Lecture 4: Random finite sets Version May 27, 2019

Multi-Object Tracking

Lennart Svensson

Section 1:
Introduction to week 4

Multi-Object Tracking

Lennart Svensson

Random finite sets: introduction

Multi-Object Tracking

Lennart Svensson

PREVIOUS WEEK

- **State is a matrix**

$$X_k = \begin{bmatrix} x_k^1 & x_k^2 & \dots & x_k^n \end{bmatrix}.$$

  - Number of objects, n, is **known and constant**.
  - Objects are present at all times.

- **Measurement is a matrix**

$$Z_k = \Pi(O_k, C_k).$$

  - Here O_k and C_k are independent matrices representing object and clutter detections.

OBSERVATIONS AND REFLECTIONS (FROM VIDEO)

Properties

- Objects appear and disappear.
- We care about states of **present objects**.
- Objects are not ordered.

STATE REPRESENTATION

State representation

- We use a set

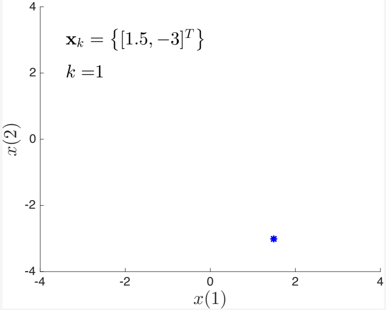
$$\mathbf{x}_k = \{x_k^1, \ldots, x_k^{n_k}\}$$

  to represent the state.

- Why sets?
  - sets are invariant to order,
  - easy to add/remove elements,
  - the set of state vectors is our quantity of interest,
  - one-to-one relation between physical reality and the set.

A possible state sequence

- A state sequence in 2D.
- Two objects present from time 3 to 23.



BAYESIAN FILTERING RECURSION FOR MOT

- Both $\mathbf{x}_k$ and $\mathbf{z}_k$ are random finite sets (RFSs).

Bayesian filtering recursions

Prediction: $$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) \, \delta \mathbf{x}_{k-1}$$

Update: $$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{1:k-1})}{\int p(\mathbf{z}_k | \mathbf{x}'_k) p(\mathbf{x}'_k | \mathbf{z}_{1:k-1}) \, \delta \mathbf{x}'_k}.$$

- **Pros:**
  - unified framework to model all aspects of MOT: appearing/disappearing objects, object motions and measurements;
  - powerful tools for derivations;
  - metrics for performance evaluation;
  - yields Bayes optimal solutions (in theory).

BAYESIAN FILTERING RECURSION FOR MOT

- Both $\mathbf{x}_k$ and $\mathbf{z}_k$ are random finite sets (RFSs).

Bayesian filtering recursions

Prediction: $$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) \, \delta \mathbf{x}_{k-1}$$

Update: $$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{1:k-1})}{\int p(\mathbf{z}_k | \mathbf{x}'_k) p(\mathbf{x}'_k | \mathbf{z}_{1:k-1}) \, \delta \mathbf{x}'_k}.$$

- New things to learn about:
  - What is an RFS? Integrals? Distributions? Models? Approximations? MOT algorithms? Metrics? ...

Section 2:
Intro to RFSs

Multi-Object Tracking

Lennart Svensson

Random finite sets

Multi-Object Tracking

Lennart Svensson

RANDOM FINITE SETS (RFSs)

Random finite sets: definition

A random variable whose possible outcomes are sets with a finite number of unique elements.

- In an RFS, $\mathbf{x} = \{x^1, \ldots, x^n\}$, both the number of elements and the elements themselves may be random.
- The elements of an RFS belong to some space, D, often $D = \mathbb{R}^{n_x}$ or $D = \mathbb{R}^{n_z}$.
- The RFS itself takes values $\mathbf{x} \in \mathcal{F}(D)$, where $\mathcal{F}(D)$ is the set of all finite subsets of D.

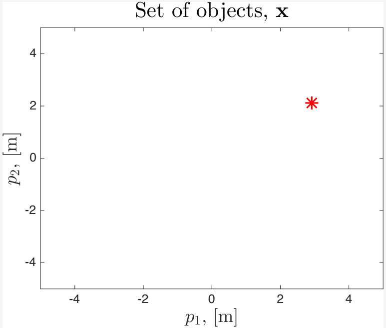
RANDOM SETS OF OBJECT STATES

- Let $\mathbf{x}_k$ be an RFS: the set of object states at time k.
- Elements of $\mathbf{x}_k$ belong to $\mathbb{R}^{n_x}$.

Possible realisations

$\mathbf{x} = \emptyset$	no objects present
$\mathbf{x} = \{x^1\}$	one object, state x^1
$\mathbf{x} = \{x^1, x^2\}$	two objects, states $x^1 \neq x^2$
$\vdots$	

Example, samples of $\mathbf{x}_k$



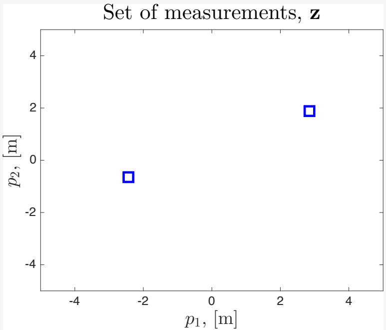
RANDOM SETS OF MEASUREMENTS

- Let $\mathbf{z}_k$ be an RFS: the set of measurements at time k.
- Elements of $\mathbf{z}_k$ belong to $\mathbb{R}^{n_z}$.

Possible realisations

$\mathbf{z} = \emptyset$	no measurements
$\mathbf{z} = \{z^1\}$	one measurement, z^1
$\mathbf{z} = \{z^1, z^2\}$	two measurements, $z^1 \neq z^2$
$\vdots$	

Example, samples of $\mathbf{z}_k$



A RECAP ON SET PROPERTIES

- Sets are **equal** if they contain the same elements.
- Sets are **invariant to order**, e.g., $\{1, 2, 3\} = \{2, 1, 3\}$.
- RFSs do not contain repeated elements, i.e., an RFS is never, e.g., $\{a, b, b, c\}$.
- A set that does not contain any elements is **empty**, denoted $\emptyset$ or (sometimes) $\{\}$.
- The **union** of two sets $\mathbf{a}$ and $\mathbf{b}$ is denoted $\mathbf{a} \cup \mathbf{b} \triangleq \{x : x \in \mathbf{a} \text{ or } x \in \mathbf{b}\}$, e.g., $\mathbf{a} = \{1, 2\}, \mathbf{b} = \{2, 3\} \Rightarrow \mathbf{a} \cup \mathbf{b} = \{1, 2, 3\}$.
- The **intersection** of two sets $\mathbf{a}$ and $\mathbf{b}$ is denoted $\mathbf{a} \cap \mathbf{b} \triangleq \{x : x \in \mathbf{a} \text{ and } x \in \mathbf{b}\}$, e.g., $\mathbf{a} = \{1, 2\}, \mathbf{b} = \{2, 3\} \Rightarrow \mathbf{a} \cap \mathbf{b} = \{2\}$.
- Two sets are **disjoint** if their intersection is empty, e.g., $\mathbf{a} = \{1, 2, 3\}$ and $\mathbf{b} = \{4, 5, 6\}$ are disjoint since $\mathbf{a} \cap \mathbf{b} = \emptyset$.
- The **cardinality** of a set $\mathbf{a}$ is denoted $|\mathbf{a}|$. For a finite set, this is the number of unique elements in $\mathbf{a}$, e.g., $\mathbf{a} = \{4, 5, 6\} \Rightarrow |\mathbf{a}| = 3$.

Multiobject pdfs

Multi-Object Tracking

Lennart Svensson

MULTIOBJECT PDFs

Multiobject pdfs

We use the multiobject probability density function (pdf) of an RFS, $\mathbf{x}$, to describe its distribution.

- A multiobject pdf, $p_{\mathbf{x}}(\{x^1, \dots, x^n\})$, is a non-negative function on sets that integrates to one.
- It captures both the distribution over cardinality and the distribution over the elements of the set (given the cardinality).
- Since sets are invariant to order so are multiobject pdfs, e.g.,

$$p_{\mathbf{x}}(\{x^1, x^2\}) = p_{\mathbf{x}}(\{x^2, x^1\}).$$

- Whenever we write $\{x^1, \dots, x^n\}$, we assume that $|x^1, \dots, x^n| = n$.

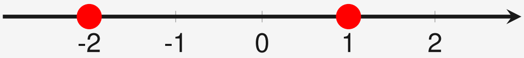
MULTIOBJECT PDFs: EXAMPLES

Example 1

- If $x \sim \mathcal{N}(0, 1)$ and $\mathbf{x} = \{x\}$ then

$$p_{\mathbf{x}}(\mathbf{x}) = \begin{cases} \mathcal{N}(v; 0, 1) & \text{if } \mathbf{x} = \{v\} \\ 0 & \text{if } |\mathbf{x}| \neq 1. \end{cases}$$

- For instance, $p_{\mathbf{x}}(\{1, -2\}) = 0$



and $p_{\mathbf{x}}(\{-0.6\}) = \mathcal{N}(-0.6; 0, 1) \approx 0.33.$

MULTIOBJECT PDFs: EXAMPLES

Example 2

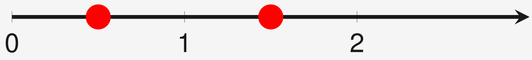
- If $x^1 \sim \text{unif}(0,1)$ and $x^2 \sim \text{unif}(1,2)$ are independent and $\mathbf{x} = \{x^1, x^2\}$, then

$$p_{\mathbf{x}}(\mathbf{x}) = \begin{cases} p_1(v^1)p_2(v^2) + p_1(v^2)p_2(v^1) & \text{if } \mathbf{x} = \{v^1, v^2\} \\ 0 & \text{if } |\mathbf{x}| \neq 2, \end{cases}$$

where

$$p_1(x) = \begin{cases} 1 & \text{if } 0 < x < 1 \\ 0 & \text{otherwise,} \end{cases} \qquad p_2(x) = \begin{cases} 1 & \text{if } 1 < x < 2 \\ 0 & \text{otherwise.} \end{cases}$$

- For instance, $p_{\mathbf{x}}(\{1.5, 0.5\}) = p_1(1.5)p_2(0.5) + p_1(0.5)p_2(1.5) = 0 + 1 = 1$.



INTERPRETATION OF MULTIOBJECT PDFs, $D = \mathbb{R}$

- For real valued random variables

$$\Pr\left[x \in (v, v+\Delta v)\right] = \int_{v}^{v+\Delta v} p_x(s)\,\mathrm{d}s \approx p_x(v)\,\Delta v, \qquad (\Delta v \text{ “small”}).$$

Interpretation

- If $\Delta v^1, \ldots, \Delta v^n$ are “small”

$$p_{\mathbf{x}}(\{v^1, \ldots, v^n\}) \times \Delta v^1 \times \cdots \times \Delta v^n$$

is (approximately) the probability that $\mathbf{x}$ contains precisely one element in each of the (disjoint) intervals $(v^1, v^1 + \Delta v^1), \ldots, (v^n, v^n + \Delta v^n)$.

INTERPRETATION OF MULTIOBJECT PDFs, $D = \mathbb{R}$

Example 2, revisited

- Suppose $v^1 = 1.5$, $v^2 = 0.5$ and $\Delta v^1 = \Delta v^2 = 0.2$. Then,

$$p_{\mathbf{x}}(\{v^1, v^2\})\,\Delta v^1 \Delta v^2 = 1 \times 0.2 \times 0.2 = 0.2^2.$$

- Reasonable? Is this the probability that **x** contains precisely one element in $(0.5, 0.7)$ and a second element in $(1.5, 1.7)$?
- Yes! That probability is

$$\Pr\left[x^1 \in (0.5, 0.7), x^2 \in (1.5, 1.7)\right] = \Pr\left[x^1 \in (0.5, 0.7)\right] \Pr\left[x^2 \in (1.5, 1.7)\right] = 0.2^2.$$

MULTIOBJECT PDFS VS ORDERED DENSITIES

Multiobject pdfs vs ordered densities

- Suppose $\mathbf{x} = \{x^1, \ldots, x^n\}$ is an RFS. If $X = \Pi([x^1, \ldots, x^n])$, then
$$p_X\left(\left[x^1, \ldots, x^n\right]\right) = \frac{1}{n!} p_{\mathbf{x}}\left(\left\{x^1, \ldots, x^n\right\}\right).$$

- **Note:** we can order $x^1, \ldots, x^n$ in $n!$ different ways.
  This gives $n!$ different matrices that correspond to the same set!
- **Example:** if $n = 2$, $p_{\mathbf{x}}(\{x^1, x^2\}) = p_X([x^1, x^2]) + p_X([x^2, x^1]) = 2p_X([x^1, x^2])$.

Example 2, revisited

- If $x^1 \sim \text{unif}(0, 1)$ and $x^2 \sim \text{unif}(1, 2)$ are independent and $X = \Pi(x^1, x^2)$, then
$$p_X(X) = \begin{cases} \frac{1}{2} p_1(v^1) p_2(v^2) + \frac{1}{2} p_1(v^2) p_2(v^1) & \text{if } X = [v^1, v^2] \\ 0 & \text{if } |X| \neq 2, \end{cases}$$
  where $p_1(x)$ and $p_2(x)$ are the pdfs of x^1 and x^2, respectively.

