Detection and tracking of dynamic objects using 3D laser range sensor on a vehicle

Josip Ćesić

University of Zagreb
Faculty of Electrical Engineering and Computing
Department of Control and Computer Engineering

September 20, 2013

University of Zagreb
Faculty of Electrical Engineering
and Computing

Content

Content

Motivation

Autonomous driving

⇒ **DATMO** - *Detection And Tracking of Moving Objects*

Motivation

Autonomous driving

- safety
  - 2011 – 42,408 accidents
  - 2012 – 37,065 accidents

$\Rightarrow$ **DATMO** - *Detection And Tracking of Moving Objects*

Motivation

Autonomous driving

- safety
  - 2011 – 42,408 accidents
  - 2012 – 37,065 accidents
- efficiency

$\Rightarrow$ **DATMO** - *Detection And Tracking of Moving Objects*

Motivation

Autonomous driving

- safety
  - 2011 – 42,408 accidents
  - 2012 – 37,065 accidents
- efficiency
- comfort

$\Rightarrow$ **DATMO** - *Detection And Tracking of Moving Objects*

Experimental setting – Lidar System

- *Velodyne HDL-32E High Definition Lidar*
- 32 laser-detector pairs – **lidar**
- Field of view:
  - horizontal: 360°
  - vertical: -30.67° – $+10.67^\circ$
- Resolution:
  - horizontal: 0.16°
  - vertical: 1.33°
- Rotation frequency: $10\,\mathrm{Hz}$
- Scanning frequency: $700,000\,\mathrm{Hz}$
- Range: $70\,\mathrm{m}$

Experimental setting – Mobile Platform

- *Husky A*200
- Localization by encoders
- Max. velocity 1 m/s

$$\mathbf{T}_{odom} = \begin{bmatrix} \mathbf{R}_{odom} & \mathbf{t}_{odom} \\ \mathbf{0}^{1\times 3} & 1 \end{bmatrix}$$

Experimental setting – Lidar Scan Example

State-of-the-art

- DATMO based on Bayesian approach [Kaestner et al., 2010], [Kaestner et al., 2012] – Zürich, Switzerland

State-of-the-art

- DATMO based on Bayesian approach [Kaestner et al., 2010], [Kaestner et al., 2012] – Zürich, Switzerland
- DATMO based on local surface geometry [Moosmann et al., 2009], [Moosmann and Fraichard, 2010] – Karlsruhe, Germany

State-of-the-art

- DATMO based on Bayesian approach [Kaestner et al., 2010], [Kaestner et al., 2012] – Zürich, Switzerland
- DATMO based on local surface geometry [Moosmann et al., 2009], [Moosmann and Fraichard, 2010] – Karlsruhe, Germany
- DATMO based on simultaneous localization and mapping [Azim and Aycard, 2012] – Grenoble, France

Kaestner et al.[1], ETH Zürich

- Bayesian approach to learning 3D representation of dynamic environment
- measurement corresponds to dynamic or static object
- set of Gaussian mixtures corresponding to objects
- generative approach
- Euclidean based clustering with adaptive distance metric

[1] [Kaestner et al., 2010],[Kaestner et al., 2012]

Moosman et al.[2], Karlsruhe Institute of Technology

- local surface geometry features based technique
- segmentation based upon the observation that
  a) many object parts have convex outlines
  b) vertical structure usually represents single object
- motion estimation links two consecutive point clouds
  a) feature vector – Nearest Neighbour Search
  b) iterative closest point refines given estimate

[2] [Moosmann et al., 2009],[Moosmann and Fraichard, 2010]

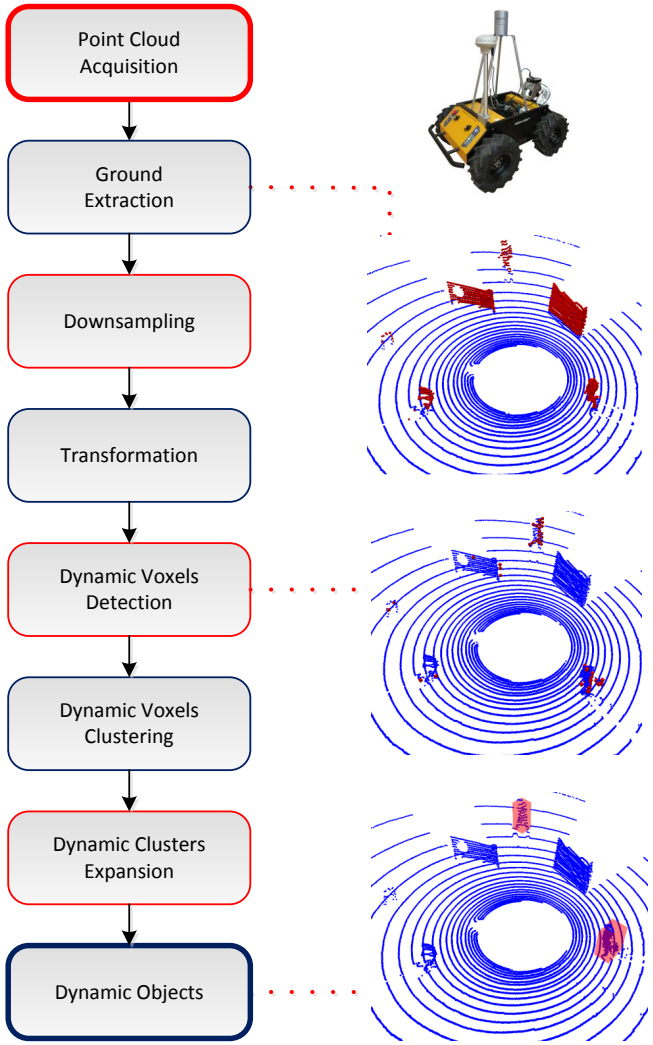
Azim and Aycard[3], Laboratoire d'Informatique de Grenoble

- simultaneous localization and mapping - OctoMap (VoxelGrid filtering)
- detection and tracking of moving objects
- Global Nearest Neighbour data association and Kalman filter

[3] [Azim and Aycard, 2012]

Content

DOD – Detection pipeline



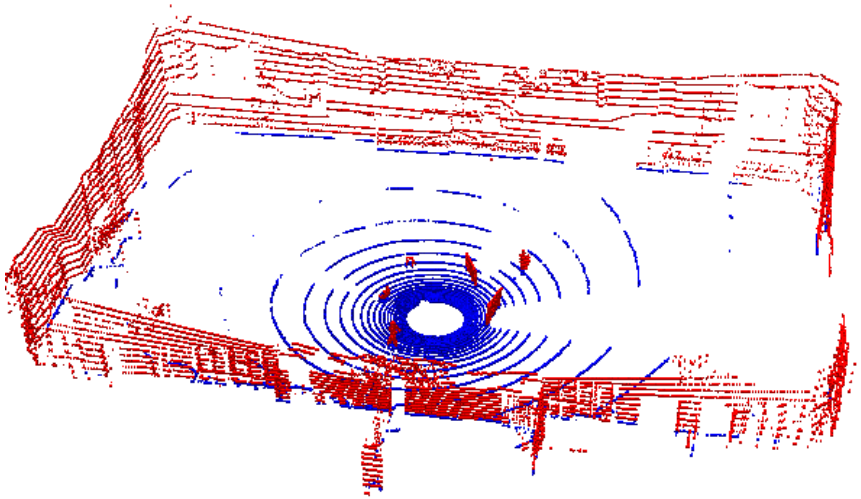
Flow Chart of the dynamic object detection algorithm

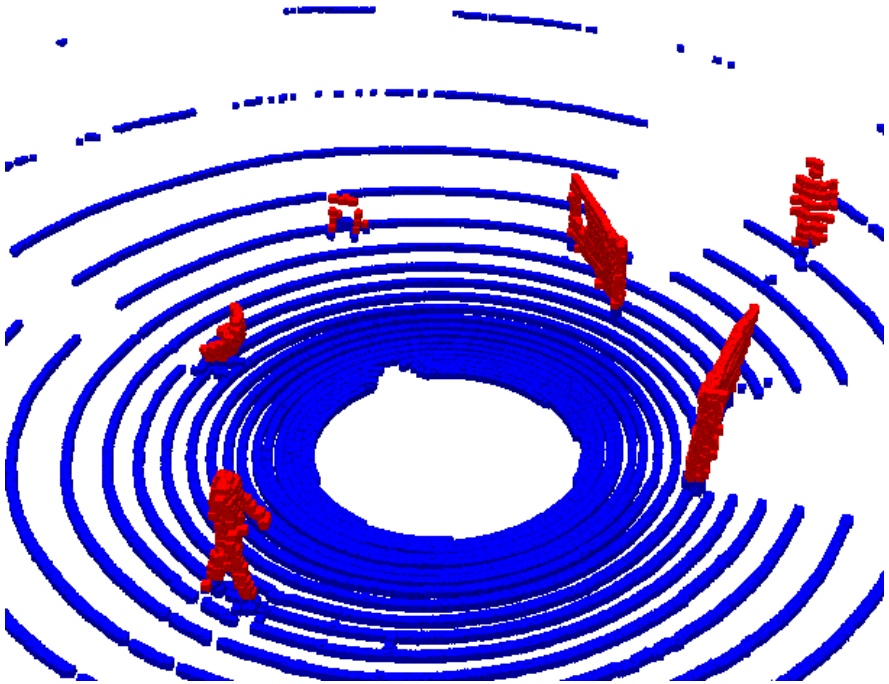
DOD – Ground Extraction

- plane model parameter estimation
- *RANSAC* – RANdom SAmple Consensus

1. iteratively selects random subset of original data – hypothetical inliers
2. model fitted to hypothetical inliers
3. other data tested against the fitted model
   $\rightarrow$ model reasonably good if sufficiently many points are classified as hypothetical inliers
4. model reestimated from all hypothetical inliers
5. model evaluation - estimating the error of the inliers

