
- For $i = 2, \dots, k$, or until there are no more solutions to evaluate:
  - Construct new assignment problems by, in turn excluding each of the assignments made in the $(i-1)^{\text{th}}$ solution.
  - Find the best solution to each of these problems (*e.g.*, Auction).
  - The i^{th} best assignment is the solution giving the maximum reward (minimum cost) among all solutions evaluated so far that have not been picked.

HO-MHT: algorithm outline

Aim: Given N_h hypotheses $\{\theta^{(i)}_{1:t-1}\}_i$ and measurements $Y_t = \{y_t^{(k)}\}_{k=1}^{m_t}$, find the N_h best hypotheses $\{\theta^{(ij)}_{1:t}\}_{ij}$ (*without* generating all hypotheses).

Recall: Hypothesis Probability

$$p(\theta^{(ij)}_{1:t}|Y_{1:t}) \propto \underbrace{\beta_{\text{FA}}^{m_t^{\text{FA}}}\beta_{\text{NT}}^{m_t^{\text{NT}}}\left[\prod_{k\in\mathcal{J}^{(j)}} \frac{P_{\text{D}} p^{(k)}_{t|t-1}(y_t^{(\theta_t^{(j)})^{-1}(k)})}{1-P_{\text{D}}P_{\text{G}}}\right]}_{\text{Assignment dependent}} \underbrace{C_i p(\theta^{(i)}_{1:t-1}|Y_{1:t-1})}_{\text{Prior information}}$$

$$C_i = \textstyle\prod_{k\in\mathcal{J}^{(j)}\cup\bar{\mathcal{J}}^{(j)}}(1-P_{\text{D}}P_{\text{G}})$$

- Find the N_h hypotheses $\{\theta^{(ij)}_{1:t}\}_{ij}$ that maximizes $p(\theta^{(ij)}_{1:t}|Y_{1:t})$.
  - Obtain the solution from the assignment (Murty's method).
  - Multiply the obtained quantity by previous hypothesis dependent terms.

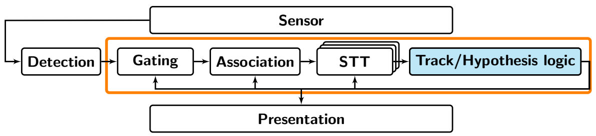
HO-MHT: Generating the N_h-best Hypotheses

Input $\{\theta^{(i)}_{1:t-1}\}_i$, $\{P(\theta^{(i)}_{1:t-1}|Y_{0:t-1})\}_i$, and $\{y_t^{(k)}\}_{k=1}^{m_t}$

Output HYP-LIST (N_h hypotheses, decreasing probability)
PROB-LIST (matching probabilities)

1. Initialize all elements in HYP-LIST and PROB-LIST to $\emptyset$ and -1, respectively.
2. Compute the assignment matrices $\{\mathcal{A}^{(i)}\}_{i=1}^{N_h}$ for $\{\theta^{(i)}_{1:t-1}\}_{i=1}^{N_h}$
3. For $i = 1, \dots, N_h$
   For $j = 1, \dots, N_h$
   - i). For the assignment matrix $\mathcal{A}^{(i)}$ find the j^{th} best solution $\theta^{(ij)}_{1:t}$.
   - ii). Compute the probability $p(\theta^{(ij)}_{1:t})$.
   - iii). Update HYP-LIST and PROB-LIST: If the new hypothesis enters the list, discard the least probable entry.
   - iv). If $p(\theta^{(ij)}_{1:t})$ is lower than the lowest probability in PROB-LIST discard $\theta^{(ij)}_{1:t}$ and never use $\mathcal{A}^{(i)}$ again in subsequent recursions.

Track-Oriented Multiple-Hypothesis Tracker



Track-Based MHT: motivation

- There are usually more hypotheses than tracks.
- Typically, hypotheses usually contain identical tracks — significantly fewer tracks than hypotheses.
- Instead of hypotheses try to build the MHT from tracks:
  - First: consider all track updates within the gating region.
  - Later: impose the usual constraint; one measurement to one track.

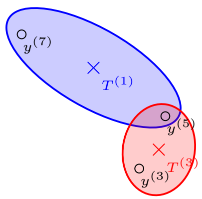
Note: hypotheses are generated as needed each time from the tracks.

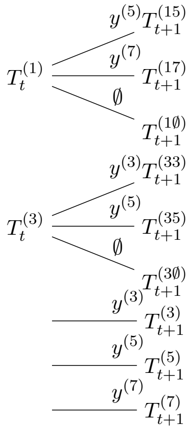
Idea

Store tracks, $T^{(i)}$, not hypotheses, $\theta^{(j)}_{1:t}$, over time.