The convolution formula

Multi-Object Tracking

Lennart Svensson

CONVOLUTION FORMULA FOR DISCRETE RANDOM VARIABLES (1)

Flipping two coins

- Let us flip a fair coin twice and let x be total number of heads.
- Let x_1 be the number of heads in first flip and x_2 in the second: $x = x_1 + x_2$.
- We get, $\Pr[x_i = j] = 1/2$ for $i = 1, 2$ and $j = 0, 1$. Also,

$$\Pr[x = 0] = \Pr[x_1 = 0] \Pr[x_2 = 0] = 0.5^2 = 0.25,$$

$$\Pr[x = 2] = \Pr[x_1 = 1] \Pr[x_2 = 1] = 0.5^2 = 0.25,$$

$$\Pr[x = 1] = \Pr[x_1 = 0] \Pr[x_2 = 1] + \Pr[x_1 = 1] \Pr[x_2 = 0] = 0.5^2 + 0.5^2 = 0.5.$$

CONVOLUTION FORMULA FOR DISCRETE RANDOM VARIABLES (2)

Rolling a die twice

- Let x_1 be the number of dots in first roll, x_2 in the second and let $x = x_1 + x_2$.
- We get, e.g.,

$$\Pr[x = 4] = p_{x_1}(3)p_{x_2}(1) + p_{x_1}(2)p_{x_2}(2) + p_{x_1}(1)p_{x_2}(3) = \frac{3}{36}.$$

Convolution formula for discrete random variable.

- Suppose x_1 and x_2 are independent, integer valued, random variables.
- If $x = x_1 + x_2$,

$$\Pr[x = v] = \sum_{s=-\infty}^{\infty} p_{x_1}(s)p_{x_2}(v - s).$$

- This is the **convolution** $\Pr[x = v] = p_{x_1} * p_{x_2}(v)$.

UNION OF TWO INDEPENDENT RFSs (1)

Two independent, scalar, RFSs

- Suppose $\mathbf{x}^1$ and $\mathbf{x}^2$ are independent RFSs.
- If $\mathbf{x} = \mathbf{x}^1 \cup \mathbf{x}^2$:

$$p_{\mathbf{x}}(\{1.3\}) = p_{\mathbf{x}^1}(\emptyset)p_{\mathbf{x}^2}(\{1.3\}) + p_{\mathbf{x}^1}(\{1.3\})p_{\mathbf{x}^2}(\emptyset).$$

Why ignore $\mathbf{x}^1 = \mathbf{x}^2 = \{1.3\}$? (Brief intuitive argument)

- The above multiobject pdfs are related to probabilities, e.g.,:

$$\begin{aligned}\Pr[\mathbf{x} = \{\tilde{x}\}, \tilde{x} \in (1.2, 1.4)] = &\Pr[\mathbf{x}^1 = \emptyset, \mathbf{x}^2 = \{\tilde{x}\}, \tilde{x} \in (1.2, 1.4)] \\ &+ \Pr[\mathbf{x}^1 = \{\tilde{x}\}, \mathbf{x}^2 = \emptyset, \tilde{x} \in (1.2, 1.4)].\end{aligned}$$

- However, since

$$\Pr[\mathbf{x}^1 = \mathbf{x}^2 = \{\tilde{x}\}, \tilde{x} \in (1.2, 1.4)] = 0$$

the corresponding density is also zero.

UNION OF TWO INDEPENDENT RFSs (2)

Two independent, scalar, RFSs (continued)

- Suppose $\mathbf{x}^1$ and $\mathbf{x}^2$ are independent RFSs.
- If $\mathbf{x} = \mathbf{x}^1 \cup \mathbf{x}^2$:

$$\begin{aligned} p_{\mathbf{x}}(\{1.3, 2.7\}) =& p_{\mathbf{x}^1}(\emptyset) p_{\mathbf{x}^2}(\{1.3, 2.7\}) + p_{\mathbf{x}^1}(\{1.3, 2.7\}) p_{\mathbf{x}^2}(\emptyset) \\ &+ p_{\mathbf{x}^1}(\{1.3\}) p_{\mathbf{x}^2}(\{2.7\}) + p_{\mathbf{x}^1}(\{2.7\}) p_{\mathbf{x}^2}(\{1.3\}). \end{aligned}$$

Convolution formula for union of two RFSs

- If $\mathbf{x}^1$ and $\mathbf{x}^2$ are independent RFSs, then $\mathbf{x} = \mathbf{x}^1 \cup \mathbf{x}^2$ has the multiobject pdf

$$p_{\mathbf{x}}(\mathbf{x}) = \sum_{\mathbf{x}^1 \subseteq \mathbf{x}} p_{\mathbf{x}^1}(\mathbf{x}^1) p_{\mathbf{x}^2}(\mathbf{x} \setminus \mathbf{x}^1).$$

SUMS OVER MUTUALLY DISJOINT SETS

- To generalize the formula to unions of n RFSs, let
$$\sum_{\mathbf{x}^1 \uplus \cdots \uplus \mathbf{x}^n = \mathbf{x}}$$
denote summation **over all mutually disjoint (and possibly empty) sets** $\mathbf{x}^1, \ldots, \mathbf{x}^n$ whose union is $\mathbf{x}$. **Recall:** $\mathbf{x}^1$ and $\mathbf{x}^2$ are **disjoint** if $\mathbf{x}^1 \cap \mathbf{x}^2 = \emptyset$.

Examples of summations

$$\sum_{\mathbf{x}^1 \uplus \mathbf{x}^2 = \{1\}} f(\mathbf{x}^1, \mathbf{x}^2) = f(\{1\}, \emptyset) + f(\emptyset, \{1\})$$
$$\sum_{\mathbf{x}^1 \uplus \mathbf{x}^2 \uplus \mathbf{x}^3 = \{4\}} f(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3) = f(\{4\}, \emptyset, \emptyset) + f(\emptyset, \{4\}, \emptyset) + f(\emptyset, \emptyset, \{4\})$$
$$\sum_{\mathbf{x}^1 \uplus \mathbf{x}^2 = \{3,5\}} f(\mathbf{x}^1, \mathbf{x}^2) = f(\{3,5\}, \emptyset) + f(\emptyset, \{3,5\}) + f(\{3\}, \{5\}) + f(\{5\}, \{3\})$$

- **Note 1:** it holds that $\sum_{\mathbf{x}^1 \uplus \mathbf{x}^2 = \mathbf{x}} f(\mathbf{x}^1, \mathbf{x}^2) = \sum_{\mathbf{x}^1 \subseteq \mathbf{x}} f(\mathbf{x}^1, \mathbf{x} \setminus \mathbf{x}^1)$.

SUMS OVER MUTUALLY DISJOINT SETS

- To generalize the formula to unions of n RFSs, let

$$\sum_{\mathbf{x}^1 \uplus \cdots \uplus \mathbf{x}^n = \mathbf{x}}$$

denote summation **over all mutually disjoint (and possibly empty) sets** $\mathbf{x}^1, \ldots, \mathbf{x}^n$ whose union is $\mathbf{x}$. **Recall:** $\mathbf{x}^1$ and $\mathbf{x}^2$ are **disjoint** if $\mathbf{x}^1 \cap \mathbf{x}^2 = \emptyset$.

Examples of summations

$$\sum_{\mathbf{x}^1 \uplus \mathbf{x}^2 = \{1\}} f(\mathbf{x}^1, \mathbf{x}^2) = f(\{1\}, \emptyset) + f(\emptyset, \{1\})$$

$$\sum_{\mathbf{x}^1 \uplus \mathbf{x}^2 \uplus \mathbf{x}^3 = \{4\}} f(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3) = f(\{4\}, \emptyset, \emptyset) + f(\emptyset, \{4\}, \emptyset) + f(\emptyset, \emptyset, \{4\})$$

$$\sum_{\mathbf{x}^1 \uplus \mathbf{x}^2 = \{3,5\}} f(\mathbf{x}^1, \mathbf{x}^2) = f(\{3,5\}, \emptyset) + f(\emptyset, \{3,5\}) + f(\{3\}, \{5\}) + f(\{5\}, \{3\})$$

- **Note 2:** every term in $\sum_{\mathbf{x}^1 \uplus \cdots \uplus \mathbf{x}^n = \mathbf{x}}$ assigns elements in $\mathbf{x}$ to $\mathbf{x}^1, \ldots, \mathbf{x}^n$.

CONVOLUTION FORMULA FOR INDEPENDENT RFSs

Convolution theorem for independent RFSs

- If $\mathbf{x}^1, \ldots, \mathbf{x}^n$ are independent RFSs, then $\mathbf{x} = \mathbf{x}^1 \cup \cdots \cup \mathbf{x}^n$ has the multiobject pdf

$$p_{\mathbf{x}}(\mathbf{x}) = \sum_{\mathbf{x}^1 \uplus \cdots \uplus \mathbf{x}^n = \mathbf{x}} \prod_{i=1}^{n} p_{\mathbf{x}^i}(\mathbf{x}^i),$$

where the summation is taken over all mutually disjoint (and possibly empty) sets $\mathbf{x}^1, \ldots, \mathbf{x}^n$ whose union is $\mathbf{x}$.

Union of three RFSs

- Suppose $\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3$ are independent RFSs.
- The multiobject pdf of $\mathbf{x} = \mathbf{x}^1 \cup \mathbf{x}^2 \cup \mathbf{x}^3$ then satisfies

$$p_{\mathbf{x}}(\{4\}) = p_{\mathbf{x}^1}(\{4\}) p_{\mathbf{x}^2}(\emptyset) p_{\mathbf{x}^3}(\emptyset) + p_{\mathbf{x}^1}(\emptyset) p_{\mathbf{x}^2}(\{4\}) p_{\mathbf{x}^3}(\emptyset) + p_{\mathbf{x}^1}(\emptyset) p_{\mathbf{x}^2}(\emptyset) p_{\mathbf{x}^3}(\{4\}).$$

CONVOLUTION FORMULA FOR INDEPENDENT RFSs

Example 2, revisited

- Suppose $\mathbf{x}^1$ and $\mathbf{x}^2$ are independent singletons, (for $i = 1, 2$)

$$p_{\mathbf{x}^i}(\mathbf{x}^i) = \begin{cases} p_i(x^i) & \text{if } \mathbf{x}^i = \{x^i\} \\ 0 & \text{if } |\mathbf{x}^i| \neq 1. \end{cases}$$

- If $\mathbf{x} = \mathbf{x}^1 \cup \mathbf{x}^2$,

$$\begin{aligned} p_{\mathbf{x}}(\{x^1, x^2\}) &= p_{\mathbf{x}^1}(\emptyset) p_{\mathbf{x}^2}(\{x^1, x^2\}) + p_{\mathbf{x}^1}(\{x^1, x^2\}) p_{\mathbf{x}^2}(\emptyset) \\ &\quad + p_{\mathbf{x}^1}(\{x^1\}) p_{\mathbf{x}^2}(\{x^2\}) + p_{\mathbf{x}^1}(\{x^2\}) p_{\mathbf{x}^2}(\{x^1\}) \\ &= p_1(x^1) p_2(x^2) + p_1(x^2) p_2(x^1). \end{aligned}$$

- We also note that $p_{\mathbf{x}}(\mathbf{x}) = 0$ if $|\mathbf{x}| \neq 2$.

Set integrals

Multi-Object Tracking

Lennart Svensson

SET INTEGRALS

Set integrals: definition

- For $f : \mathcal{F}(D) \to \mathbb{R}$, the set integral is defined as

$$\int f(\mathbf{x})\,\delta\mathbf{x} = \sum_{i=0}^{\infty} \frac{1}{i!} \int f(\{x^1, \ldots, x^i\})\, dx^1 \cdots dx^i$$

$$= f(\emptyset) + \sum_{i=1}^{\infty} \frac{1}{i!} \int f(\{x^1, \ldots, x^i\})\, dx^1 \cdots dx^i.$$

Example 1, revisited

- Any multiobject pdf must integrate to 1. For $p_{\mathbf{x}}(\mathbf{x}) = \begin{cases} \mathcal{N}(x; 0, 1) & \text{if } \mathbf{x} = \{x\} \\ 0 & \text{if } |\mathbf{x}| \neq 1, \end{cases}$

  the set integral is $\int p_{\mathbf{x}}(\mathbf{x})\,\delta\mathbf{x} = \int p_{\mathbf{x}}(\{x^1\})\mathrm{d}x^1 = \int \mathcal{N}(x^1; 0, 1)\,\mathrm{d}x^1 = 1.$

EXAMPLE 2 AND INTUITION FOR $1/i!$

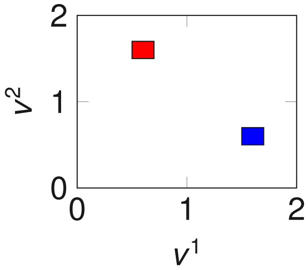
- In example 2, we had

$$p_{\mathbf{x}}(\mathbf{x}) = \begin{cases} p_1(v^1)p_2(v^2) + p_1(v^2)p_2(v^1) & \text{if } \mathbf{x} = \{v^1, v^2\} \\ 0 & \text{if } |\mathbf{x}| \neq 2. \end{cases}$$

Set integral of $p_{\mathbf{x}}(\mathbf{x})$

$$\begin{aligned} \int p_{\mathbf{x}}(\mathbf{x})\delta\mathbf{x} &= \sum_{i=0}^{\infty} \frac{1}{i!} \int p_{\mathbf{x}}(\{v^1, \ldots, v^2\})\mathrm{d}v^1 \cdots \mathrm{d}v^i \\ &= \frac{1}{2} \int p_{\mathbf{x}}(\{v^1, v^2\})\mathrm{d}v^1\mathrm{d}v^2 \\ &= \frac{1}{2} \int \left(p_1(v^1)p_2(v^2) + p_1(v^2)p_2(v^1)\right) \mathrm{d}v^1\mathrm{d}v^2 \\ &= \frac{2}{2} \int p_1(v^1)\mathrm{d}v^1 \int p_2(v^2)\mathrm{d}v^2 = 1 \end{aligned}$$

- **Why $\frac{1}{2}$?** integrating over blue and red areas $\Rightarrow$ account for **same set twice**.

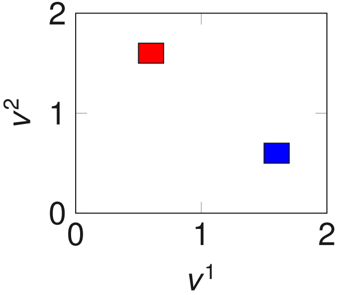


ORDERED STATES AND INTUITION FOR $1/i!$

- For the above toy example,

$$\int_{v^1>v^2} p_{\mathbf{x}}(\{v^1, v^2\})\mathrm{d}v^1\mathrm{d}v^2 = \frac{1}{2}\int p_{\mathbf{x}}(\{v^1, v^2\})\mathrm{d}v^1\mathrm{d}v^2.$$

- Integrating over $\{(v^1, v^2) : v^1 > v^2\}$ means that we integrate over **every set precisely one time**.