DOD – Ground Extraction example



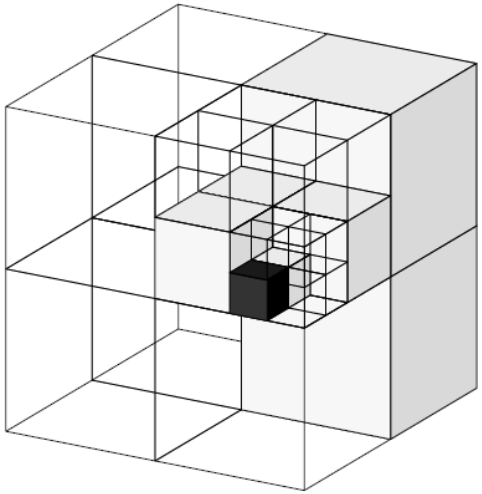


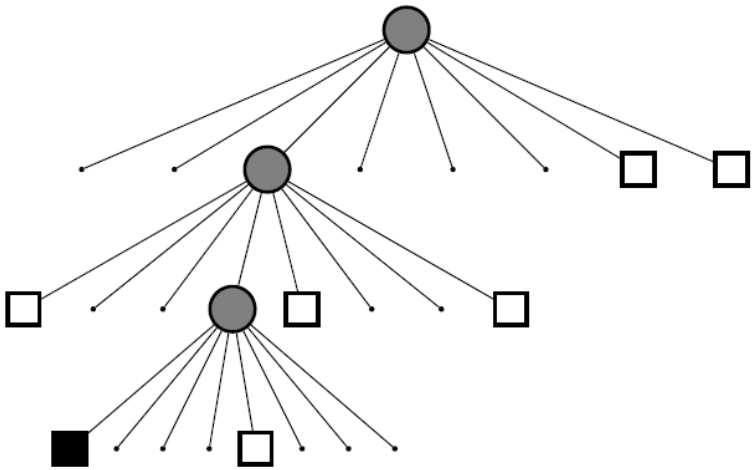
DOD – Point Cloud Downsampling

- Point Cloud filtering
  - Removing outliers using a Conditional or Radius Outlier removal
  - Removing outliers using a Statistical Outlier Removal filter
  - **Downsampling a Point Cloud using a VoxelGrid filter**

DOD – Point Cloud Downsampling

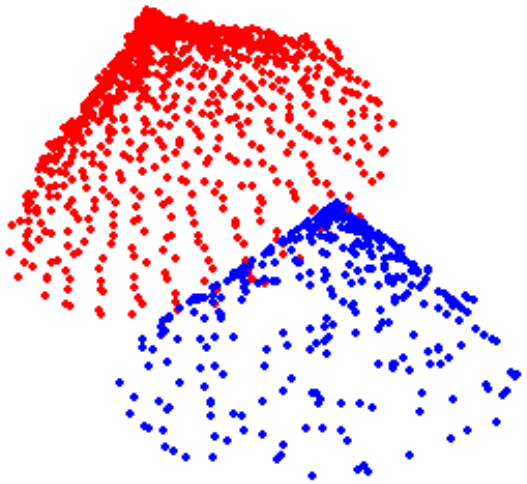
- Point Cloud filtering
  - Removing outliers using a Conditional or Radius Outlier removal
  - Removing outliers using a Statistical Outlier Removal filter
  - **Downsampling a Point Cloud using a VoxelGrid filter**
- Space voxelization – voxel resolution
- Octree data structure





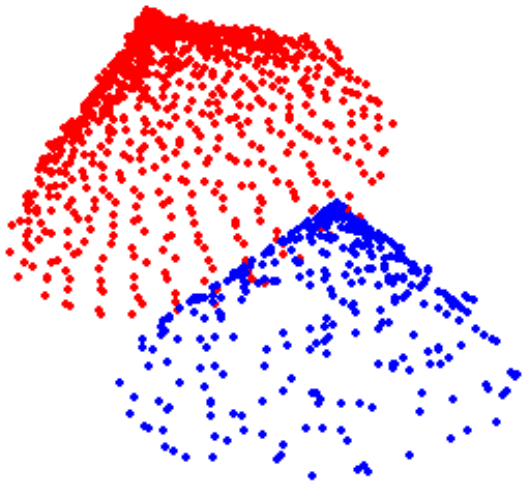
DOD – Point Cloud Transformation

- alignment of two consecutive point clouds



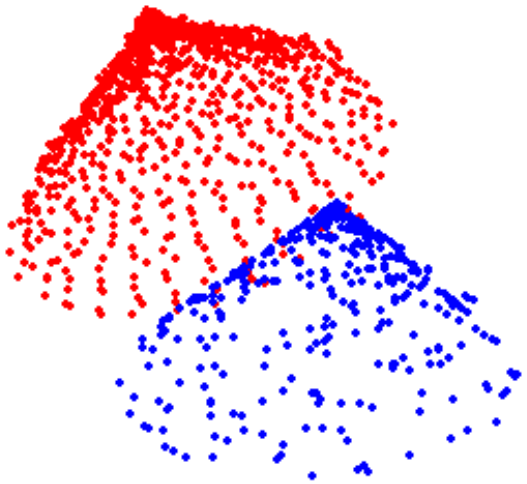
DOD – Point Cloud Transformation

- alignment of two consecutive point clouds
- accurate mobile platform localization required



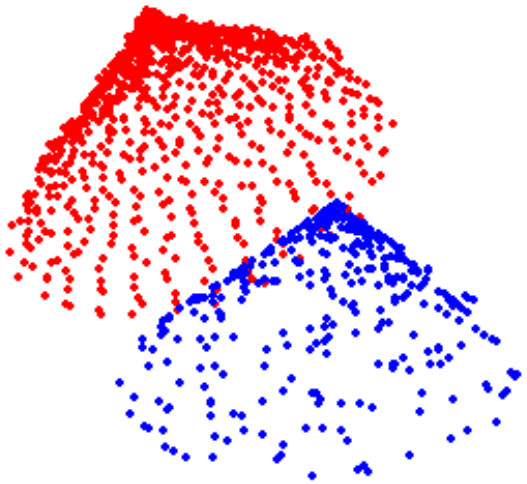
DOD – Point Cloud Transformation

- alignment of two consecutive point clouds
- accurate mobile platform localization required
  - advanced sensing systems – GPS, IMU, VSS
  - point cloud processing



DOD – Point Cloud Transformation

- alignment of two consecutive point clouds
- accurate mobile platform localization required
  - advanced sensing systems – GPS, IMU, VSS
  - point cloud processing



$\Rightarrow$ **ICP** – *Iterative Closest Point algorithm*

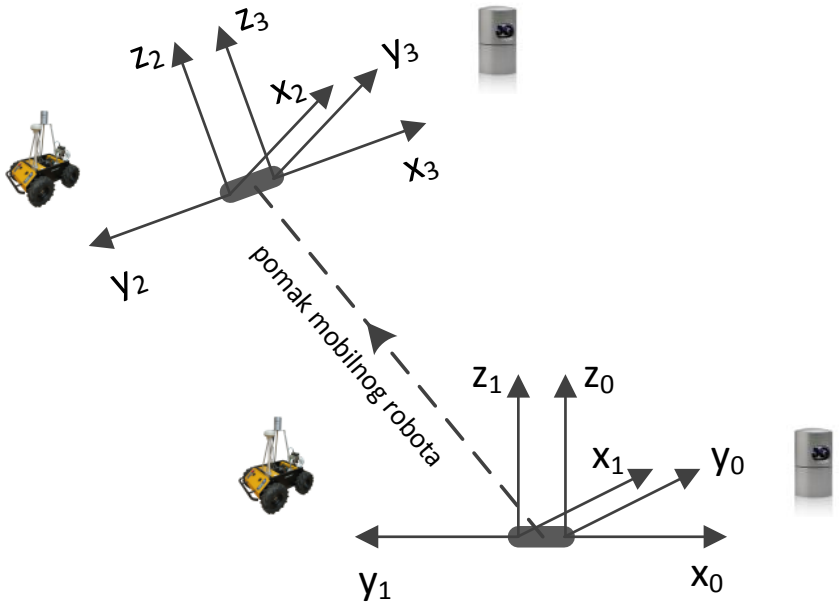
$+$ initial transformation guess by odometry

DOD – Point Cloud Registration – ICP

- ICP optimization problem:

$$\min_{\mathbf{R},\mathbf{t},j\in\{1,2,\ldots,N_m\}} \sum_{i=1}^{N_p} ||\mathbf{R}_{icp}\mathbf{p}_i + \mathbf{t}_{icp} - \mathbf{m}_j||$$

$$\Rightarrow \mathbf{T}_{icp} = \begin{bmatrix} \mathbf{R}_{icp} & \mathbf{t}_{icp} \\ \mathbf{0}^{1\times 3} & 1 \end{bmatrix}$$



DOD – Point Cloud Registration – ICP

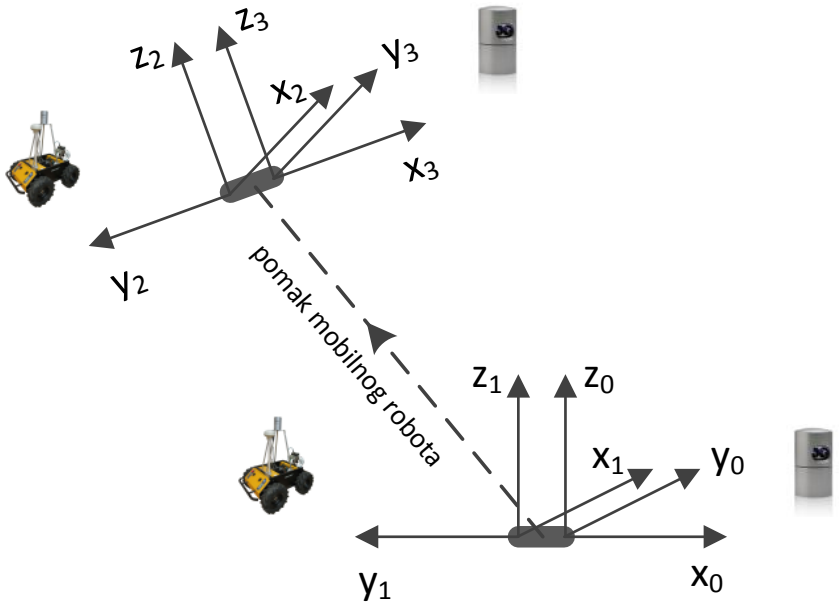
- ICP optimization problem:

$$\min_{\mathbf{R},\mathbf{t},j\in\{1,2,\ldots,N_m\}} \sum_{i=1}^{N_p} ||\mathbf{R}_{icp}\mathbf{p}_i + \mathbf{t}_{icp} - \mathbf{m}_j||$$

$$\Rightarrow \mathbf{T}_{icp} = \begin{bmatrix} \mathbf{R}_{icp} & \mathbf{t}_{icp} \\ \mathbf{0}^{1\times 3} & 1 \end{bmatrix}$$