Track-Based MHT: principle

- Tracks at time t, $\{T_t^{(i)}\}_i$
- Track scores, $\mathrm{Sc}(T_t^{(i)})$
- Form a track tree, not a hypothesis tree
- Delete tracks with low scores

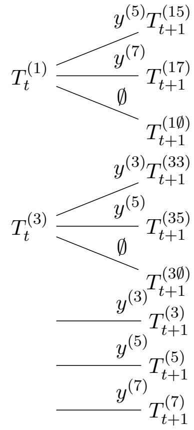




Track-Based MHT: hypotheses generation

- Hypothesis: a collection of compatible tracks:
  $\theta_{1:t+1}^{(1)} = \{T_{t+1}^{(17)}, T_{t+1}^{(35)}\}$, $\theta_{1:t+1}^{(2)} = \{T_{t+1}^{(1\emptyset)}, T_{t+1}^{(35)}, T_{t+1}^{(3)}, T_{t+1}^{(7)}\}$
- Generating hypothesis is needed for reducing the number of tracks further and for user presentation
- Use only tracks with high score
- Keep track compatibility information (*e.g.*, in a binary matrix)

	$T_{t+1}^{(15)}$	$T_{t+1}^{(17)}$	$T_{t+1}^{(1\emptyset)}$	$T_{t+1}^{(33)}$	$T_{t+1}^{(35)}$	$T_{t+1}^{(3\emptyset)}$	$T_{t+1}^{(3)}$	$T_{t+1}^{(5)}$	$T_{t+1}^{(7)}$
$T_{t+1}^{(15)}$	0	0	0	1	0	1	1	0	1
$T_{t+1}^{(17)}$		0	0	1	1	1	1	1	0
$T_{t+1}^{(1\emptyset)}$			0	1	1	1	1	1	1
$T_{t+1}^{(33)}$				0	0	0	0	1	1
$T_{t+1}^{(35)}$					0	0	1	0	1
$T_{t+1}^{(3\emptyset)}$						0	1	1	1
$T_{t+1}^{(3)}$							0	1	1
$T_{t+1}^{(5)}$								0	1
$T_{t+1}^{(7)}$									0



Track Scores and Hypotheses Probabilities

From Lecture 3:

$$\begin{aligned} \mathcal{H}_0 &: \quad \mathbb{Y}_t \text{ all originate from FA} \\ \mathcal{H}_1 &: \quad \mathbb{Y}_t \text{ originate from a single target} \end{aligned}$$

The track score is the matching log probability ratio

$$L_t = \log \frac{\Pr(\mathcal{H}_1 | \mathbb{Y}_t)}{\Pr(\mathcal{H}_0 | \mathbb{Y}_t)}$$

The probabilities of a track can be obtained from the track score

$$\Pr(\mathcal{H}_1 | \mathbb{Y}_t) = \frac{e^{L_t}}{1 + e^{L_t}},$$

How do we do this for the MHT?

MHT: Track Scores and Hypotheses Probabilities

- Track probability:

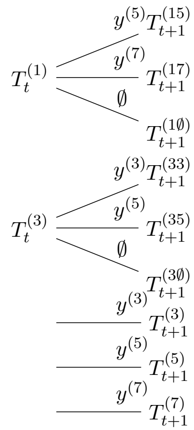
$$P(T_t^{(i)}) = \sum_{T_t^{(i)} \in \theta_{1:t}^{(j)}} P(\theta_{1:t}^{(j)})$$

- Hypothesis score:

$$\mathrm{Sc}(\theta_{1:t}^{(i)}) = \sum_{T_t^{(j)} \in \theta_{1:t}^{i}} \mathrm{Sc}(T_t^{(j)})$$

- Hypothesis probability:

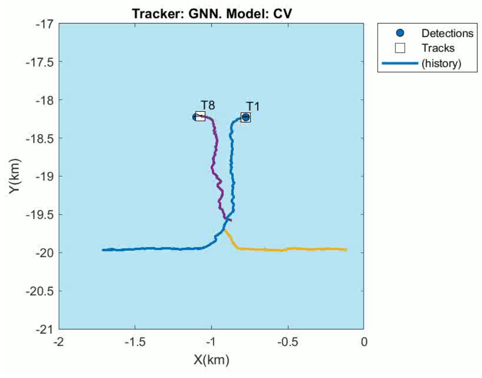
$$P(\theta_{1:t}^{(i)}) = \frac{\exp\big(\mathrm{Sc}(\theta_{1:t}^{(i)})\big)}{1 + \sum_j \exp\big(\mathrm{Sc}(\theta_{1:t}^{(j)})\big)}$$



Complexity Reducing Techniques

- Cluster incompatible tracks for efficient hypothesis generation
- Apply N-scan pruning to the track trees
- Merge tracks with common recent measurement history

MTT: GNN CV-model (from last time)