- More generally, for scalar states, it holds that

$$\int_{x^1>\cdots>x^i} f(\{x^1, \ldots, x^i\})\, dx^1 \cdots dx^i = \frac{1}{i!}\int f(\{x^1, \ldots, x^i\})\, dx^1 \cdots dx^i.$$

- What about when the states are vectors?
  Left hand side does not generalize easily. Instead **we use the expression with $1/i!$**.

SET INTEGRALS AND EXPECTED VALUES

Expected values

- For $f : \mathcal{F}(D) \to \mathbb{R}$, the expected value is

$$\mathbb{E}\left[f(\mathbf{x})\right] = \int f(\mathbf{x}) p_{\mathbf{x}}(\mathbf{x})\, \delta\mathbf{x} = \sum_{i=0}^{\infty} \frac{1}{i!} \int f(\{x^1, \ldots, x^i\}) p_{\mathbf{x}}(\{x^1, \ldots, x^i\})\, dx^1 \cdots dx^i.$$

- The expected value appears, e.g., in the Chapman-Kolmogorov equation

$$p(\mathbf{x}_k \middle| \mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k \middle| \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} \middle| \mathbf{z}_{1:k-1})\, \delta\mathbf{x}_{k-1}.$$

- **Note:** the expected value of $\mathbf{x}$ is undefined.
  Why? We cannot add (average) sets, e.g., $\{0.3, 0.7\} + \{1\} + \{2, 0\}$ is not defined.

CARDINALITY DISTRIBUTIONS

Cardinality distributions

- The cardinality distribution of an RFS, $\mathbf{x} \sim p_{\mathbf{x}}(\cdot)$, is
$$p_{\mathbf{x}}(n) = \Pr\left[|\mathbf{x}| = n\right].$$

- Let the Kronecker delta function be denoted $\delta_i = \begin{cases} 1 & \text{if } i = 0 \\ 0 & \text{otherwise.} \end{cases}$
- It then holds that
$$\begin{aligned} \Pr\left[|\mathbf{x}| = n\right] &= \mathbb{E}\left[\delta_{n-|\mathbf{x}|}\right] \\ &= \sum_{i=0}^{\infty} \frac{1}{i!} \int \delta_{n-i} p_{\mathbf{x}}(\{x^1, \dots, x^i\})\, dx^1 \cdots dx^i \\ &= \frac{1}{n!} \int p_{\mathbf{x}}(\{x^1, \dots, x^n\})\, dx^1 \cdots dx^n. \end{aligned}$$
- **Note:** $\mathbb{E}\left[\delta_{n-|\mathbf{x}|}\right] = 0 \times \Pr[\delta_{n-|\mathbf{x}|} = 0] + 1 \times \Pr[\delta_{n-|\mathbf{x}|} = 1] = \Pr\left[|\mathbf{x}| = n\right]$.

CARDINALITY DISTRIBUTIONS, EXAMPLE 1

- As a sanity check, let us compute the cardinality distribution in a trivial example.

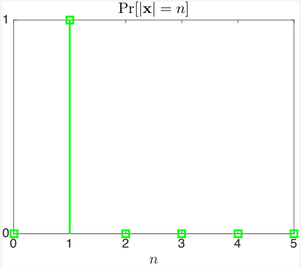
Example 1

- The cardinality distribution of

$$\mathbf{x} \sim p_{\mathbf{x}}(\mathbf{x}) = \begin{cases} \mathcal{N}(x; 0, 1) & \text{if } \mathbf{x} = \{x\} \\ 0 & \text{if } |\mathbf{x}| \neq 1, \end{cases}$$

is

$$\begin{aligned} \Pr\left[|\mathbf{x}| = n\right] &= \frac{1}{n!} \int p_{\mathbf{x}}(\{x^1, \ldots, x^n\})\, dx^1 \cdots dx^n \\ &= \begin{cases} \int \mathcal{N}(x^1; 0, 1) \mathrm{d}x^1 = 1 & \text{if } n = 1 \\ 0 & \text{if } n \neq 1. \end{cases} \end{aligned}$$



Belief mass functions and probability generating functionals

Multi-Object Tracking

Lennart Svensson

BELIEF MASS FUNCTIONS AND p.g.fl.s

- **Belief mass functions** and probability generating functionals (**p.g.fl.s**): alternative descriptors of a RFS $\mathbf{x}$.
- They are **very useful** for deriving expressions for models and filtering recursions:
  - mathematically rigorous,
  - “turn-the-crank” type of derivations,
  - transparent derivations.
- Important argument for using RFSs/point processes!
- On the other hand:
  - initially complicated to understand,
  - less intuitive compared to multiobject pdfs,
  - **beyond the scope of this course**.

Section 3:
Common RFSs

Multi-Object Tracking

Lennart Svensson

Poisson point processes

Multi-Object Tracking

Lennart Svensson

POISSON POINT PROCESSES

Poisson point process pdf

- The multiobject pdf of a Poisson point process (PPP) $\mathbf{x}$ is
$$p_{\mathbf{x}}(\mathbf{x}) = \exp\left(-\int \lambda(x')\,\mathrm{d}x'\right) \prod_{x \in \mathbf{x}} \lambda(x)$$
where $\lambda(x)$ is its intensity function.

- Using the Poisson rate $\bar{\lambda} = \int \lambda(x)\,\mathrm{d}x$ we can write the pdf as
$$p_{\mathbf{x}}(\{x_1, \dots, x_n\}) = \exp(-\bar{\lambda}) \prod_{i=1}^{n} \lambda(x_i).$$

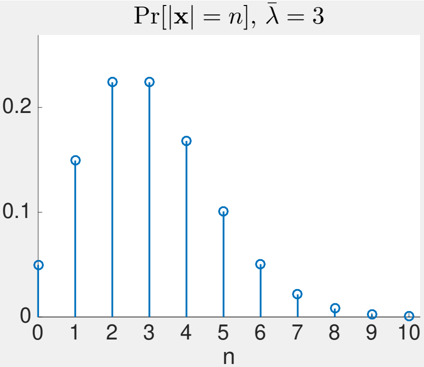
- PPPs are commonly used to **model**:
  - clutter detections, $D = \mathbb{R}^{n_z}$,
  - appearing objects, $D = \mathbb{R}^{n_x}$,
  - measurements from extended objects, $D = \mathbb{R}^{n_z}$.

PPP, CARDINALITY DISTRIBUTION

- Let us rederive the cardinality pmf for a PPP:

$$
\begin{aligned}
\Pr\left[|\mathbf{x}| = n\right] &= \frac{1}{n!}\int p_{\mathbf{x}}(\{x_1,\ldots,x_n\})\mathrm{d}x_1\cdots\mathrm{d}x_n \\
&= \frac{1}{n!}\int \exp(-\bar{\lambda})\lambda(x_1)\cdots\lambda(x_n)\mathrm{d}x_1\cdots\mathrm{d}x_n \\
&= \frac{1}{n!}\exp(-\bar{\lambda})\prod_{i=1}^{n}\int\lambda(x_i)\mathrm{d}x_i \\
&= \frac{1}{n!}\exp(-\bar{\lambda})\bar{\lambda}^n \\
&= \mathrm{Po}(n;\bar{\lambda})
\end{aligned}
$$

Example:



- This confirms that the **cardinality is Poisson distributed**.

PPP: GENERATING SAMPLES

Algorithm Sampling a PPP

1: Initialize $\mathbf{x} = \emptyset$

2: Generate $n \sim \text{Po}(\bar{\lambda})$

3: **for** $i = 1$ to n **do**

4: Generate $x_i \sim \frac{\lambda(\cdot)}{\bar{\lambda}}$

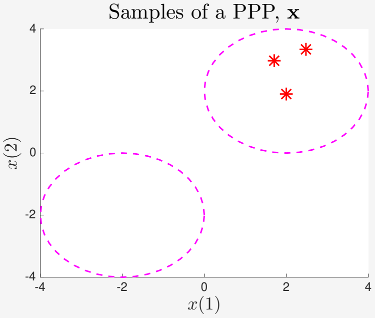
5: Set $\mathbf{x} = \mathbf{x} \bigcup \{x_i\}$

6: **end for**

Example: PPP samples

- Suppose

$$\lambda(x) = 4\mathcal{N}\left(x; \begin{bmatrix} 3 \\ 3 \end{bmatrix}, \mathbf{I}\right) + \mathcal{N}\left(x; \begin{bmatrix} -3 \\ -3 \end{bmatrix}, \mathbf{I}\right).$$



Bernoulli RFSs

Multi-Object Tracking

Lennart Svensson

BERNOULLI RFSs

Bernoulli RFSs

- A Bernoulli RFS (or a Bernoulli process) $\mathbf{x}$ has the multiobject pdf

$$p_{\mathbf{x}}(\mathbf{x}) = \begin{cases} 1-r & \text{if } \mathbf{x} = \emptyset \\ r\, p_x(x) & \text{if } \mathbf{x} = \{x\} \\ 0 & \text{if } |\mathbf{x}| > 1, \end{cases}$$

where $0 \leq r \leq 1$ and $p_x(x)$ is a pdf.

- It is easy to show that

$$\Pr[|\mathbf{x}| = n] = \begin{cases} 1-r & \text{if } n = 0 \\ r & \text{if } n = 1 \\ 0 & \text{if } n > 1. \end{cases}$$

- Bernoulli RFSs are used to **model**, e.g.,
  - measurements from a single object, $D = \mathbb{R}^{n_z}$,
  - a potential object, $D = \mathbb{R}^{n_x}$.

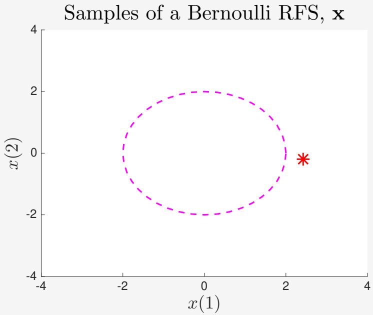
BERNOULLI RFSs: GENERATING SAMPLES

Algorithm Sampling Bernoulli RFSs

1: Initialize $\mathbf{x} = \emptyset$
2: **if** rand<r **then**
3: $\quad x \sim p_x(\cdot)$
4: $\quad \mathbf{x} = \{x\}$
5: **end if**

Example: Bernoulli samples

- Suppose $\mathbf{x}$ is a Bernoulli RFS with $r = 0.7$ and $p_x(x) = \mathcal{N}(x; \mathbf{0}, \mathbf{I})$.



Multi-Bernoulli RFSs

Multi-Object Tracking

Lennart Svensson

MULTI-BERNOULLI RFSs

Multi-Bernoulli RFSs

- Suppose $\mathbf{x}_1, \ldots, \mathbf{x}_N$ are independent Bernoulli RFSs with multiobject pdfs $p_{\mathbf{x}_1}(\mathbf{x}_1), \ldots, p_{\mathbf{x}_N}(\mathbf{x}_N)$, respectively.
- Then $\mathbf{x} = \bigcup_{i=1}^{N} \mathbf{x}_i$ is a multi-Bernoulli (MB) RFS (or a multi-Bernoulli process) with multiobject pdf

$$p_{\mathbf{x}}(\mathbf{x}) = \sum_{\biguplus_{i=1}^{N} \mathbf{x}_i = \mathbf{x}} \prod_{j=1}^{N} p_{\mathbf{x}_j}(\mathbf{x}_j).$$

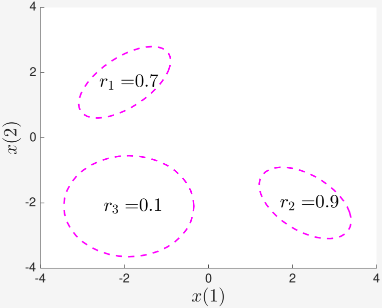
- MB RFSs are used to **model** potential objects, e.g.,
  - according to the posterior, $D = \mathbb{R}^{n_x}$,
  - appearing objects, $D = \mathbb{R}^{n_x}$.

A MULTI-BERNOULLI PROCESS EXAMPLE

- Suppose $p_{\mathbf{x}_i}(\mathbf{x}_i)$ is parametrised by r_i and $p_i(\cdot)$.

Example: a MB modelling potential objects

- Suppose $N = 3$, $r_1 = 0.7$, $r_2 = 0.9$ and $r_3 = 0.1$.
- Also, let $p_1(x)$, $p_2(x)$ and $p_3(x)$ be Gaussian, see figure.
- The MB RFS $\mathbf{x}$ represents that there are three potential objects.



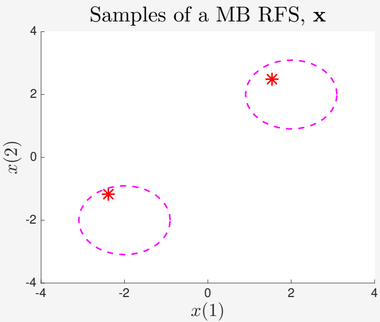
MULTI-BERNOULLI RFSs: GENERATING SAMPLES

Algorithm 3 Sampling a MB RFS

1: Initialize $\mathbf{x} = \emptyset$
2: **for** $i = 1$ to N **do**
3:     **if** rand$< r_i$ **then**
4:         $x_i \sim p_i(\cdot)$
5:         $\mathbf{x} = \mathbf{x} \cup \{x_i\}$
6:     **end if**
7: **end for**

Example: MB samples

- Suppose $N = 2$, $r_1 = r_2 = 0.8$, $p_1(x) = \mathcal{N}(x; [2 \quad 2]^T, 0.3\mathbf{I})$ and $p_2(x) = \mathcal{N}(x; [-2 \quad -2]^T, 0.3\mathbf{I})$.



MB VS POISSON

MB $\approx$ PPP?