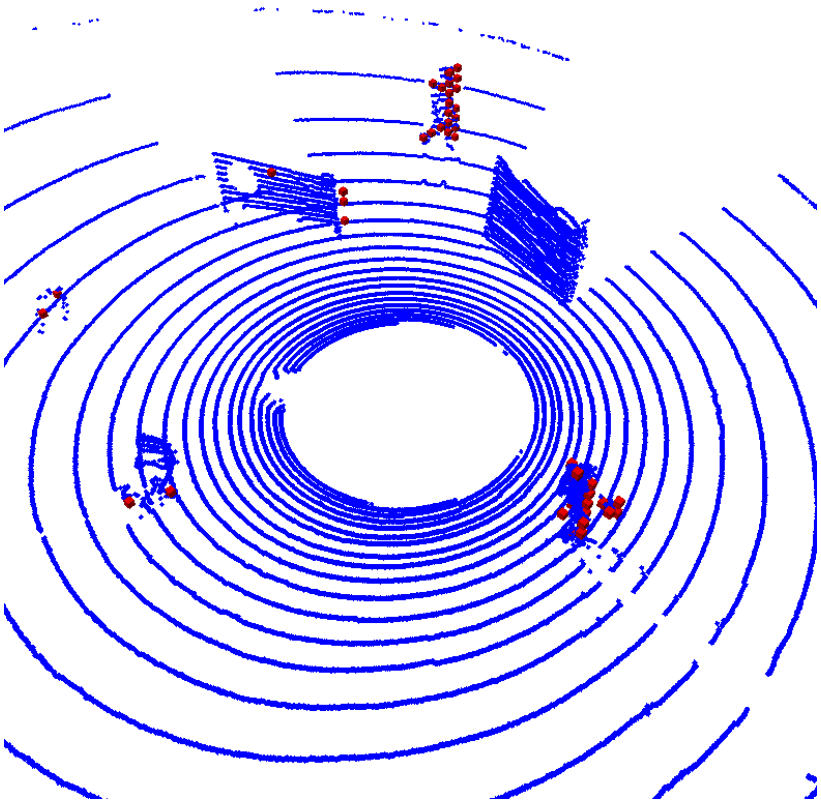
- Final transformation:

$$\mathbf{T} = \mathbf{T}_{odom} \cdot \mathbf{T}_{icp}$$



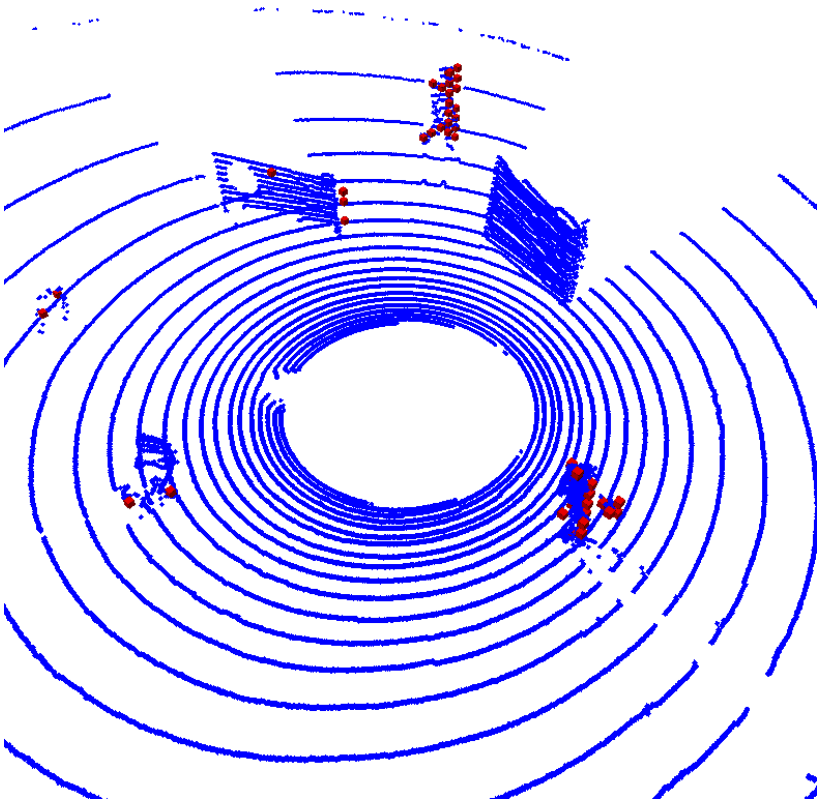
DOD – Dynamic Voxels Detection (v1)

- comparison of two consecutive point clouds
- octree data structure



DOD – Dynamic Voxels Detection (v1)

- comparison of two consecutive point clouds
- octree data structure
- voxels labelled as dynamic iff
  - $S_{t-1} = free \rightarrow S_t = occupied$
  - $S_{t-1} = unobserved \rightarrow S_t = occupied$



DOD – Dynamic Voxels Detection (v2)

- comparison of point cloud and map of environment

DOD – Dynamic Voxels Detection (v2)

- comparison of point cloud and map of environment
- Probability of a voxel being occupied:

$$P(n|z_{1:t}) = [1 + \frac{1 - P(n|z_t)}{P(n|z_t)} \cdot \frac{1 - P(n|z_{1:t-1})}{P(n|z_{1:t-1})} \cdot \frac{P(n)}{1 - P(n)}]^{-1}$$

DOD – Dynamic Voxels Detection (v2)

- comparison of point cloud and map of environment
- Probability of a voxel being occupied:

$$P(n|z_{1:t}) = [1 + \frac{1 - P(n|z_t)}{P(n|z_t)} \cdot \frac{1 - P(n|z_{1:t-1})}{P(n|z_{1:t-1})} \cdot \frac{P(n)}{1 - P(n)}]^{-1}$$

- Using *logOdds*:

$$L(n|z_{1:t}) = L(n|z_{1:t-1}) + L(n|z_t) \quad ; \quad L(\cdot) = log[\frac{P(\cdot)}{1 - P(\cdot)}]$$

DOD – Dynamic Voxels Detection (v2)

- comparison of point cloud and map of environment
- Probability of a voxel being occupied:

$$P(n|z_{1:t}) = [1 + \frac{1 - P(n|z_t)}{P(n|z_t)} \cdot \frac{1 - P(n|z_{1:t-1})}{P(n|z_{1:t-1})} \cdot \frac{P(n)}{1 - P(n)}]^{-1}$$

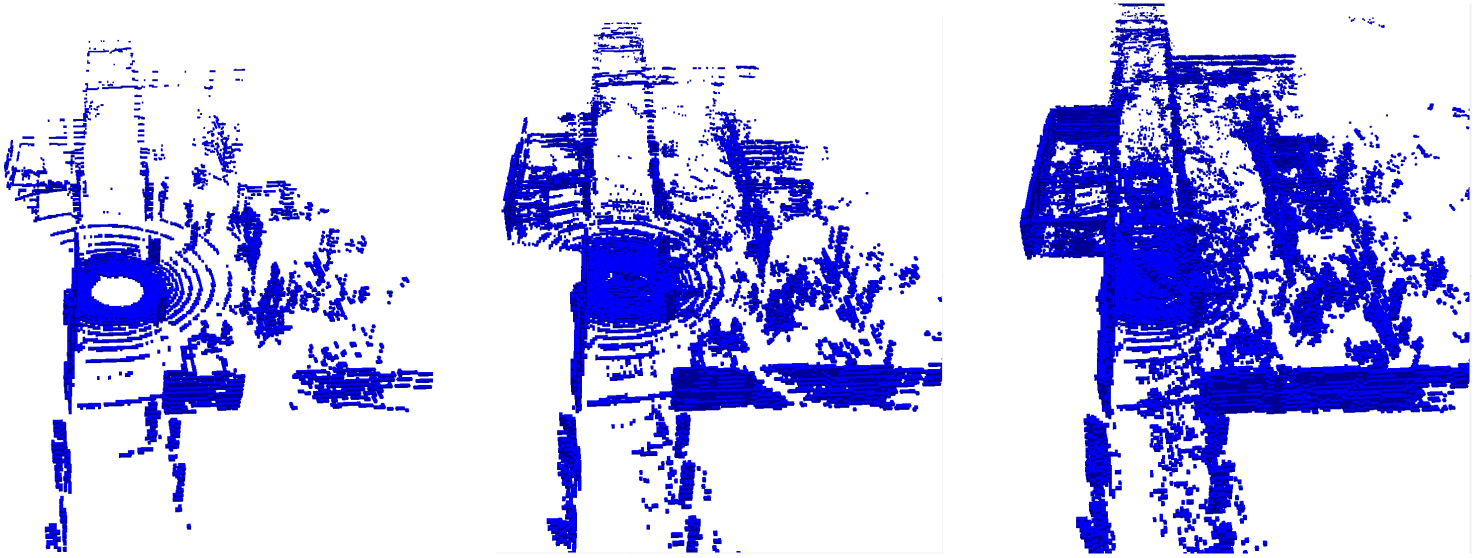
- Using *logOdds*:

$$L(n|z_{1:t}) = L(n|z_{1:t-1}) + L(n|z_t) \quad ; \quad L(\cdot) = log[\frac{P(\cdot)}{1 - P(\cdot)}]$$