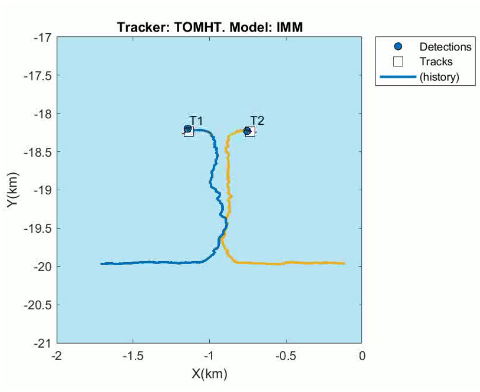


https://youtu.be/WPA2z-kw1wg

- *Global nearest neighbor* (GNN) tracker
- Simple *constant velocity* (CV) model
- Note the label switch and that one of the tracks is lost half way, and restarted as a new one.

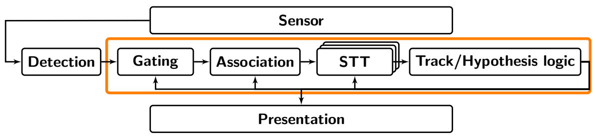
MTT: MHT IMM

- *Multi-hypothesis tracker* (MHT) resolves measurement ambiguities
- *Interacting multiple models* (IMM) better captures the mixed level of agility



https://youtu.be/ke37FrXg_1M

Practicalities



Examples of Two Tracking Frameworks

Frameworks to simplify prototyping target tracking solutions exist. Two that are well aligned with how the material is presented in this course are:

- Sensor Fusion and Tracking Toolbox in MATLAB
- Stone Soup (Python)

Both frameworks provides the ability to rather quickly prototype and experiment with tracking solutions, but should probably not be used in production code where speed is of essence.

Tracking Frameworks: Sensor Fusion and Tracking Toolbox in MATLAB

- An official MATLAB toolbox.
- Contains a fairly complete implementation of tracking methods. The toolbox also contains sensor fusion components often found in tracking applications, *e.g.*, *inertial navigation systems* (INS), as well as IMM, JPDA, TO-MHT, ...
- `https://www.mathworks.com/products/sensor-fusion-and-tracking.html`

Stone Soup (Python)

- An open source tracking framework in Python.
- Development lead by *Defence Science and Technology Laboratory* (DSTL), the UK, i collaboration with similar institutions around the world.
- Still in beta.
- Will include sensor management and similar components to be able to evaluate complete tracking solutions.
- `https://stonesoup.readthedocs.io`
- Examples (live).

User Presentation Logic

- Maximum probability hypothesis: simplest alternative.
  - Possibly jumpy; the maximum probability hypothesis can change erratically.
- Show track clusters: (weighted) mean, covariance and expected number of targets.
- Keep a separate track list: update at each step with a selection of tracks from different hypotheses.
- Consult (Blackman and Popoli, 1999) for details.

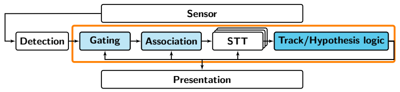
Which MTT Method to Use?

Computation \ SNR	Low	Medium	High
Low	Group TT / PHD	GNN	GNN
Medium	MHT	GNN or JPDA	GNN
High	TrBD / MHT	MHT	Any

- GNN and JPDA are very bad in low SNR.
- When using GNN, one generally has to enlarge the overconfident covariances to account for neglected data association uncertainty.
- JPDA has track coalescence and should not be used with closely spaced targets, see the "coalescence avoiding" versions.
- MHT requires significantly higher computational load but it is said to be able to work reasonably under 10–100 times worse SNR.

Summary

Summary



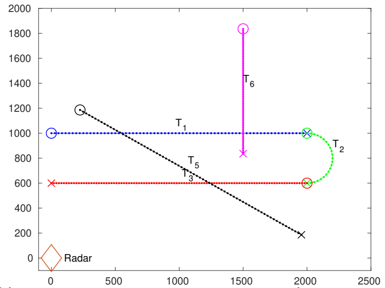
Multi-Hypotheses Tracker

- The conseptual MHT given by Reid 1979
- The Hypothesis-Oriented MHT (HO-MHT)
  - Use the k-best solutions to the assignment problem (Murty's method)
  - Find the N_h-best hypothesis, generating as few unnecessary hypothesis as possible
- Track-Oriented MHT (TO-MHT)
  - Maintain tracks, create hypotheses when needed.
  - Less tracks than global hypotheses.
- Presentation of the current state is not trivial.

Exercises

Exercise 3

1. Apply the MHT to the simulated scenario from previous exercise



- Simulate trajectory
- Generate measurement:
  - P_{D}
  - P_{FA}
  - clutter
- Details specificed in the previous exercise
- Murty's method provided

Note: see separate exercise document.

Exercise 3

2. Apply the MHT to the mysterious data set from previous exercise

- MHT
- Compare with JPDA, GNN tracking.

Details specificed in the previous exercise.