- A Bernoulli RFS with $r < 0.1$ is approximately a PPP.
- $\Rightarrow$ a MB with $r_1, \ldots, r_N < 0.1$ is approximately a PPP.
- Any PPP can be approximated by a MB, but it may require a large N.

- Often **computationally efficient to use a PPP**.

Why use MB instead of PPP?

- If $\mathbf{x}$ is a PPP, both the mean and variance of $|\mathbf{x}|$ is $\bar{\lambda}$.
- Problematic if we are certain that there are, say, 10 objects present.
- The MB distribution is better at expressing the posterior in such situations.
- MB RFSs are not restricted to i.i.d. states
  $\Rightarrow$ "there is one object in each lane"!

Multi-Bernoulli mixture RFSs

Multi-Object Tracking

Lennart Svensson

MULTI-BERNOULLI MIXTURE RFSs

Multi-Bernoulli mixture RFSs

- Suppose $p^h_{\mathbf{x}_i}(\mathbf{x}_i)$ are Bernoulli multiobject pdfs for $i = 1, \ldots, N$ and $h = 1, \ldots, \mathcal{H}$.
- Then $\mathbf{x}$ is a multi-Bernoulli mixture (MBM) RFS (or a MBM process) if it has the multiobject pdf

$$p_{\mathbf{x}}(\mathbf{x}) = \sum_{h=1}^{\mathcal{H}} w_h \, p^h_{\mathbf{x}}(\mathbf{x}),$$

where $p^h_{\mathbf{x}}(\mathbf{x})$ is multi-Bernoulli pdf

$$p^h_{\mathbf{x}}(\mathbf{x}) = \sum_{\uplus_{i=1}^{N} \mathbf{x}_i = \mathbf{x}} \prod_{j=1}^{N} p^h_{\mathbf{x}_j}(\mathbf{x}_j),$$

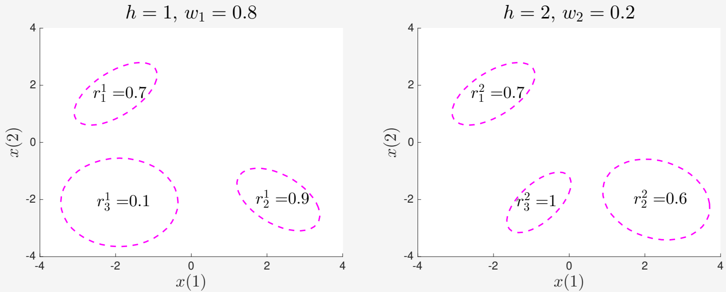
and $w_1, \ldots, w_{\mathcal{H}}$ are non-negative weights such that $\sum_{h=1}^{\mathcal{H}} w_h = 1$.

POSTERIOR UNCERTAINTIES AND MBMs

- MBM RFSs are used to **model**, e.g.,
  - posterior distribution of set of detected objects, $D = \mathbb{R}^{n_x}$, where $h = 1, \ldots, \mathcal{H}$ representation association hypotheses.

Example: an MBM modelling potential objects

- The MBM visualised below could model a posterior distribution with two hypotheses.



MBM RFSs: GENERATING SAMPLES (1)

- Suppose $w = [w_1, \ldots, w_{\mathcal{H}}]^T$.

Categorical distribution

- A random variable h is *categorical*, $h \sim \text{Cat}(w)$, if

$$\Pr[h = j] = w_j.$$

- **Example:** for $w = [1/6, \ldots, 1/6]^T$, $h \sim \text{Cat}(w)$ is rolling a fair dice.
- Sometimes easier to generate multinomial variables.

- Suppose $p^h_{\mathbf{x}_i}(\mathbf{x}_i)$ is parametrised by r^h_i and $p^h_i(\cdot)$.

Algorithm Sampling a MBM RFS

1: Initialize $\mathbf{x} = \emptyset$
2: Generate $h \sim \text{Cat}(w)$
3: **for** $i = 1$ to N **do**
4:     **if** rand$< r^h_i$ **then**
5:         $x_i \sim p^h_i(\cdot)$
6:         $\mathbf{x} = \mathbf{x} \cup \{x_i\}$
7:     **end if**
8: **end for**

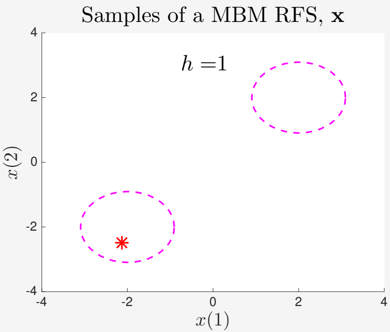
MBM RFSs: GENERATING SAMPLES (2)

Example: MBM samples

- Suppose $\mathcal{H} = 2$, $w_1 = 0.75$, $w_2 = 1 - w_1 = 0.25$ and that $r_i^h = 0.8$ for $i, h \in \{1, 2\}$.
- Also assume that

$$h = 1 : \begin{cases} p_1^1(x) = \mathcal{N}\left(x; [2 \quad 2]^T, 0.3\mathbf{I}\right) \\ p_2^1(x) = \mathcal{N}\left(x; [-2 \quad -2]^T, 0.3\mathbf{I}\right) \end{cases}$$

$$h = 2 : \begin{cases} p_1^2(x) = \mathcal{N}\left(x; [2 \quad -2]^T, 0.3\mathbf{I}\right) \\ p_2^2(x) = \mathcal{N}\left(x; [-2 \quad 2]^T, 0.3\mathbf{I}\right). \end{cases}$$



Section 4:
Standard models in MOT

Multi-Object Tracking

Lennart Svensson

Bayesian filtering recursions and models

Multi-Object Tracking

Lennart Svensson

MULTIOBJECT TRACKING

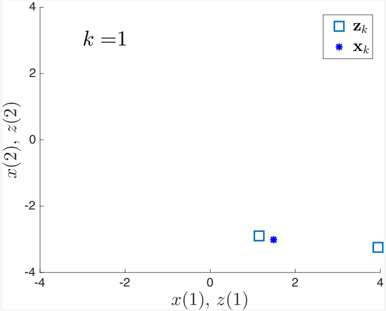
Objective

- Recursively compute $p(\mathbf{x}_k | \mathbf{z}_{1:k})$.

- The posterior can be used, e.g., to estimate $\mathbf{x}_k$.

A visualization

- Both states and measurements are in 2D (uncommon).



BAYESIAN FILTERING RECURSION FOR MOT

Bayesian filtering recursions

- The Chapman-Kolmogorov equation for prediction and Bayes' rule for update:

$$\text{prediction:} \qquad p(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} \,|\, \mathbf{z}_{1:k-1}) \, \delta\mathbf{x}_{k-1}$$

$$\text{update:} \qquad p(\mathbf{x}_k \,|\, \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k \,|\, \mathbf{x}_k) p(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1})}{\int p(\mathbf{z}_k \,|\, \mathbf{x}'_k) p(\mathbf{x}'_k \,|\, \mathbf{z}_{1:k-1}) \, \delta\mathbf{x}'_k}.$$

- We need models for

$$\text{motion:} \qquad p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})$$

$$\text{measurements:} \qquad p(\mathbf{z}_k \,|\, \mathbf{x}_k).$$

Measurement models – object detections

Multi-Object Tracking

Lennart Svensson

STANDARD MEASUREMENT MODEL

- Measurement model is as before.
- We assume

$$\mathbf{z}_k = \mathbf{o}_k \cup \mathbf{c}_k,$$

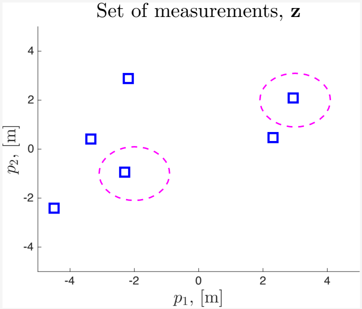
where $\mathbf{o}_k$ are object detections and $\mathbf{c}_k$ clutter detections.

- In this video, we present the **standard model** for

$$\mathbf{g}_k(\mathbf{o}_k | \mathbf{x}_k) = p(\mathbf{o}_k | \mathbf{x}_k).$$

Example, samples of $\mathbf{z}_k$

- Two objects, $P^{\mathrm{D}} = 0.95$, Gaussian $g_k(\cdot|x^1)$ and $g_k(\cdot|x^2)$ (see dashed ellipsoids), and $\bar{\lambda} = 2$.



OBJECT MEASUREMENTS: STANDARD ASSUMPTIONS

Single object measurement model

- An object with state x is detected with probability $P^{\mathrm{D}}(x)$.
- If detected, it generates a measurement from the single object measurement density $g_k(o|x)$.

In the presence of other objects:

- Conditioned on the object states, each object measurement is independent of all other objects and measurements (including clutter detections).
- Each measurement is the result of at most one object.

SINGLE OBJECT MEASUREMENT MODEL

Case 1: $\mathbf{x}_k = \emptyset$

$$\mathbf{g}_k(\mathbf{o}|\emptyset) = \begin{cases} 1 & \text{if } \mathbf{o} = \emptyset \\ 0 & \text{otherwise} \end{cases}$$

- **Note:** $\mathbf{o}_k|\mathbf{x}_k = \emptyset$ is a Bern. RFS with $r = 0$.

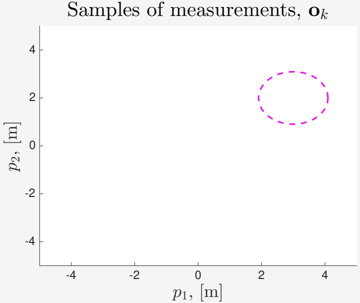
Case 2: $\mathbf{x}_k = \{x\}$

$$\mathbf{g}_k(\mathbf{o}|\{x\}) = \begin{cases} 1 - P^{\mathrm{D}}(x) & \text{if } \mathbf{o} = \emptyset \\ P^{\mathrm{D}}(x) g_k(o|x) & \text{if } \mathbf{o} = \{o\} \\ 0 & \text{if } |\mathbf{o}| > 1. \end{cases}$$

- **Note:** $\mathbf{o}_k|\mathbf{x}_k = \{x\}$ is a Bernoulli RFS with $r = P^{\mathrm{D}}(x)$ and pdf $g_k(\cdot|x)$.

Example, samples of $\mathbf{o}_k$

- Suppose $\mathbf{x}_k = \{x\}$, $P^{\mathrm{D}}(x) = 0.85$ and $g_k(o|x) = \mathcal{N}(o; [3\ ,2]^T, 0.3\mathbf{I})$.



MULTI-OBJECT MEASUREMENT MODEL (1)

Basic result

- The set of object measurements from a single object is a Bernoulli RFS.
- The set of object **measurements from multiple objects** is therefore a **multi-Bernoulli RFS**.

- Suppose $\mathbf{x}_k = \{x_k^1, \ldots, x_k^{n_k}\}$ and let $\mathbf{o}_k(x_k^i)$ be an RFS representing the set of object measurements from x_k^i.
- Given $\mathbf{x}_k = \{x_k^1, \ldots, x_k^{n_k}\}$ we have

$$\mathbf{o}_k = \mathbf{o}_k(x_k^1) \cup \mathbf{o}_k(x_k^2) \cup \cdots \cup \mathbf{o}_k(x_k^{n_k}).$$

MULTI-OBJECT MEASUREMENT MODEL (2)

- Given $\mathbf{x}_k = \{x_k^1, \ldots, x_k^{n_k}\}$, $\mathbf{o}_k(x_k^1), \ldots, \mathbf{o}_k(x_k^{n_k})$ are independent Bernoulli RFSs,
$$\mathbf{o}_k(x_k^i) \big| x_k^i \sim \mathbf{g}_k(\,\cdot\,\big|\{x_k^i\}).$$
- To understand the general expression, we introduce the shorthand notation $\mathbf{o}^i = \mathbf{o}_k(x_k^i)$:
$$\mathbf{o}_k = \mathbf{o}^1 \cup \mathbf{o}^2 \cup \cdots \cup \mathbf{o}^{n_k}.$$

General multi-object measurement model, $\mathbf{x}_k = \{x^1, x^2, \ldots, x^{n_k}\}$

- The convolution formula yields
$$\mathbf{g}_k(\mathbf{o}_k \big| \{x^1, \ldots, x^{n_k}\}) = \sum_{\mathbf{o}^1 \uplus \cdots \uplus \mathbf{o}^{n_k} = \mathbf{o}_k} \prod_{i=1}^{n_k} \mathbf{g}_k(\mathbf{o}^i \big| \{x^i\}).$$

In short, $\mathbf{o}_k \big| \mathbf{x}_k$ is a **multi-Bernoulli RFS**.

OBJECT MEASUREMENT SAMPLES

Samples of $\mathbf{o}_k$ when $\mathbf{x}_k = \{x^1, x^2\}$

- Suppose $P^{\mathrm{D}} = 0.85$ and that

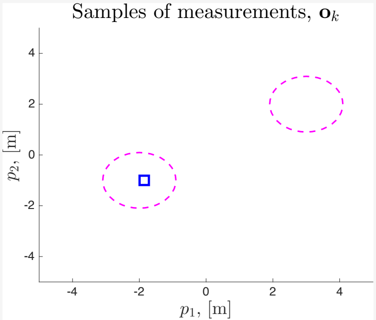
$$g_k(o|x) = \mathcal{N}(o; x, 0.3\mathbf{I}).$$

- When $\mathbf{x}_k = \{x^1, x^2\}$, where

$$x^1 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}, \quad x^2 = \begin{bmatrix} -2 \\ -1 \end{bmatrix},$$

we get

$$\mathbf{g}_k(\mathbf{o}_k|\mathbf{x}_k) = \sum_{\mathbf{o}^1 \uplus \mathbf{o}^2 = \mathbf{o}_k} \mathbf{g}_k\left(\mathbf{o}^1 \middle| \{x^1\}\right) \mathbf{g}_k\left(\mathbf{o}^2 \middle| \{x^2\}\right).$$



Measurement models – complete model

Multi-Object Tracking

Lennart Svensson

MEASUREMENT DISTRIBUTION (1)

- Given $\mathbf{x}_k$, we have
$$\mathbf{z}_k = \mathbf{c}_k \cup \mathbf{o}_k,$$
where $\mathbf{o}_k$ and $\mathbf{c}_k$ are independent:
$$p(\mathbf{z}_k | \mathbf{x}_k) = \sum_{\mathbf{c} \uplus \mathbf{o} = \mathbf{z}_k} p_{\mathbf{c}_k}(\mathbf{c}) \mathbf{g}_k(\mathbf{o} | \mathbf{x}_k).$$

Clutter model

- We assume **clutter is a Poisson RFS**
$$p_{\mathbf{c}_k}(\mathbf{c}) = \exp\left(-\int \lambda_c(c')\,\mathrm{d}c'\right) \prod_{c \in \mathbf{c}} \lambda_c(c),$$
where $\lambda_c(c)$ is its intensity function.