- To enable adaptivity:

$$L(n|z_{1:t}) = \max\{\min[L(n|z_{1:t-1}) + L(n|z_t), l_{max}], l_{min}\}$$

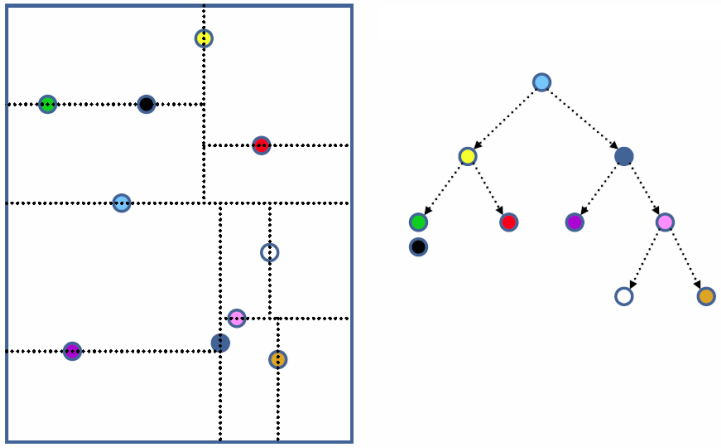
DOD – Map Example



Indoor Map – FER entrance hall after 1, 10 and 50 registered scans

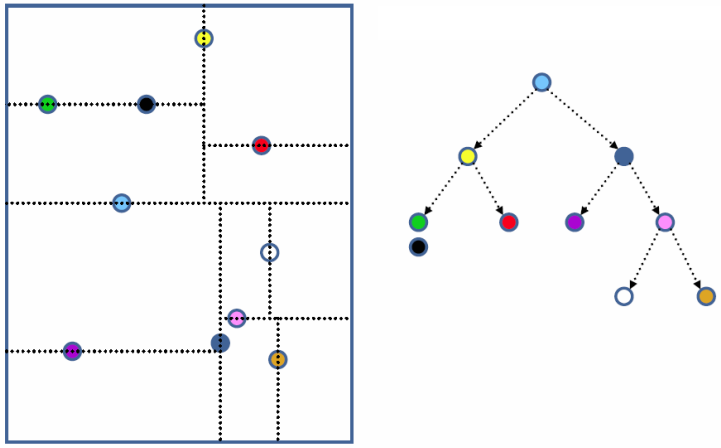
DOD – Dynamic Objects Clustering

- clustering over list of dynamic voxels
  – euclidean distance based



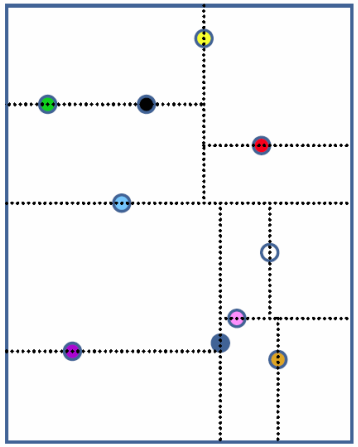
DOD – Dynamic Objects Clustering

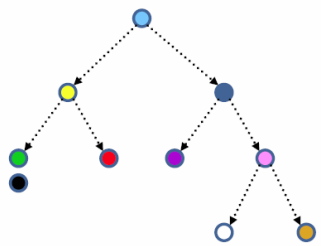
- clustering over list of dynamic voxels
  – euclidean distance based
- kd-tree data structure
  $\rightarrow$ Nearest Neighbour Search $\mathcal{O}(k \cdot logn)$
  $\rightarrow$ Within Radius Search $\mathcal{O}(n^{1/k} + m)$



DOD – Dynamic Objects Clustering

- clustering over list of dynamic voxels
  – euclidean distance based
- kd-tree data structure
  $\rightarrow$ Nearest Neighbour Search $\mathcal{O}(k \cdot logn)$
  $\rightarrow$ Within Radius Search $\mathcal{O}(n^{1/k} + m)$
- clusters broadened over list of voxels such that $S_t = occupied$

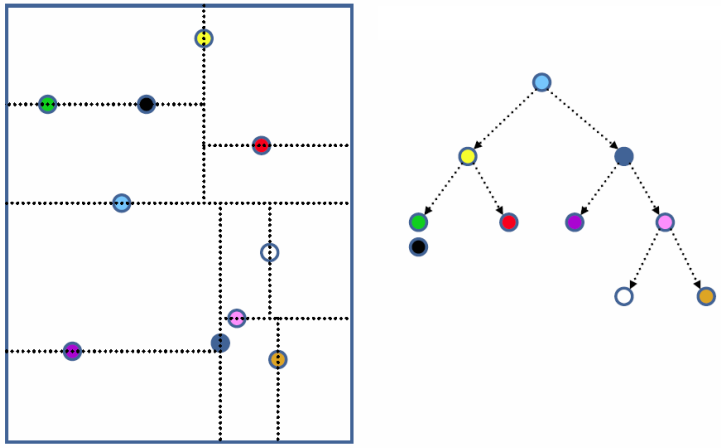




DOD – Dynamic Objects Clustering

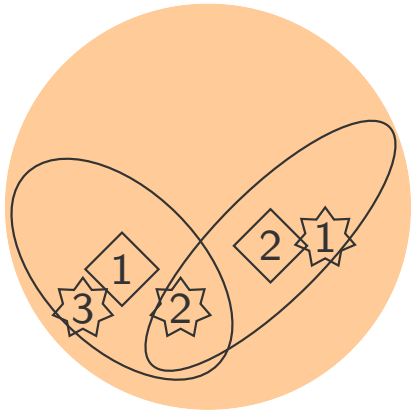
- clustering over list of dynamic voxels
  – euclidean distance based
- kd-tree data structure
  $\rightarrow$ Nearest Neighbour Search $\mathcal{O}(k \cdot logn)$
  $\rightarrow$ Within Radius Search $\mathcal{O}(n^{1/k} + m)$
- clusters broadened over list of voxels such that $S_t = occupied$

  $\Rightarrow$ Entire Dynamic Objects Clustered

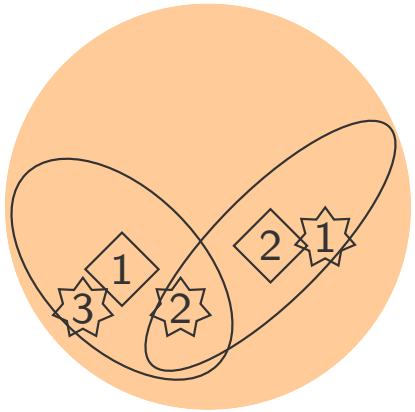


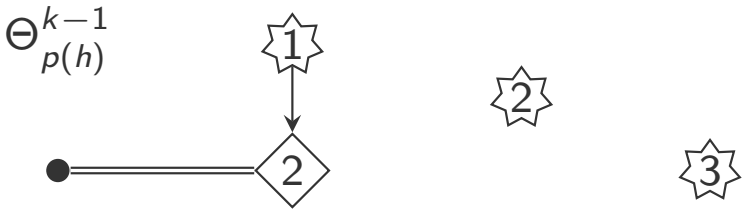
Content

DOT – Data Association – Optimal solution (MHT)

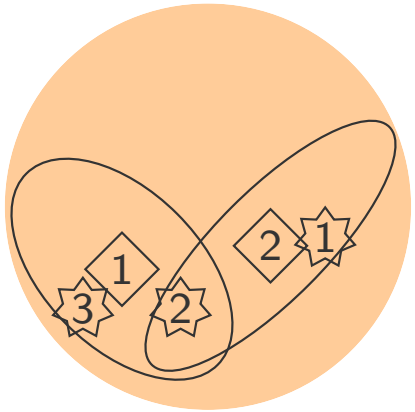


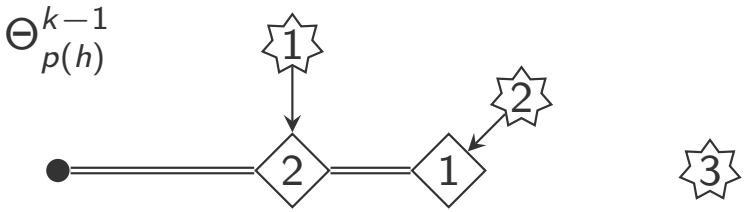
DOT – Data Association – Optimal solution (MHT)



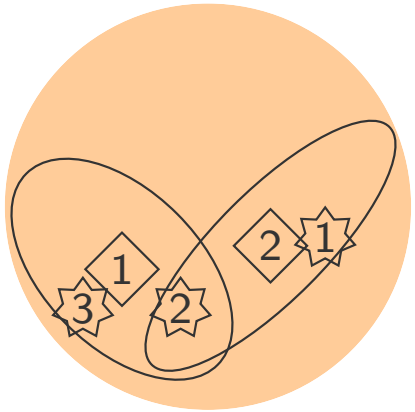


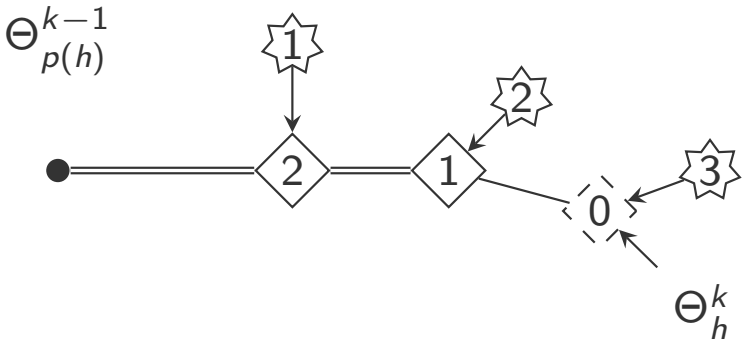
DOT – Data Association – Optimal solution (MHT)



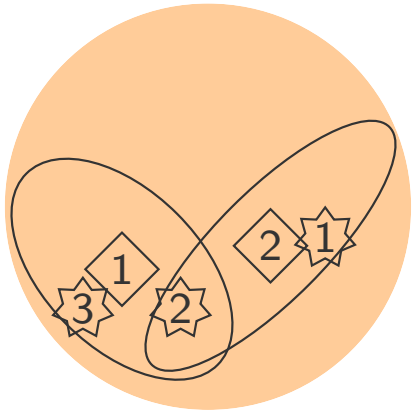


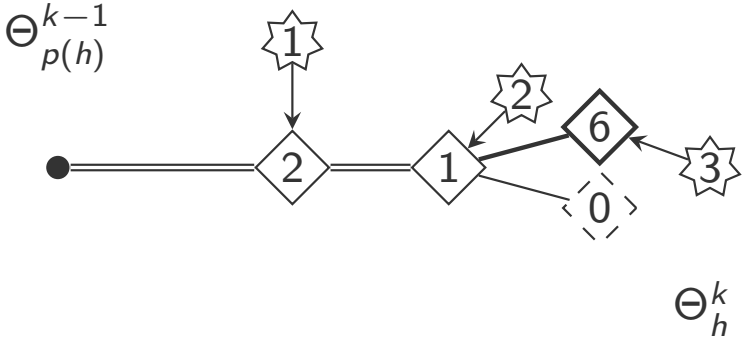
DOT – Data Association – Optimal solution (MHT)



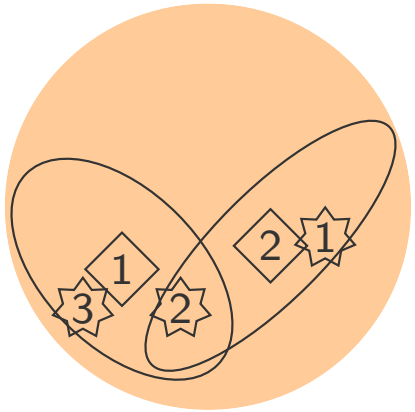


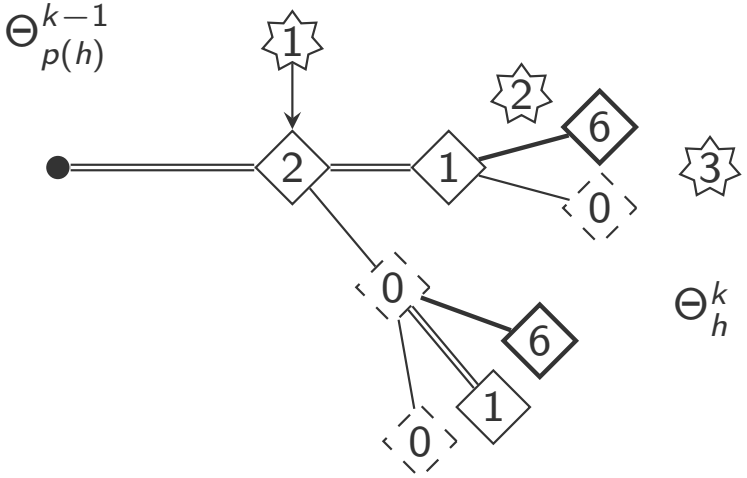
DOT – Data Association – Optimal solution (MHT)



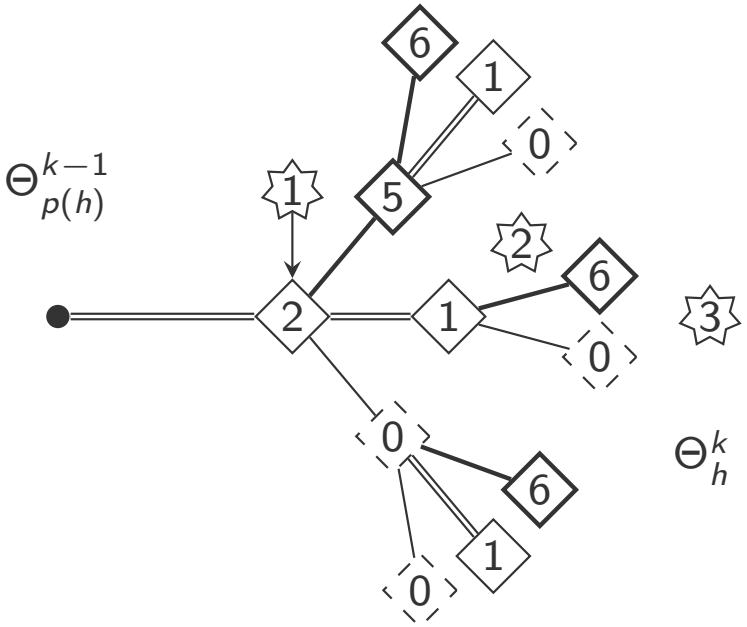


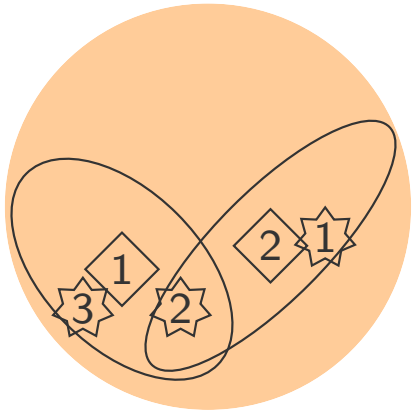
DOT – Data Association – Optimal solution (MHT)



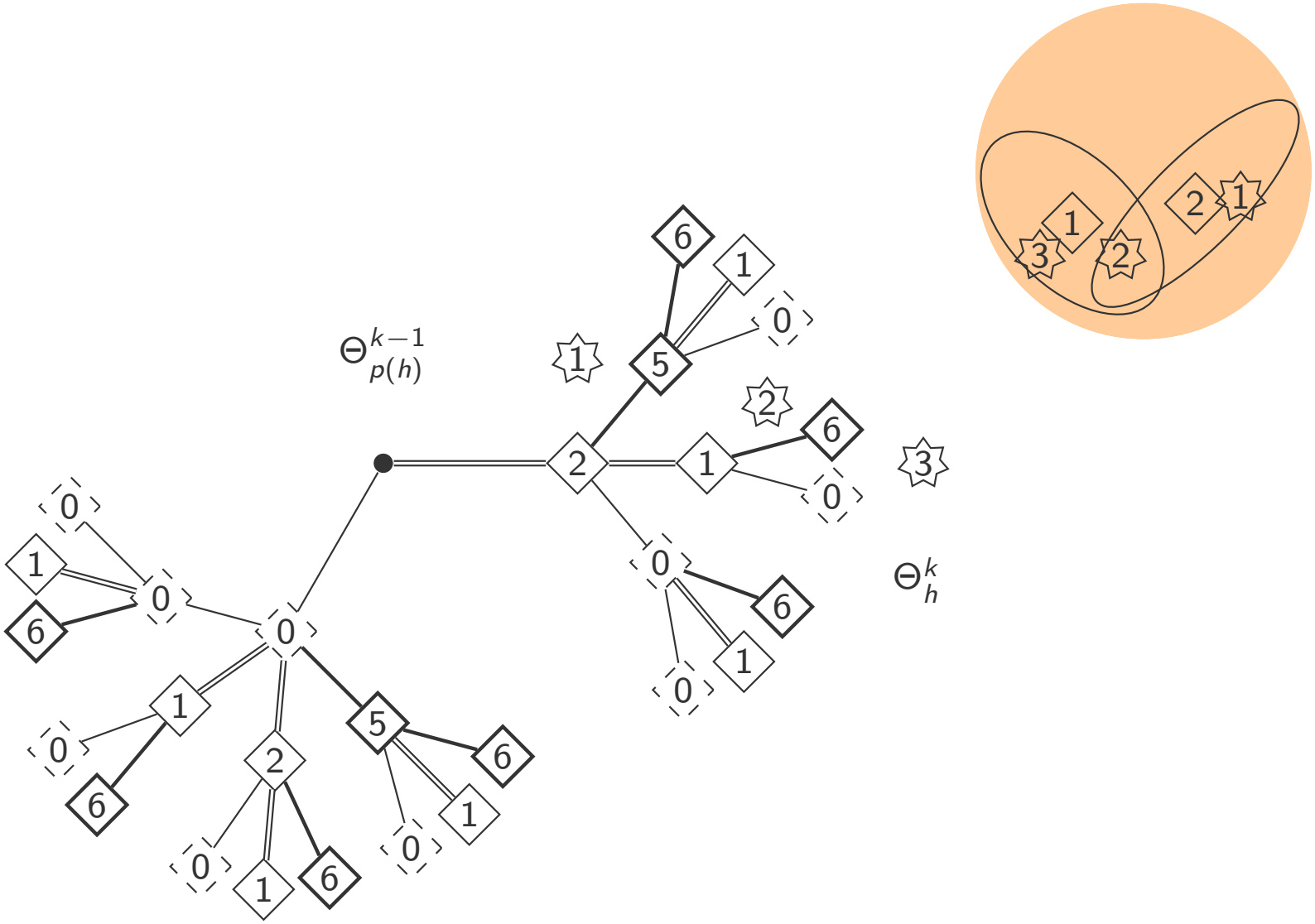


DOT – Data Association – Optimal solution (MHT)