- We say that $\mathbf{z}_k | \mathbf{x}_k$ is a **Poisson multi-Bernoulli RFS**, since it is the union of a Poisson RFS $\mathbf{c}_k$ and a multi-Bernoulli RFS $\mathbf{o}_k | \mathbf{x}_k$.

MEASUREMENT DISTRIBUTION (2)

- Given $\mathbf{x}_k = \{x_k^1, \ldots, x_k^{n_k}\}$, we have $\mathbf{z}_k = \mathbf{c}_k \cup \mathbf{o}_k(x_k^1) \cup \cdots \cup \mathbf{o}_k(x_k^{n_k})$.

Measurement multiobject pdf

- For $\mathbf{x}_k = \{x_k^1, \ldots, x_k^{n_k}\}$, the measurement model is

$$p(\mathbf{z}_k | \mathbf{x}_k) = \sum_{\mathbf{c} \uplus \mathbf{o}^1 \uplus \cdots \uplus \mathbf{o}^{n_k} = \mathbf{z}_k} p_{\mathbf{c}_k}(\mathbf{c}) \prod_{i=1}^{n_k} \mathbf{g}_k(\mathbf{o}^i | \{x_k^i\})$$

where

$$p_{\mathbf{c}_k}(\mathbf{c}) = \exp\left(-\bar{\lambda}_c\right) \prod_{c \in \mathbf{c}} \lambda_c(c)$$

$$\mathbf{g}_k(\mathbf{o} | \{x\}) = \begin{cases} P^{\mathrm{D}}(x) g_k(o|x) & \text{if } \mathbf{o} = \{o\} \\ 1 - P^{\mathrm{D}}(x) & \text{if } \mathbf{o} = \emptyset \\ 0 & \text{if } |\mathbf{o}| > 1. \end{cases}$$

ASSOCIATION HYPOTHESES (1)

- In the formula

$$p(\mathbf{z}_k \big| \{x_k^1, \ldots, x_k^{n_k}\}) = \sum_{\mathbf{c} \uplus \mathbf{o}^1 \uplus \cdots \uplus \mathbf{o}^{n_k} = \mathbf{z}_k} p_{\mathbf{c}_k}(\mathbf{c}) \prod_{i=1}^{n_k} \mathbf{g}_k(\mathbf{o}^i \big| \{x_k^i\}),$$

  we sum over all possible **association hypotheses**.

- In earlier lectures we used $\theta_k = [\theta_k^1,\ \theta_k^2,\ \ldots, \theta_k^{n_k}]$, where

$$\theta_k^i = \begin{cases} j & \text{if object } i \text{ is associated to measurement } j \\ 0 & \text{if object } i \text{ is undetected,} \end{cases}$$

  and we summed over all hypotheses θ_k.

- For $\mathbf{z}_k = \{z_k^1, \ldots, z_k^{m_k}\}$, summing over $\mathbf{c} \uplus \mathbf{o}^1 \uplus \cdots \uplus \mathbf{o}^{n_k} = \mathbf{z}_k$ or θ_k is analogous:

$$\mathbf{o}^i = \begin{cases} \emptyset & \text{if } \theta_k^i = 0 \\ \{z_k^{\theta_k^i}\} & \text{if } \theta_k^i > 0, \end{cases} \qquad \mathbf{c} = \mathbf{z}_k \setminus \cup_{i=1}^{n_k} \mathbf{o}^i.$$

ASSOCIATION HYPOTHESES (2)

Example: Poisson Bernoulli measurement RFSs

- If $\mathbf{x}_k = \{x^1\}$ and $\mathbf{z}_k = \{z^1\}$ we get

$$p(\mathbf{z}_k|\mathbf{x}_k) = \sum_{\mathbf{c} \uplus \mathbf{o}^1 = \mathbf{z}_k} p_{\mathbf{c}_k}(\mathbf{c})\mathbf{g}_k(\mathbf{o}^1|\{x^1\})$$

$$= p_{\mathbf{c}_k}(\{z^1\})\mathbf{g}_k(\emptyset|\{x^1\}) + p_{\mathbf{c}}(\emptyset)\mathbf{g}_k(\{z^1\}|\{x^1\})$$

$$= \exp(-\bar{\lambda}_c)\lambda_c(z^1)(1 - P^{\mathrm{D}}(x^1)) + \exp(-\bar{\lambda}_c)P^{\mathrm{D}}(x^1)g_k(z^1|x^1).$$

- Using $\theta_k = [\theta_k^1]$, we get

$$p(\mathbf{z}_k|\mathbf{x}_k) = \sum_{\theta_k^1=0}^{1} \exp(-\bar{\lambda}_c)\lambda_c(z^1) \prod_{i:\theta_k^i=0} (1 - P^{\mathrm{D}}(x^i)) \prod_{i:\theta_k^i>0} \frac{P^{\mathrm{D}}(x^i)g_k(z^{\theta_k^i}|x^i)}{\lambda_c(z^{\theta_k^i})}.$$

A GENERAL MEASUREMENT MODEL (1)

A general measurement model (in terms of RFSs)

- For $\mathbf{z}_k = \{z_k^1, \dots, z_k^{m_k}\}$ and $\mathbf{x}_k = \{x_k^1, \dots, x_k^{n_k}\}$:

$$p(\mathbf{z}_k | \mathbf{x}_k) = \sum_{\theta_k} \exp(-\bar{\lambda}_c) \prod_{j=1}^{m_k} \lambda_c(z_k^j) \prod_{i:\theta^i=0} (1 - P^{\mathrm{D}}(x_k^i)) \prod_{i:\theta^i>0} \frac{P^{\mathrm{D}}(x_k^i) g_k(z_k^{\theta_k^i} | x_k^i)}{\lambda_c(z_k^{\theta_k^i})}.$$

Measurement models: multiobject pdf vs matrix distribution

- If $\mathbf{z}_k = \{z_k^1, \dots, z_k^{m_k}\}$, $Z_k = [z_k^1, \dots, z_k^{m_k}]$, $\mathbf{x}_k = \{x_k^1, \dots, x_k^{n_k}\}$ and $X_k = [x_k^1, \dots, x_k^{n_k}]$:

$$p(\mathbf{z}_k | \mathbf{x}_k) = m_k! \, p(Z_k | X_k).$$

A GENERAL MEASUREMENT MODEL (2)

A general measurement model – alternative form

- For $\mathbf{z}_k = \{z_k^1, \ldots, z_k^{m_k}\}$ and $\mathbf{x}_k = \{x_k^1, \ldots, x_k^{n_k}\}$:

$$p(\mathbf{z}_k|\mathbf{x}_k) = p_{\mathbf{c}_k}(\mathbf{z}_k)\mathbf{g}_k(\emptyset|\mathbf{x}_k) \sum_{\theta_k} \prod_{i:\theta_k^i>0} \frac{P^{\mathrm{D}}(x_k^i) g_k(z_k^{\theta_k^i}|x_k^i)}{\lambda_c(z_k^{\theta_k^i})(1 - P^{\mathrm{D}}(x_k^i))},$$

where

$$p_{\mathbf{c}_k}(\mathbf{z}_k) = \exp(-\bar{\lambda}_c) \prod_{s=1}^{m_k} \lambda_c(z_k^s)$$

$$\mathbf{g}_k(\emptyset|\mathbf{x}_k) = \prod_{j=1}^{n_k} (1 - P^{\mathrm{D}}(x_k^j)).$$

CONCLUSIONS

- The measurement model has not changed.
- We found that $\mathbf{z}_k \big| \mathbf{x}_k$ is a Poisson multi-Bernoulli (PMB) RFS.
- Simple to derive the measurement model using the convolution formula (no need to condition on m_k).
- Also: same derivation can be used for extended objects.
- It holds that $p(\mathbf{z}_k \big| \mathbf{x}_k) = m_k!\, p(Z_k \big| X_k)$: derivations give the same result.

Motion models – surviving objects

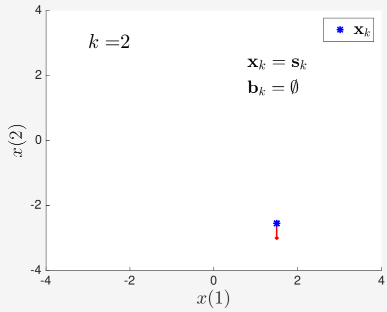
Multi-Object Tracking

Lennart Svensson

STANDARD MOTION MODEL

- Objects appear/disappear with time.
- Given $\mathbf{x}_{k-1}$, we assume
$$\mathbf{x}_k = \mathbf{s}_k \cup \mathbf{b}_k,$$
where $\mathbf{s}_k$ and $\mathbf{b}_k$ are independent,
  - $\mathbf{s}_k$: objects present also at time $k-1$,
  - $\mathbf{b}_k$: objects that have appeared since time $k-1$.
- In this video, we present the **standard model** for
$$\pi_k(\mathbf{s}_k | \mathbf{x}_{k-1}).$$

Example: a sequence of $\mathbf{x}_k$



- **Note:** some similarities to measurement model ($\mathbf{s}_k \leftrightarrow \mathbf{o}_k$, $\mathbf{b}_k \leftrightarrow \mathbf{c}_k$).

MOTION MODEL: STANDARD ASSUMPTIONS (SURVIVING OBJECTS)

Single object motion model (for already present objects)

- An object with state x survives/persists with probability $P^S(x)$.
- If it survives, it moves according to a single object motion model $\pi_k(s|x)$.

In the presence of other objects:

- Conditioned on its state, each object moves independently of all other objects.

SINGLE OBJECT MOTION MODEL

Case 1: $\mathbf{x}_{k-1} = \emptyset$

$$\pi_k(\mathbf{s}|\emptyset) = \begin{cases} 1 & \text{if } \mathbf{s} = \emptyset \\ 0 & \text{otherwise} \end{cases}$$

- **Note:** $\mathbf{s}_k | \mathbf{x}_{k-1} = \emptyset$ is a Ber. RFS with $r = 0$.

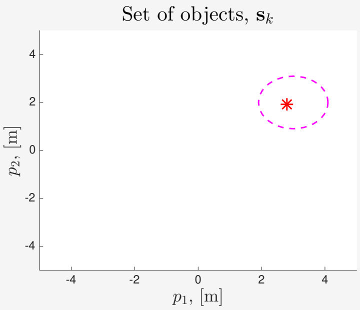
Case 2: $\mathbf{x}_{k-1} = \{x\}$

$$\pi_k(\mathbf{s}|\{x\}) = \begin{cases} 1 - P^{\mathrm{S}}(x) & \text{if } \mathbf{s} = \emptyset \\ P^{\mathrm{S}}(x)\pi_k(s|x) & \text{if } \mathbf{s} = \{s\} \\ 0 & \text{if } |\mathbf{s}| > 1. \end{cases}$$

- **Note:** $\mathbf{s}_k | \mathbf{x}_{k-1} = \{x\}$ is a Bernoulli RFS with $r = P^{\mathrm{S}}(x)$ and pdf $\pi_k(\cdot|x)$.

Example, samples of $\mathbf{s}_k$

- Suppose $\mathbf{x}_{k-1} = \{x\}$, $P^{\mathrm{S}}(x) = 0.85$ and $\pi_k(s|x) = \mathcal{N}(s; [3\ ,2]^T, 0.3\mathbf{I})$.



MULTI-OBJECT SURVIVING MODEL (1)

Basic result

- The set of surviving objects from a single object is a Bernoulli RFS.
- The set of surviving objects **from multiple objects** is therefore a **multi-Bernoulli RFS**.

- Suppose $\mathbf{x}_{k-1} = \{x_{k-1}^1, \ldots, x_{k-1}^{n_{k-1}}\}$ and let $\mathbf{s}_k(x_{k-1}^i)$ be an RFS representing the set of surviving objects from x_{k-1}^i.
- Given $\mathbf{x}_{k-1} = \{x_{k-1}^1, \ldots, x_{k-1}^{n_{k-1}}\}$ we have

$$\mathbf{s}_k = \mathbf{s}_k(x_{k-1}^1) \cup \mathbf{s}_k(x_{k-1}^2) \cup \cdots \cup \mathbf{s}_k(x_{k-1}^{n_{k-1}}).$$

MULTI-OBJECT SURVIVING MODEL (2)

- Given $\mathbf{x}_{k-1} = \{x_{k-1}^1, \ldots, x_{k-1}^{n_{k-1}}\}$, $\mathbf{s}_k(x_{k-1}^1), \ldots, \mathbf{s}_k(x_{k-1}^{n_{k-1}})$ are independent Bernoulli RFSs,
$$\mathbf{s}_k(x_{k-1}^i)\big|x_{k-1}^i \sim \pi_k(\,\cdot\,\big|\{x_{k-1}^i\}).$$
- To understand the general expression, we introduce the shorthand notation $\mathbf{s}^i = \mathbf{s}_{k-1}(x_{k-1}^i)$:
$$\mathbf{s}_k = \mathbf{s}^1 \cup \mathbf{s}^2 \cup \cdots \cup \mathbf{s}^{n_{k-1}}.$$

General multi-object surviving model, $\mathbf{x}_{k-1} = \{x^1, x^2, \ldots, x^{n_{k-1}}\}$

- The convolution formula yields:
$$\pi_k(\mathbf{s}_k\big|\{x^1, \ldots, x^{n_{k-1}}\}) = \sum_{\mathbf{s}^1 \uplus \cdots \uplus \mathbf{s}^{n_{k-1}} = \mathbf{s}_k} \prod_{i=1}^{n_{k-1}} \pi_k(\mathbf{s}^i\big|\{x^i\}).$$

In short, $\mathbf{s}_k\big|\mathbf{x}_{k-1}$ is a **multi-Bernoulli RFS**.