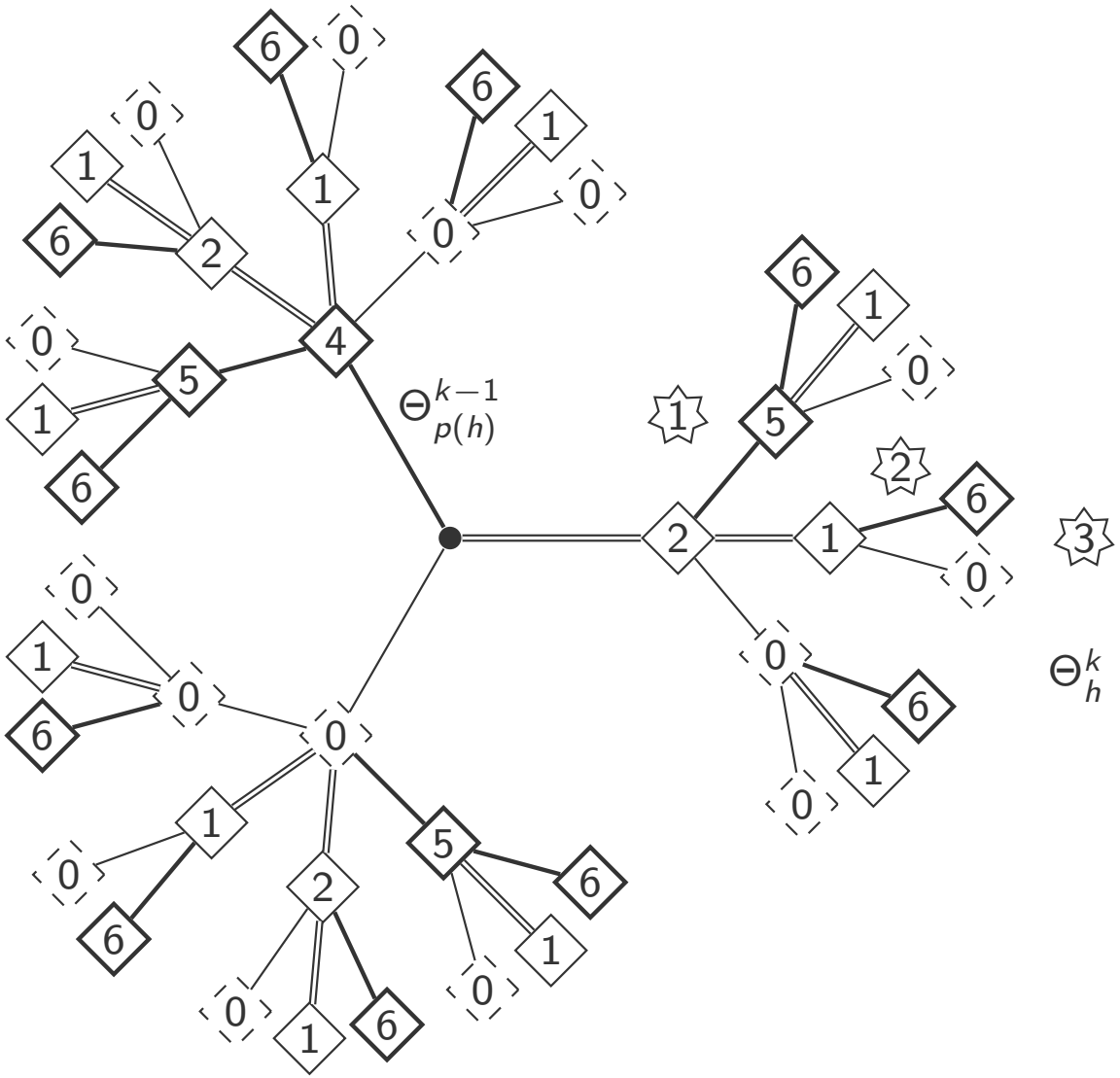


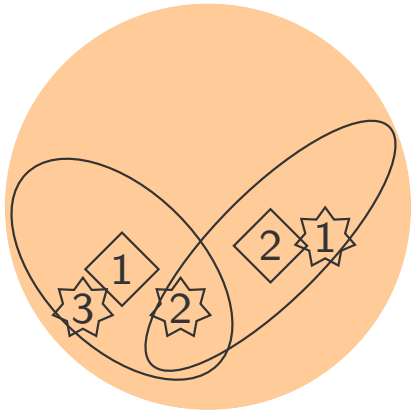


DOT – Data Association – Optimal solution (MHT)



DOT – Data Association – Optimal solution (MHT)





DOT – Data Association

- measurement to track assignment problem

DOT – Data Association

- measurement to track assignment problem
- **optimal** solution – Multiple Hypotheses Tracking (MHT)

$$
\begin{aligned}
P(\Theta_h^k|\mathbf{Z}^k) = \frac{1}{c} &\cdot P(\mathbf{Z}_k|\Theta_{p(h)}^{k-1}, \theta(k), \mathbf{Z}^{k-1}) \\
&\cdot P(\theta_h(k)|\Theta_{p(h)}^{k-1}, \mathbf{Z}^{k-1}) \\
&\cdot P(\Theta_{p(h)}^{k-1}|\mathbf{Z}^{k-1})
\end{aligned}
$$

DOT – Data Association

- measurement to track assignment problem
- **optimal** solution – Multiple Hypotheses Tracking (MHT)

$$P(\Theta_h^k|\mathbf{Z}^k) = \frac{1}{c} \cdot P(\mathbf{Z}_k|\Theta_{p(h)}^{k-1}, \theta(k), \mathbf{Z}^{k-1}) \cdot P(\theta_h(k)|\Theta_{p(h)}^{k-1}, \mathbf{Z}^{k-1}) \cdot P(\Theta_{p(h)}^{k-1}|\mathbf{Z}^{k-1})$$

- **suboptimal** solution – Joint Probabilistic Data Association Filter (JPDAF)

$$P(\Theta_h^k|\mathbf{Z}^k) = \frac{1}{c} \cdot P(\mathbf{Z}_k|\theta_h(k), \theta(k-1), \mathbf{Z}^{k-1}) \cdot P(\theta_h(k)|\theta(k-1), \mathbf{Z}^{k-1}) \cdot P(\theta(k-1)|\mathbf{Z}^{k-1})$$

DOT – Data Association – JPDAF

$$
\begin{aligned}
P(\mathbf{Z}_k|\theta_h(k), \theta(k-1), \mathbf{Z}^{k-1}) &= P(\mathbf{Z}_k|\theta_h(k)) \\
&= \prod_{j=1}^{m_k} P(z_j^k|\theta_h(k))
\end{aligned}
$$

DOT – Data Association – JPDAF

$$P(\mathbf{Z}_k|\theta_h(k),\theta(k-1),\mathbf{Z}^{k-1}) = P(\mathbf{Z}_k|\theta_h(k))$$
$$= \prod_{j=1}^{m_k} P(z_j^k|\theta_h(k))$$

$$P(z_j^k|\theta_h(k)) = \begin{cases} P_F, & z_j^k \text{ false alarm} \\ P_D\ P(z_j^k|\hat{\mathbf{x}}_t^k), & z_j^k \text{ existing track} \end{cases}$$

DOT – Data Association – JPDAF

$$P(\mathbf{Z}_k|\theta_h(k),\theta(k-1),\mathbf{Z}^{k-1}) = P(\mathbf{Z}_k|\theta_h(k)) = \prod_{j=1}^{m_k} P(z_j^k|\theta_h(k))$$

$$P(z_j^k|\theta_h(k)) = \begin{cases} P_F, & z_j^k \text{ false alarm} \\ P_D\ P(z_j^k|\hat{\mathbf{x}}_t^k), & z_j^k \text{ existing track} \end{cases}$$

- hypotheses tree reduced to single branch $\theta(k)$ – assoc. probs. given as

$$\beta_j^t = \sum_{\theta\in\Theta_{jt}^k} P(\theta|\mathbf{Z}^k) \quad \Rightarrow \quad \beta_j^t = \frac{1}{c}\sum_{\theta\in\Theta_{jt}^k}\prod_{j=1}^{m_k} P(z_j|\theta)$$

DOT – Data Association – JPDAF

$$P(\mathbf{Z}_k|\theta_h(k),\theta(k-1),\mathbf{Z}^{k-1}) = P(\mathbf{Z}_k|\theta_h(k)) = \prod_{j=1}^{m_k} P(z_j^k|\theta_h(k))$$

$$P(z_j^k|\theta_h(k)) = \begin{cases} P_F, & z_j^k \text{ false alarm} \\ P_D\ P(z_j^k|\hat{\mathbf{x}}_t^k), & z_j^k \text{ existing track} \end{cases}$$

- hypotheses tree reduced to single branch $\theta(k)$ – assoc. probs. given as

$$\beta_j^t = \sum_{\theta\in\Theta_{jt}^k} P(\theta|\mathbf{Z}^k) \quad \Rightarrow \quad \beta_j^t = \frac{1}{c}\sum_{\theta\in\Theta_{jt}^k}\prod_{j=1}^{m_k} P(z_j|\theta)$$

- Θ_{jt}^k – hypotheses that associate **measurement** j with **track** t at **time** k

DOT – Kalman JPDAF prediction

- constant velocity model – position and velocity in 2D $\mathbf{x} = [x\ \dot{x}\ y\ \dot{y}]^T$

$$\mathbf{x}^{k+1} = \mathbf{F}\mathbf{x}^k + \mathbf{G}w_k = \begin{bmatrix} 1 & \Delta T_k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}^k + \begin{bmatrix} \frac{\Delta T_k^2}{2} & 0 \\ \Delta T_k & 0 \\ 0 & \frac{\Delta T_k^2}{2} \\ 0 & \Delta T_k \end{bmatrix} w_k$$

DOT – Kalman JPDAF prediction

- constant velocity model – position and velocity in 2D $\mathbf{x} = [x\ \dot{x}\ y\ \dot{y}]^T$

$$\mathbf{x}^{k+1} = \mathbf{F}\mathbf{x}^k + \mathbf{G}w_k = \begin{bmatrix} 1 & \Delta T_k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}^k + \begin{bmatrix} \frac{\Delta T_k^2}{2} & 0 \\ \Delta T_k & 0 \\ 0 & \frac{\Delta T_k^2}{2} \\ 0 & \Delta T_k \end{bmatrix} w_k$$

- Kalman filtering **prediction**

$$\begin{aligned} \hat{\mathbf{x}}_t^{k-} &= \mathbf{F}\hat{\mathbf{x}}_t^{k-1} \\ \mathbf{P}_t^{k-} &= \mathbf{F}\mathbf{P}_t^{k-1}\mathbf{F}^T + \mathbf{G}\mathbf{Q}_t^k\mathbf{G}^T \end{aligned}$$

DOT – Kalman JPDAF update

- Innovation vector and its covariance matrix

$$\nu_j^t = \mathbf{x}_j - \mathbf{H}\hat{\mathbf{x}}_t^{k-}$$
$$\mathbf{S}_t = \mathbf{H}\mathbf{P}_t^{k-}\mathbf{H}^T + \mathbf{R}_t^k$$