SAMPLES OF SURVIVING OBJECTS

Samples of $\mathbf{s}_k$ when $\mathbf{x}_{k-1} = \{x^1, x^2\}$

- Suppose $P^S = 0.9$ and that

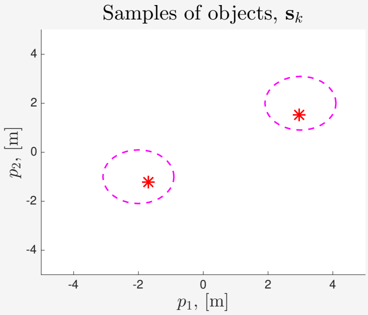
$$\pi_k(s|x) = \mathcal{N}(s; x, 0.3\mathbf{I}).$$

- When $\mathbf{x}_{k-1} = \{x^1, x^2\}$, where

$$x^1 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}, \quad x^2 = \begin{bmatrix} -2 \\ -1 \end{bmatrix},$$

we get

$$\pi_k(\mathbf{s}_k|\mathbf{x}_{k-1}) = \sum_{\mathbf{s}^1 \uplus \mathbf{s}^2 = \mathbf{s}_k} \pi_k\left(\mathbf{s}^1 \middle| \{x^1\}\right) \pi_k\left(\mathbf{s}^2 \middle| \{x^2\}\right).$$



Complete motion model

Multi-Object Tracking

Lennart Svensson

MOTION MODEL (1)

- Given $\mathbf{x}_{k-1}$, we have
$$\mathbf{x}_k = \mathbf{s}_k \cup \mathbf{b}_k,$$
where $\mathbf{s}_k$ and $\mathbf{b}_k$ are independent:
$$p(\mathbf{x}_k | \mathbf{x}_{k-1}) = \sum_{\mathbf{b} \uplus \mathbf{s} = \mathbf{x}_k} p_{\mathbf{b}_k}(\mathbf{b}) \pi_k(\mathbf{s} | \mathbf{x}_{k-1}).$$

Birth model

- We assume the **birth process is a Poisson RFS**
$$p_{\mathbf{b}_k}(\mathbf{b}) = \exp\left(-\int \lambda_b(b')\,\mathrm{d}b'\right) \prod_{b \in \mathbf{b}} \lambda_b(b),$$
where $\lambda_b(b)$ is its intensity function.

- We say that $\mathbf{x}_k | \mathbf{x}_{k-1}$ is a **Poisson multi-Bernoulli RFS**, since it is the union of a Poisson RFS $\mathbf{b}_k$ and a multi-Bernoulli RFS $\mathbf{s}_k | \mathbf{x}_{k-1}$.

MOTION MODEL (2)

- Given $\mathbf{x}_{k-1} = \{x_{k-1}^1, \ldots, x_{k-1}^{n_{k-1}}\}$, we have $\mathbf{x}_k = \mathbf{b}_k \cup \mathbf{s}_k(x_{k-1}^1) \cup \cdots \cup \mathbf{s}_k(x_{k-1}^{n_{k-1}})$.

Motion model

- The motion model is

$$\pi_k(\mathbf{x}_k | \{x_{k-1}^1, \ldots, x_{k-1}^{n_{k-1}}\}) = \sum_{\mathbf{b} \uplus \mathbf{s}^1 \uplus \cdots \uplus \mathbf{s}^{n_{k-1}} = \mathbf{x}_k} p_{\mathbf{b}_k}(\mathbf{b}) \prod_{i=1}^{n_{k-1}} \pi_k(\mathbf{s}^i | \{x_{k-1}^i\})$$

where

$$p_{\mathbf{b}_k}(\mathbf{b}) = \exp\left(-\bar{\lambda}_b\right) \prod_{b \in \mathbf{b}} \lambda_b(b)$$

$$\pi_k(\mathbf{s} | \{x\}) = \begin{cases} P^{\mathrm{S}}(x)\pi_k(s|x) & \text{if } \mathbf{s} = \{s\} \\ 1 - P^{\mathrm{S}}(x) & \text{if } \mathbf{s} = \emptyset \\ 0 & \text{if } |\mathbf{s}| > 1. \end{cases}$$

CONCLUDING REMARKS

- Objects can appear and disappear with time.
- We assume that
  - $\mathbf{s}_k \big| \mathbf{x}_{k-1} = \{x\}$ is a Bernoulli RFS,
  - $\mathbf{b}_k$ is a Poisson point process,
  - given $\mathbf{x}_{k-1}$, $\mathbf{x}_k = \mathbf{s}_k \cup \mathbf{b}_k$ is a Poisson multi-Bernoulli RFS.
- We can use the convolution formula to express $p(\mathbf{x}_k \big| \mathbf{x}_{k-1})$.

Section 5:
Probability hypothesis density filtering

Multi-Object Tracking

Lennart Svensson

PHD filtering – introduction

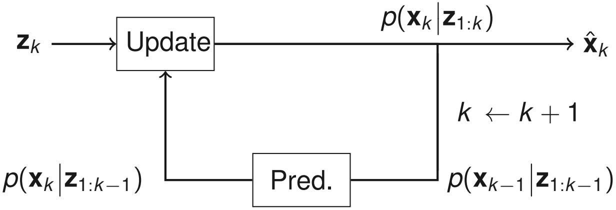
Multi-Object Tracking

Lennart Svensson

PHD FILTERING: BASIC IDEA

Assumed density filtering

- To obtain a recursive algorithm both $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ and $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ should belong to the same family of distributions.



PHD filtering

- Both $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ and $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ are approximated as Poisson multi-object pdfs.

APPROXIMATING MULTI-OBJECT PDFS AS POISSON

- Suppose $p(\mathbf{x}_{k-1} \middle| \mathbf{z}_{1:k-1})$ is a Poisson multi-object pdf.
- How can we approximate $p(\mathbf{x}_k \middle| \mathbf{z}_{1:k-1})$ and $p(\mathbf{x}_k \middle| \mathbf{z}_{1:k})$ as Poisson multi-object pdfs?

Poisson RFS approximations

- To approximate a RFS $\mathbf{x} \sim p(\cdot)$ as a Poisson RFS, we set the Poisson intensity to

$$\lambda(x) = D(x),$$

where $D(x)$ is the **probability hypothesis density (PHD)** of $\mathbf{x} \sim p(\mathbf{x})$.

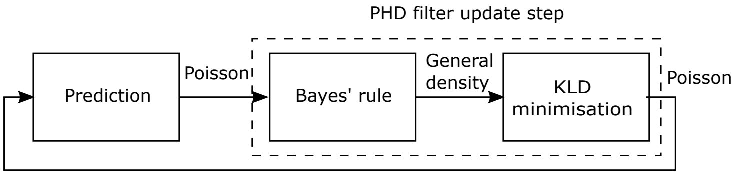
- The above is optimal in the Kullback-Leibler sense.

OVERVIEW OF PHD FILTERING

PHD filtering: basic principles

- Recursively compute the PHDs $D_{k|k-1}(x)$ of $p(\mathbf{x}_k | \mathbf{z}_{1:k-1})$ and $D_{k|k}(x)$ of $p(\mathbf{x}_k | \mathbf{z}_{1:k})$.
- Approximate $p(\mathbf{x}_k | \mathbf{z}_{1:k-1})$ and $p(\mathbf{x}_k | \mathbf{z}_{1:k})$ as Poisson multi-object pdfs with intensity functions $D_{k|k-1}(x)$ and $D_{k|k}(x)$, respectively.

- **Note:** It turns out that $p(\mathbf{x}_k | \mathbf{z}_{1:k-1})$ is a Poisson multi-object pdf $\Rightarrow$ no approximations needed.



CONCLUDING REMARKS

- The PHDs $D_{k|k-1}(x)$ and $D_{k|k}(x)$ are functions in **single object state**.
- The PHDs parametrise the multiobject pdfs, e.g.,

$$p(\mathbf{x}_k \middle| \mathbf{z}_{1:k}) = \exp\left(-\int D_{k|k}(x')\mathrm{d}x'\right) \prod_{x \in \mathbf{x}_k} D_{k|k}(x).$$

  That is, we approximate $p(\mathbf{x}_k | \mathbf{z}_{1:k})$ as a Poisson point process (PPP) with intensity function $D_{k|k}(x)$.

- Elements in a PPP are independent and identically distributed (given its cardinality) $\Rightarrow$ often a rough approximation of the posterior.
- The PHD filter is a simple and efficient algorithm that performs well in simple scenarios.

The PHD and its properties

Multi-Object Tracking

Lennart Svensson

PHD: DEFINITION

PHD definition

- The **probability hypothesis density** (PHD) function, $D_{\mathbf{x}}(x)$, of a RFS $\mathbf{x}$ is

$$\begin{aligned} D_{\mathbf{x}}(x) &= \int p_{\mathbf{x}}(\mathbf{x}) \sum_{x' \in \mathbf{x}} \delta(x - x') \delta\mathbf{x} \\ &= \int p_{\mathbf{x}}(\{x\} \cup \mathbf{x}) \, \delta\mathbf{x}. \end{aligned}$$

- The PHD is a **first-order statistical moment** of the RFS.
- We sometimes refer to $D_{\mathbf{x}}(x)$ as the **intensity function** of $\mathbf{x}$.

INTEGRATING THE PHD

Expected cardinality in region

- If $A \subseteq \mathbb{R}^{n_x}$, then
$$\int_A D_{\mathbf{x}}(x)\,\mathrm{d}x = \mathbb{E}\left[|\mathbf{x} \cap A|\right].$$

- That is, $D(x)\mathrm{d}x$ is the expected number of objects in $\mathrm{d}x$ and $\int D(x)\,\mathrm{d}x = \mathbb{E}\left[|\mathbf{x}|\right]$.
- **Proof:** The integral of a PHD is

$$\begin{aligned}
\int_A D_{\mathbf{x}}(x)\,\mathrm{d}x &= \int_A \int p_{\mathbf{x}}(\mathbf{x}) \sum_{x' \in \mathbf{x}} \delta(x - x')\delta\mathbf{x}\,\mathrm{d}x \\
&= \int p_{\mathbf{x}}(\mathbf{x}) \underbrace{\sum_{x' \in \mathbf{x}} \underbrace{\int_A \delta(x - x')\,\mathrm{d}x}_{=1 \text{ if } x' \in A}}_{|\mathbf{x} \cap A|} \;\delta\mathbf{x} \\
&= \mathbb{E}\left[|\mathbf{x} \cap A|\right].
\end{aligned}$$

THE PHD OF A BERNOULLI RFS

The PHD of a Bernoulli RFS

- Consider a Bernoulli RFS $\mathbf{x}$

$$p_{\mathbf{x}}(x) = \begin{cases} 1 - r & \text{if } \mathbf{x} = \emptyset \\ r\, p_x(x) & \text{if } \mathbf{x} = \{x\} \\ 0 & \text{if } |\mathbf{x}| \geq 2. \end{cases}$$

- The PHD of $\mathbf{x}$ is

$$\begin{aligned} D_{\mathbf{x}}(x) &= \int p_{\mathbf{x}}(\{x\} \cup \mathbf{x})\, \delta\mathbf{x} \\ &= p_{\mathbf{x}}(\{x\} \cup \emptyset) + 0 \\ &= r\, p_x(x) \end{aligned}$$

- That is, $D_{\mathbf{x}}(x) = r\, p_x(x)$.

THE PHD OF A POISSON RFS

- Suppose $\mathbf{x}$ is a Poisson RFS with intensity $\lambda(x)$.
- What is the PHD of $\mathbf{x}$?

PHD of Poisson RFS

- The PHD of a Poisson RFS with intensity $\lambda(x)$ is

$$D_{\mathbf{x}}(x) = \lambda(x).$$

- Useful **sanity check**! To "approximate" $\mathbf{x}$ as a Poisson RFS with intensity $D_{\mathbf{x}}(x)$, the best choice is $D_{\mathbf{x}}(x) = \lambda(x)$.

PHDs AND UNION OF RFSs

Union of RFSs

- If $\mathbf{x}$ is the union of the independent RFSs $\mathbf{x}_1, \dots, \mathbf{x}_N$, then

$$D_{\mathbf{x}}(x) = D_{\mathbf{x}_1}(x) + \cdots + D_{\mathbf{x}_N}(x).$$

- For $A \in \mathbb{R}^{n_x}$, it follows that $\mathbb{E}[|\mathbf{x} \cap A|] = \sum_{i=1}^{N} \mathbb{E}[|\mathbf{x}_i \cap A|]$.

PHD of multi-Bernoulli RFS

- If $\mathbf{x}$ is a multi-Bernoulli RFS whose N Bernoulli components are parametrised by $(r_1, p_1(x)), \dots, (r_N, p_N(x))$:

$$D_{\mathbf{x}}(x) = \sum_{i=1}^{N} r_i \, p_i(x).$$

PHD filter prediction

Multi-Object Tracking

Lennart Svensson

GAUSSIAN MIXTURE PHD FILTERING

Gaussian mixture (GM) parametrisation

- We assume the PHDs are parametrised as

$$D_{k-1|k-1}(x) = \sum_{h=1}^{\mathcal{H}_{k-1|k-1}} w^h_{k-1|k-1}\mathcal{N}(x; \mu^h_{k-1|k-1}, P^h_{k-1|k-1})$$

$$D_{k|k-1}(x) = \sum_{h=1}^{\mathcal{H}_{k|k-1}} w^h_{k|k-1}\mathcal{N}(x; \mu^h_{k|k-1}, P^h_{k|k-1}).$$

- **Note 1:** the weights do not have to sum to 1, e.g.,

$$\mathbb{E}\left[|\mathbf{x}_k|\big|\mathbf{z}_{1:k-1}\right] = \int D_{k|k-1}(x)\,\mathrm{d}x = \sum_{h=1}^{\mathcal{H}_{k|k-1}} w^h_{k|k-1}.$$

- **Note 2:** the GM form may introduce additional approximations (apart from the PPP approximation).
- **Prediction step:** find parameters in $D_{k|k-1}(x)$ given $D_{k-1|k-1}(x)$.

MOTION MODEL

Standard motion models with linear and Gaussian π_k

- We assume the standard motion model, with

$$\lambda_{b,k}(x) = \sum_{h=1}^{\mathcal{H}_k^b} w_{b,k}^h \mathcal{N}(x; \mu_{b,k}^h, P_{b,k}^h)$$

$$\pi_k(\mathbf{x}_k \big| \{x_{k-1}\}) = \begin{cases} P^S \mathcal{N}(x_k; F_k x_{k-1}, Q_{k-1}) & \text{if } \mathbf{x}_k = \{x_k\} \\ 1 - P^S & \text{if } \mathbf{x}_k = \emptyset. \end{cases}$$

- **Remarks:**
  - The $\lambda_{b,k}(x)$ captures where we expect objects to appear.
  - Probability of survival is constant.
  - Surviving objects move according to a linear and Gaussian model.

PPP PREDICTION

PPP prediction

- Suppose $\mathbf{x}_{k-1} \big| \mathbf{z}_{1:k-1}$ is a PPP with PHD (intensity function)

$$D_{k-1|k-1}(x) = \sum_{h=1}^{\mathcal{H}_{k-1|k-1}} w^h_{k-1|k-1} \mathcal{N}(x; \mu^h_{k-1|k-1}, P^h_{k-1|k-1}).$$

- It follows that $\mathbf{x}_k \big| \mathbf{z}_{1:k-1}$ is a PPP with PHD

$$D_{k|k-1}(x) = D^{\mathrm{S}}_{k|k-1}(x) + \lambda_{b,k}(x),$$

where $D^{\mathrm{S}}_{k|k-1}(x)$ is a Gaussian mixture with parameters

$$\begin{aligned} \mathcal{H}^s_{k|k-1} &= \mathcal{H}_{k-1|k-1} & \qquad w^{s,h}_{k|k-1} &= P^{\mathrm{S}}\, w^h_{k-1|k-1} \\ \mu^{s,h}_{k|k-1} &= F_{k-1} \mu^h_{k-1|k-1} & \qquad P^{s,h}_{k|k-1} &= F_{k-1} P^h_{k-1|k-1} F^T_{k-1} + Q_{k-1}. \end{aligned}$$

- **Note:** $\mathbf{x}_k \big| \mathbf{z}_{1:k-1}$ is a PPP with GM-PHD $\Rightarrow$ no new approximations needed!

GM-PHD PREDICTION

Algorithm GM-PHD prediction.

Set $\mathcal{H}_{k|k-1} = \mathcal{H}_k^b + \mathcal{H}_{k-1|k-1}$.

for $h = 1$ **to** $\mathcal{H}_k^b$ **do**

  Set $w_{k|k-1}^h = w_{b,k}^h$, $\mu_{k|k-1}^h = \mu_{b,k}^h$ and $P_{k|k-1}^h = P_{b,k}^h$.

end for

for $h = 1$ **to** $\mathcal{H}_{k-1|k-1}$ **do**

  Set

$$w_{k|k-1}^{h+\mathcal{H}_k^b} = P^{\mathrm{S}}\, w_{b,k}^h, \qquad \mu_{k|k-1}^{h+\mathcal{H}_k^b} = F_{k-1}\mu_{k-1|k-1}^h,$$

$$P_{k|k-1}^{h+\mathcal{H}_k^b} = F_{k-1} P_{k-1|k-1}^h F_{k-1}^T + Q_{k-1}.$$

end for

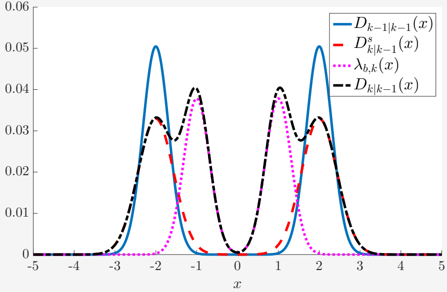
GM-PHD PREDICTION: VISUALIZATION

A GM-PHD prediction example

- Suppose $\mathcal{H}_{k-1|k-1} = 2$, and that

$$w^1_{k-1|k-1} = w^2_{k-1|k-1} = 0.04$$
$$P^1_{k-1|k-1} = P^2_{k-1|k-1} = 0.1$$
$$\mu^1_{k-1|k-1} = -2, \qquad \mu^2_{k-1|k-1} = 2.$$

- Also, suppose $P^S = 0.9$, $F_{k-1} = 1$, $Q_{k-1} = 0.3^2$ and let $\lambda_{b,k}(x)$ be a GM with two components.
- The predicted PHD, $D_{k|k-1}(x)$ is then a GM with 4 components.



PHD filter update – part 1

Multi-Object Tracking

Lennart Svensson

GM-PHD UPDATE

GM parametrisation

- We assume that $\mathbf{x}_k \big| \mathbf{z}_{1:k-1}$ is a PPP with PHD (intensity function)

$$D_{k|k-1}(x) = \sum_{k=1}^{\mathcal{H}_{k|k-1}} w^h_{k|k-1} \mathcal{N}(x; \mu^h_{k|k-1}, P^h_{k|k-1}).$$

GM-PHD filter update (conceptual description)

- 1) Find $p(\mathbf{x}_k \big| \mathbf{z}_{1:k})$.
- 2) Find the GM-PHD, $D_{k|k}(x)$, of $p(\mathbf{x}_k \big| \mathbf{z}_{1:k})$, and its parameters

$$\left\{ w^h_{k|k}, \mu^h_{k|k}, P^h_{k|k} \right\}_{h=1}^{\mathcal{H}_{k|k}}.$$

- 3) Approximate $\mathbf{x}_k \big| \mathbf{z}_{1:k}$ as a PPP with PHD $D_{k|k}(x)$.

MEASUREMENT MODEL

Measurement model

- We assume the standard measurement model, with

$$\mathbf{g}_k(\mathbf{z}_k \big| \{x_k\}) = \begin{cases} P^{\mathrm{D}} \mathcal{N}(z_k; H_k x_k, R_k) & \text{if } \mathbf{z}_k = \{z_k\} \\ 1 - P^{\mathrm{D}} & \text{if } \mathbf{z}_k = \emptyset \\ 0 & \text{if } |\mathbf{z}_k| > 1, \end{cases}$$

whereas we can handle general clutter intensities $\lambda_{c,k}(z)$.

- **Remarks:**
  - Probability of detection is constant and g_k is linear and Gaussian.
  - We observe $\mathbf{z}_k = \{z_k^1, \ldots, z_k^{m_k}\}$.

EXACT POSTERIOR, $p(\mathbf{x}_k | \mathbf{z}_{1:k})$

Posterior multi-Bernoulli posterior

- Given above assumptions, $\mathbf{x}_k | \mathbf{z}_{1:k}$ is a Poisson multi-Bernoulli (PMB) RFS, where the PPP has intensity

$$\lambda_{k|k}(x) = (1 - P^{\mathrm{D}})\, D_{k|k-1}(x),$$

  and the MB process has m_k components, for $i = 1, \ldots, m_k$:

$$r^i_{k|k} = \frac{P^{\mathrm{D}} \int \mathcal{N}(z^i_k; H_k x', R_k) D_{k|k-1}(x')\,\mathrm{d}x'}{\lambda_c(z^i_k) + P^{\mathrm{D}} \int \mathcal{N}(z^i_k; H_k \tilde{x}, R_k) D_{k|k-1}(\tilde{x})\,\mathrm{d}\tilde{x}}$$

$$p^i_{k|k}(x) \propto \mathcal{N}(z^i_k; H_k x, R_k) D_{k|k-1}(x).$$

- **Remarks:**
  - The posterior has the PHD

$$D_{k|k}(x) = \lambda_{k|k}(x) + \sum_{i=1}^{m_k} r^i_{k|k}\, p^i_{k|k}(x).$$

  - $D_{k|k-1}(x)$ is a GM with $\mathcal{H}_{k|k-1}$ components $\Rightarrow \mathcal{H}_{k|k} = \mathcal{H}_{k|k-1} \times (m_k + 1)$.

PHD filter update – part 2

Multi-Object Tracking

Lennart Svensson

GM-PHD UPDATE: PPP

- The exact posterior contains a PPP with PHD

$$(1 - P^{\mathrm{D}})D_{k|k-1}(x) = \sum_{k=1}^{\mathcal{H}_{k|k-1}} (1 - P^{\mathrm{D}})w_{k|k-1}^{h}\mathcal{N}(x; \mu_{k|k-1}^{h}, P_{k|k-1}^{h}).$$

- This PPP represents objects that are undetected at time k.

- We store these as the first $\mathcal{H}_{k|k-1}$ components in $D_{k|k}(x)$.

Algorithm GM-PHD update (1).

for $h = 1$ **to** $\mathcal{H}_{k|k-1}$ **do**
  $w_{k|k}^{h} = (1 - P^{\mathrm{D}})\, w_{k|k-1}^{h}$
  $\mu_{k|k}^{h} = \mu_{k|k-1}^{h}$
  $P_{k|k}^{h} = P_{k|k-1}^{h}$
end for

GM-PHD UPDATE: MB (1)

- The posterior also contains m_k Bernoulli components.
- These represent the set of detected objects at time k.
  $\rightsquigarrow$ One potential (detected) object for each measurement.
- We can write

$$r^i_{k|k} p^i_{k|k}(x) = \sum_{h=1}^{\mathcal{H}_{k|k-1}} w^{h,i}_k \mathcal{N}(x; \mu^{h,i}_k, P^{h,i}_k),$$

  where

$$\mathcal{N}(x; \mu^{h,i}_k, P^{h,i}_k) \propto \mathcal{N}(z^i_k; H_k x, R_k)\mathcal{N}(x; \mu^h_{k|k-1}, P^h_{k|k-1}).$$

- That is, $(\mu^{h,i}_k, P^{h,i}_k)$ are given by a Kalman filter update of $\mathcal{N}(x; \mu^h_{k|k-1}, P^h_{k|k-1})$ using $\mathcal{N}(z^i_k; H_k x, R_k)$.

GM-PHD UPDATE: MB (2)

- First compute parameters for the $\mathcal{H}_{k|k-1}$ Kalman filters.

Algorithm GM-PHD update (2).

for $h = 1$ **to** $\mathcal{H}_{k|k-1}$ **do**

$\quad \hat{z}^h_{k|k-1} = H_k \mu^h_{k|k-1}$

$\quad S^h_k = R_k + H_k P^h_{k|k-1} H^T_k$

$\quad K^h_k = P^h_{k|k-1} H^T_k \left(S^h_k\right)^{-1}$

$\quad P^h_k = \left(I - K^h_k H_k\right) P^h_{k|k-1}$

end for

- We now compute and store the GM-variables.

Algorithm GM-PHD update (3).

for $i = 1$ **to** m_k **do**

$\quad$ **for** $h = 1$ **to** $\mathcal{H}_{k|k-1}$ **do**

$\qquad \mu^{i\mathcal{H}_{k|k-1}+h}_{k|k} = \mu^h_{k|k-1} + K^h_k(z^i_k - \hat{z}^h_{k|k-1})$

$\qquad P^{i\mathcal{H}_{k|k-1}+h}_{k|k} = P^h_k$

$\qquad \tilde{w}^{i\mathcal{H}_{k|k-1}+h}_{k|k} = P^{\mathrm{D}} w^h_{k|k-1} \mathcal{N}(z^i_k; \hat{z}^h_{k|k-1}, S^h_k)$

$\quad$ **end for**

$\quad$ **for** $h = 1$ **to** $\mathcal{H}_{k|k-1}$ **do**

$\qquad w^{i\mathcal{H}_{k|k-1}+h}_{k|k} = \dfrac{\tilde{w}^{i\mathcal{H}_{k|k-1}+h}_{k|k}}{\lambda_c(z^i_k) + \sum_{h'=1}^{\mathcal{H}_{k|k-1}} \tilde{w}^{i\mathcal{H}_{k|k-1}+h'}_{k|k}}$

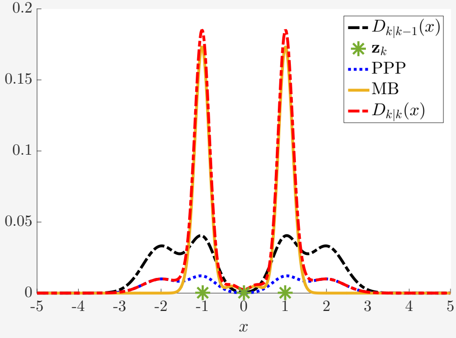
$\quad$ **end for**

end for

GM-PHD UPDATE: VISUALIZATION

A GM-PHD update example

- Suppose
$$\mathbf{z}_k = \{-1, 0, 1\}.$$
and $D_{k|k-1}(x)$ is a GM with four components.
- Also, suppose $P^{\mathrm{D}} = 0.7$, and that $H_k = 1$, $R_k = 0.2^2$ and
$$\lambda_c(c) = \begin{cases} 0.3 & \text{if } |c| \geq 5, \\ 0 & \text{otherwise.} \end{cases}$$
- Posterior PHD is dominated by two peaks due to measurements at ± 1.



CONCLUDING REMARKS

- The GM-PHD update step is very simple.

- We perform m_k different updates for each of the $\mathcal{H}_{k|k-1}$ predicted Gaussian densities.