DOT – Kalman JPDAF update

- Innovation vector and its covariance matrix

$$\nu_j^t = \mathbf{x}_j - \mathbf{H}\hat{\mathbf{x}}_t^{k-}$$
$$\mathbf{S}_t = \mathbf{H}\mathbf{P}_t^{k-}\mathbf{H}^T + \mathbf{R}_t^k$$

- Kalman filtering **update**

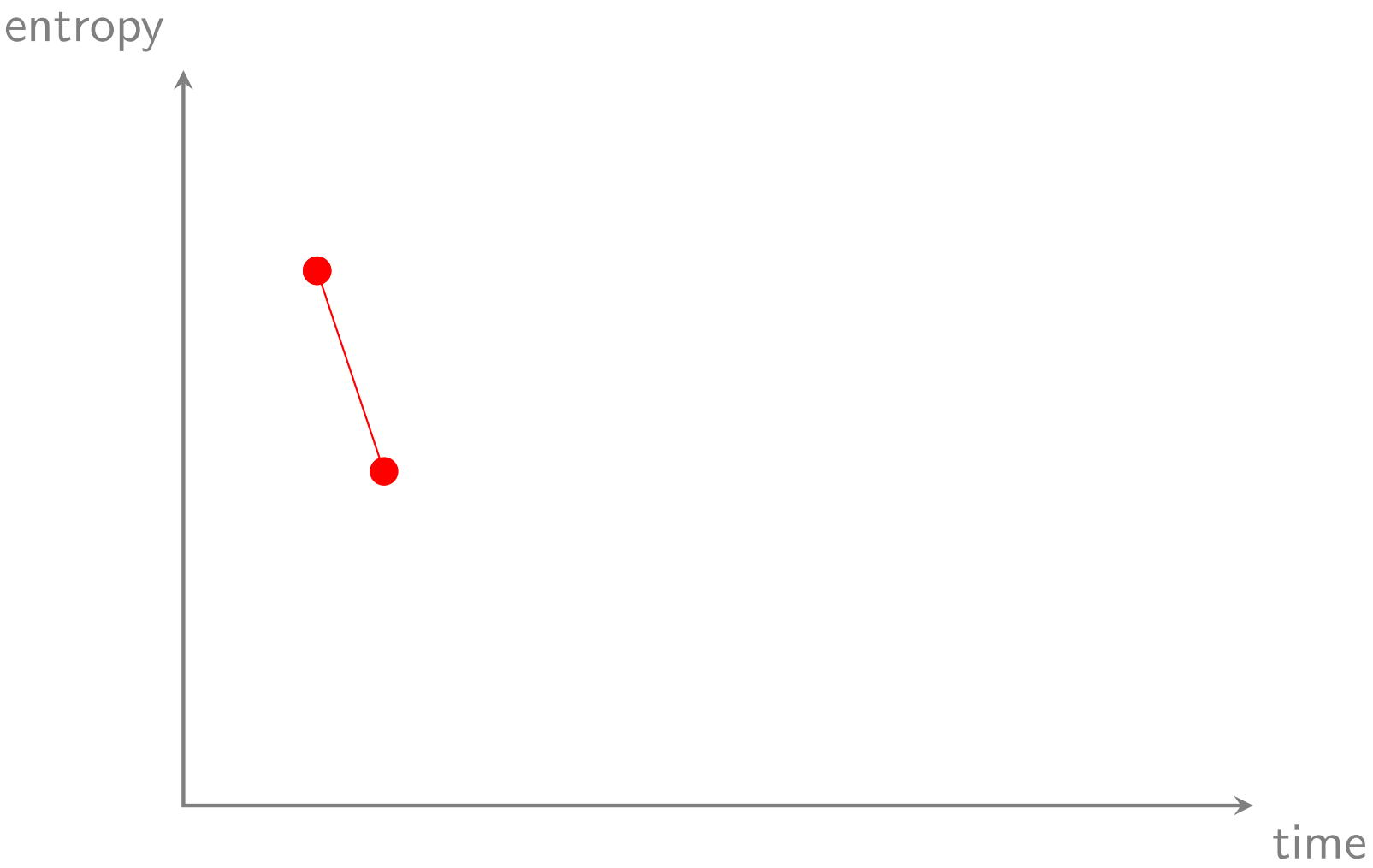
$$\nu_t = \sum_{j=1}^{m_k} \beta_j^t \nu_j^t$$
$$\mathbf{K}_k = \mathbf{P}_t^{k-}\mathbf{H}^T\mathbf{S}_t^{-1}$$
$$\hat{\mathbf{x}}_t^k = \hat{\mathbf{x}}_t^{k-} + \mathbf{K}_k\nu^t$$
$$\mathbf{P}_t^k = \beta^t\mathbf{P}_t^{k-} + (1-\beta^t)[\mathbf{I} - \mathbf{P}_k\mathbf{H}]\mathbf{P}_t^{k-} + \mathbf{K}_k\mathbf{P}_{\nu^t}\mathbf{K}_k^T$$

where

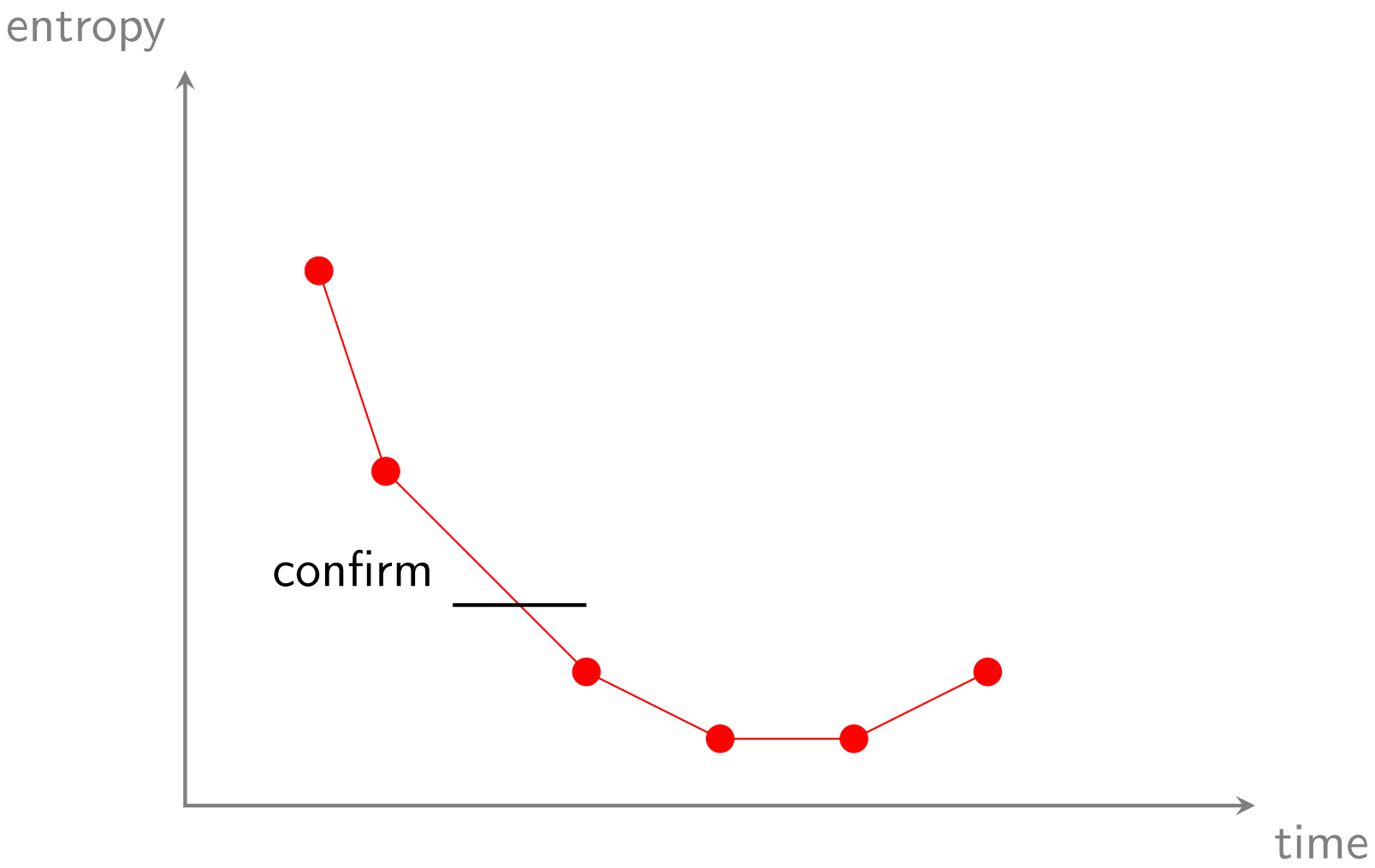
$$\beta_t = 1 - \sum_{j=1}^{m_k} \beta_j^t$$
$$\mathbf{P}_{\nu^t} = \sum^{m_k} \beta_j^t \nu_j^t (\nu_j^t)^T - \nu^t(\nu^t)^T$$

DOT – Entropy based track management

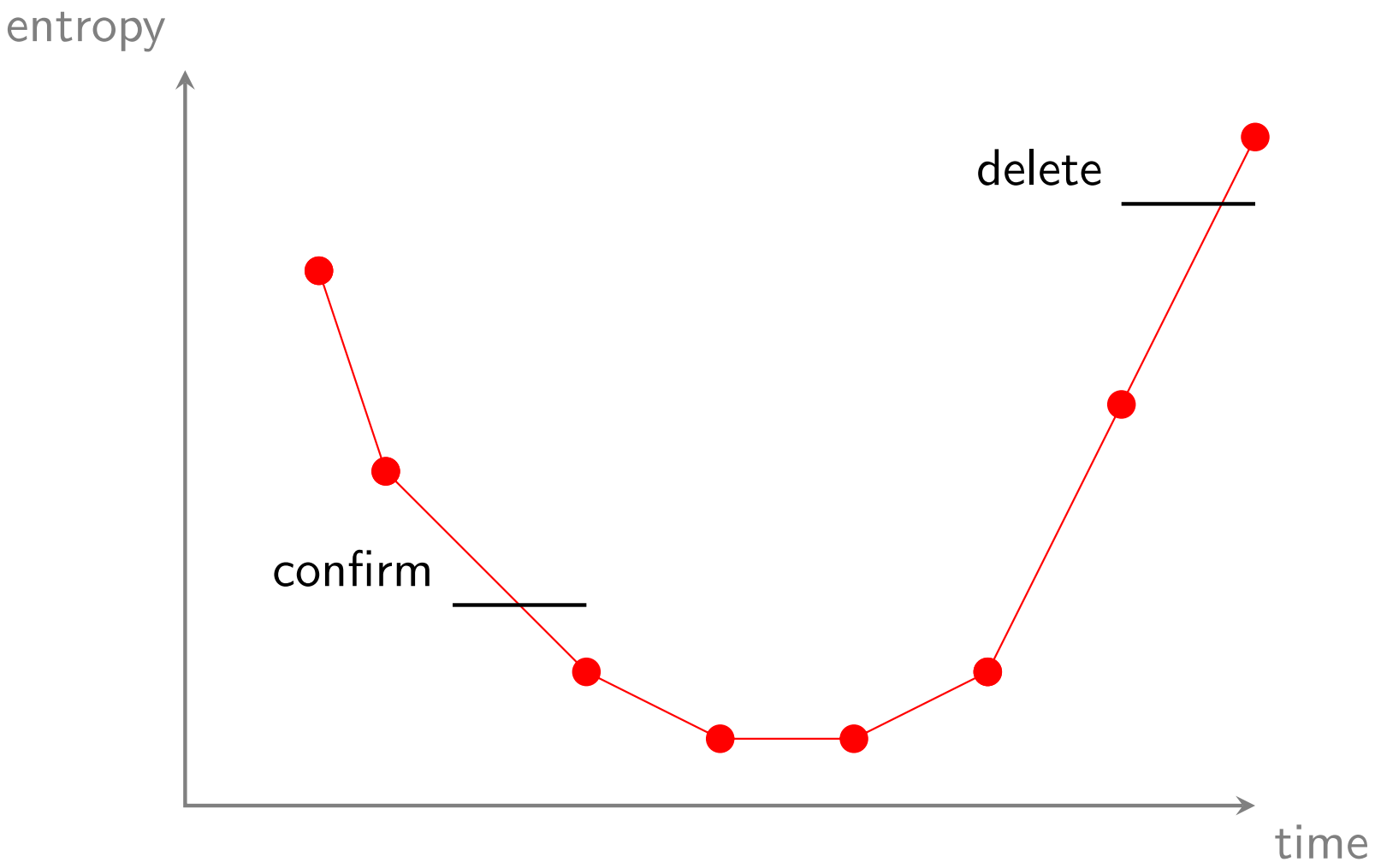
DOT – Entropy based track management



DOT – Entropy based track management



DOT – Entropy based track management



DOT – Entropy based track management

- Rényi entropy

$$H_2(\hat{\mathbf{x}}_t) = -\log \int p(\hat{\mathbf{x}}_t)^2 \, \mathrm{d}\hat{\mathbf{x}}_t$$

DOT – Entropy based track management

- Rényi entropy

$$H_2(\hat{\mathbf{x}}_t) = -\log \int p(\hat{\mathbf{x}}_t)^2 \, \mathrm{d}\hat{\mathbf{x}}_t$$

- in the case of Gaussian distribution

$$\begin{aligned} H_2(\hat{\mathbf{x}}_t) &= \frac{n}{2} \log 4\pi + \frac{1}{2} \log |\mathbf{P}_t| \\ &= \frac{n}{2} \log 4\pi + \frac{1}{2} \log \prod_{i=1}^{n} \lambda_i \end{aligned}$$

DOT – Entropy based track management

- Rényi entropy

$$H_2(\hat{\mathbf{x}}_t) = -\log \int p(\hat{\mathbf{x}}_t)^2 \, \mathrm{d}\hat{\mathbf{x}}_t$$

- in the case of Gaussian distribution

$$\begin{aligned} H_2(\hat{\mathbf{x}}_t) &= \frac{n}{2} \log 4\pi + \frac{1}{2} \log |\mathbf{P}_t| \\ &= \frac{n}{2} \log 4\pi + \frac{1}{2} \log \prod_{i=1}^{n} \lambda_i \end{aligned}$$

- modification in accordance to sensor characteristics

$$H_{2,mod}(\hat{\mathbf{x}}_t) = \frac{n}{2} \log 4\pi + \frac{1}{2} \log \prod_{i=1}^{n} \lambda_{i,mod}$$

where

$$\lambda_{i,mod} = \lambda_i \left[\alpha + (1-\alpha) \frac{1 - ||Proj(\mathbf{l}_{i,\mathbf{v}}, \mathbf{v})||}{||\mathbf{l}_{i,\mathbf{v}}||} \frac{||\mathbf{v}||}{v_{max}}\right]$$

DOT – modified track management

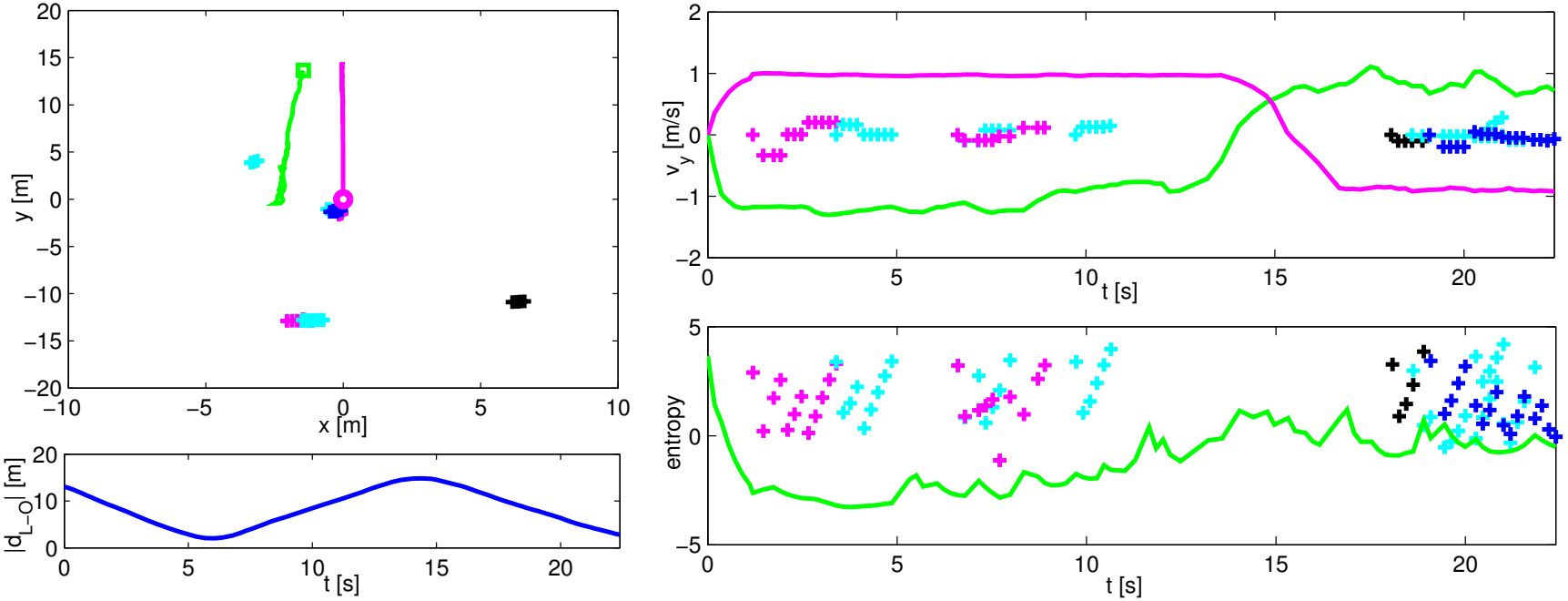
- using modified eigenvalues – tolerance onto *added* process noise

$$\lambda_{i,mod} = \lambda_i \left[\alpha + (1-\alpha)\,\frac{1 - ||Proj(\mathbf{l}_{i,\mathbf{v}}, \mathbf{v})||}{||\mathbf{l}_{i,\mathbf{v}}||}\,\frac{||\mathbf{v}||}{v_{max}}\right]$$

$\Rightarrow$ if $||\mathbf{v}|| < v_{max}$ then $H_{2,mod} < H_2$

DOT – Experimental Results

DOT – Experimental Results



Content

Conclusion – DATMO

Conclusion – DATMO

- Dynamic Objects Detection pipeline
  - processing in 3D – voxelization
  - no map of environment – direct point cloud comparison
  - no advanced localization sensing systems – ICP + odometry

Conclusion – DATMO

- Dynamic Objects Detection pipeline
  - processing in 3D – voxelization
  - no map of environment – direct point cloud comparison
  - no advanced localization sensing systems – ICP + odometry
- Dynamic Objects Tracking
  - Joint Probabilistic Data Association Filter
  - modified entropy based track management

Thank You for the attention! ☺

Azim, A. and Aycard, O. (2012).

Detection, classification and tracking of moving objects in a 3d environment.

In *Intelligent Vehicles Symposium*, pages 802–807. IEEE.

Kaestner, R., Engelhard, N., Triebel, R., and Siegwart, R. (2010).

A bayesian approach to learning 3d representations of dynamic environments.

In *Proceedings of The 12th International Symposium on Experimental Robotics (ISER)*, Berlin. Springer Press.

Kaestner, R., Maye, J., and Siegwart, R. (2012).

Generative object detection and tracking in 3d range data.

In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*.

Moosmann, F. and Fraichard, T. (2010).

Motion estimation from range images in dynamic outdoor scenes.

In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 142–147.

Moosmann, F., Pink, O., and Stiller, C. (2009).

Segmentation of 3d lidar data in non-flat urban environments using a local convexity criterion.

In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 215–220.