- GM grows as $\mathcal{H}_{k|k} = (m_k + 1) \times \mathcal{H}_{k|k-1}$ regardless of

$$\hat{n}_{k|k-1} = \mathbb{E}_{p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}\left[|\mathbf{x}_k|\right] = \sum_{h=1}^{\mathcal{H}_{k|k-1}} w_{k|k-1}^{h}.$$

- The factor $m_k + 1$ corresponds $N_A(m_k, 1)$, which is generally much smaller than $N_A(m_k, \mathrm{round}(\hat{n}_{k|k-1}))$.

GM-PHD: mixture reduction and estimation

Multi-Object Tracking

Lennart Svensson

MIXTURE REDUCTION FOR THE GM-PHD

- Without approximations, the number of terms in the GM grows as

$$\mathcal{H}_{k|k-1} = \mathcal{H}_{k-1|k-1} + \mathcal{H}_k^b$$
$$\mathcal{H}_{k|k} = (m_k + 1) \times \mathcal{H}_{k|k-1}.$$

- Clearly, $\mathcal{H}_{k|k}$ grows quickly with time!
- How can we reduce the number of terms?
  - As usual: using **pruning** and **merging**.
  - **Note:** we do not normalize weights after pruning.

A common reduction strategy

- 1) Remove components with weights $< \gamma$.
- 2) Merge similar components.
- 3) Cap the number of components at $N_{\max}$.

ESTIMATING THE SET OF OBJECTS

An estimator for GM-PHD

- Estimate the number of objects:
$$\hat{n}_{k|k} = \text{round}\left(\sum_{h=1}^{\mathcal{H}_{k|k}} w_{k|k}^{h}\right).$$
- Include $\mu_{k|k}^{h}$ for the $\hat{n}_{k|k}$ largest weights in the set $\hat{\mathbf{x}}_k$.

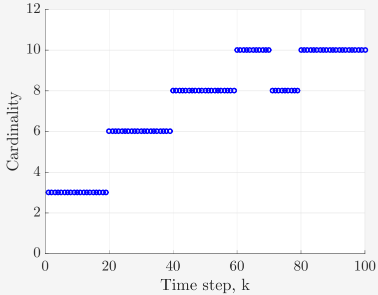
Algorithm Forming a set of estimates.

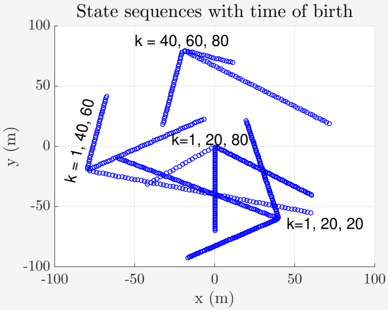
Input: $\hat{n}, w^h, \mu^h, h = 1, \dots, \mathcal{H}$.
Output: $\hat{\mathbf{x}}$
$[out, ind] = \text{sort}([w^1, \dots, w^{\mathcal{H}}], \text{'descend'})$.
Initialize $\hat{\mathbf{x}} = \emptyset$
for $i = 1$ **to** $\hat{n}$ **do**
    Set $\hat{\mathbf{x}} = \hat{\mathbf{x}} \cup \{\mu^{\text{ind}(i)}\}$.
end for

A SIMULATION EXAMPLE (1)

A GM-PHD simulation example

- State sequence is generated deterministically.





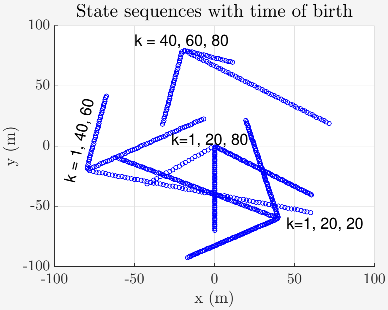
A SIMULATION EXAMPLE (1)

A GM-PHD simulation example

- State sequence is generated deterministically.
- The PHD filter assumes:
  - CV motion: $T = 1$, $Q_k = 4$.
  - Observations: $R_k = 4 \times I_{2\times 2}$,

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

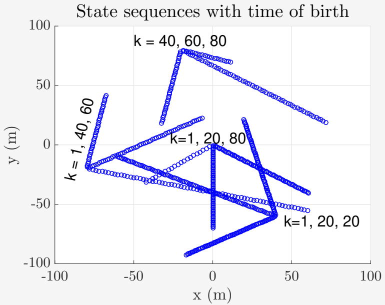
  - $P^{\mathrm{D}} = 0.98$, $P^{\mathrm{S}} = 0.99$, $\lambda_c(c) = 1.25 \times 10^{-4}$.
  - $\lambda_{b,k}$ is a GM with 4 components, means where objects appear.
- Measurements: generated from model.



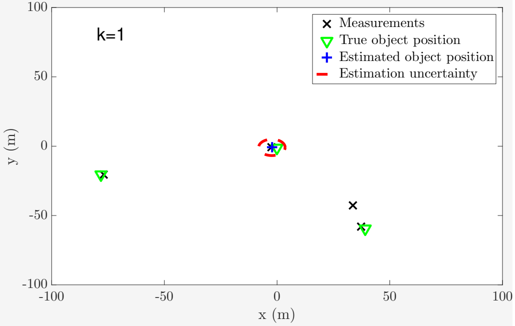
A SIMULATION EXAMPLE (2)

A GM-PHD simulation example

- Recall the true sequences.



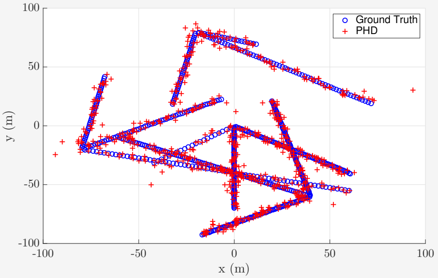
- The PHD filter yields the estimates:



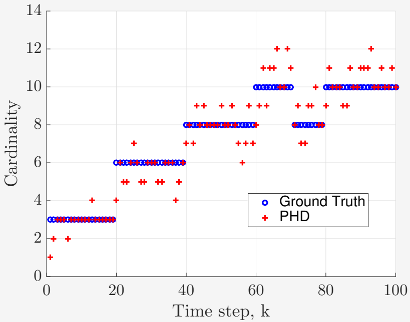
A SIMULATION EXAMPLE (3)

A GM-PHD simulation example

- The PHD filter outputs fairly reasonable estimates.



- Still, the filter yields many missed/false objects.



Section 6:
Metrics in MOT

Multi-Object Tracking

Lennart Svensson

Metrics for performance evaluation

Multi-Object Tracking

Lennart Svensson

- Our MOT algorithms output estimates $\hat{\mathbf{x}}_k$ of $\mathbf{x}_k$.
- How can we evaluate how accurate an estimator $\hat{\mathbf{x}}_k$ is?
  $\rightsquigarrow$ Which algorithm is the best?

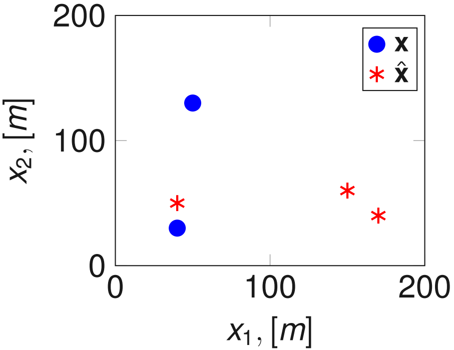
Key question

- How close is $\hat{\mathbf{x}}_k$ to $\mathbf{x}_k$?

- **Note:** both $\hat{\mathbf{x}}_k$ and $\mathbf{x}_k$ are sets.

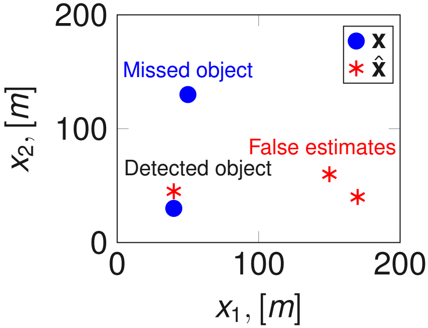
Objective

- Find a metric $d(\hat{\mathbf{x}}, \mathbf{x})$, suitable for MOT.



METRICS ON SETS (2)

- **Objective:** find a metric that grows with
  - localisation error for "properly detected objects",
  - $\sharp$ missed objects,
  - $\sharp$ false objects.
- We use the *generalised optimal sub-pattern assignment* (GOSPA) metric.



Informal definition

$$\text{GOSPA} = \text{localisation error} + \frac{c}{2}\left(\sharp\text{missed objects} + \sharp\text{false objects}\right)$$

METRICS AND NORMS

Metrics: definition

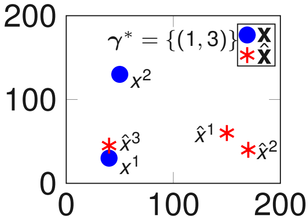
- A metric (on some space) is a distance function that satisfies
  1. $d(x, y) \geq 0$
  2. $d(x, y) = 0$ if and only if $x = y$
  3. $d(x, y) = d(y, x)$
  4. $d(x, y) \leq d(x, z) + d(z, y)$

- For $x, y \in \mathbb{R}^n$, the L^p-norm,
$$\|x\|_p = \left(|x_1|^p + |x_2|^p + \cdots + |x_n|^p\right)^{1/p}$$
can be used to define metrics.

- In our examples below, we use the **Euclidean distance**
$$d(x, y) = \|x - y\|_2 = \sqrt{(x - y)^T (x - y)}.$$

HOW TO COMPUTE GOSPA?

- Computing GOSPA ($p = 1$):
  1) Find optimal assignments between sets.
     Remark 1: pairs are left unassigned if $d(x, \hat{x}) > c$.
     Remark 2: we refer to unassigned elements as false/missed objects.
  2) Assigned pairs cost $d(x, \hat{x})$.
  3) Unassigned elements cost $c/2$.



- If $c = 40$, GOSPA$=$ $15 + 3 \times c/2 = 75$.

Formal definition, GOSPA, $\alpha = 2$

$$d_p^{(c,2)}(\mathbf{x}, \hat{\mathbf{x}}) = \left[\min_{\gamma \in \boldsymbol{\Gamma}} \left(\sum_{(i,j)\in\gamma} d(x^i, \hat{x}^j)^p + \frac{c^p}{2} \left(\underbrace{|\mathbf{x}| - |\gamma|}_{\sharp\text{missed}} + \underbrace{|\hat{\mathbf{x}}| - |\gamma|}_{\sharp\textit{false}} \right) \right) \right]^{\frac{1}{p}}$$

where $\boldsymbol{\Gamma}$ is the set of possible assignment sets.

Examples of GOSPA

Multi-Object Tracking

Lennart Svensson

GOSPA, EXAMPLES (1)

- Recall the definition of GOSPA, $p = 1$:

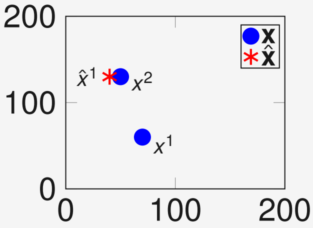
$$d_1^{(c,2)}(\mathbf{x}, \hat{\mathbf{x}}) = \min_{\gamma \in \mathbf{\Gamma}} \left(\sum_{(i,j)\in\gamma} d(x^i, \hat{x}^j) + \frac{c}{2} \left(\underbrace{|\mathbf{x}| - |\gamma|}_{\sharp\text{missed}} + \underbrace{|\hat{\mathbf{x}}| - |\gamma|}_{\sharp \textit{false}} \right) \right)$$

where $\mathbf{\Gamma}$ is the set of possible assignment sets.

Example: GOSPA, one missed object

- Suppose $p = 1$ and $c = 40$.
- Optimal assignment: $\gamma^* = \{(2, 1)\}$.
- GOSPA is

$$\begin{aligned} d_1^{(40,2)}(\mathbf{x}, \hat{\mathbf{x}}) &= d(x^2, \hat{x}^1) + c/2 \\ &= 10 + 20 = 30. \end{aligned}$$



GOSPA, EXAMPLES (2)

- Recall the definition of GOSPA, $p = 1$:

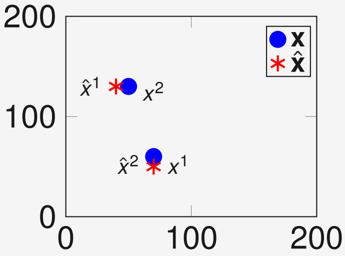
$$d_1^{(c,2)}(\mathbf{x},\hat{\mathbf{x}}) = \min_{\gamma\in\mathbf{\Gamma}} \left(\sum_{(i,j)\in\gamma} d(x^i,\hat{x}^j) + \frac{c}{2}\left(\underbrace{|\mathbf{x}| - |\gamma|}_{\sharp\text{missed}} + \underbrace{|\hat{\mathbf{x}}| - |\gamma|}_{\sharp\textit{false}} \right)\right)$$

where $\mathbf{\Gamma}$ is the set of possible assignment sets.

Example: GOSPA, two properly detected objects

- Suppose $p = 1$ and $c = 40$.
- Optimal assign.: $\gamma^* = \{(2,1),(1,2)\}$.
- GOSPA is

$$d_1^{(40,2)}(\mathbf{x},\hat{\mathbf{x}}) = d(x^2,\hat{x}^1) + d(x^1,\hat{x}^2)$$
$$= 10 + 10 = 20.$$



GOSPA, EXAMPLES (3)

- Recall the definition of GOSPA, $p = 1$:

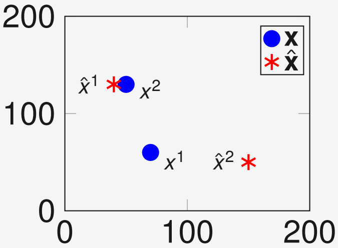
$$d_1^{(c,2)}(\mathbf{x}, \hat{\mathbf{x}}) = \min_{\gamma \in \boldsymbol{\Gamma}} \left(\sum_{(i,j) \in \gamma} d(x^i, \hat{x}^j) + \frac{c}{2} \left(\underbrace{|\mathbf{x}| - |\gamma|}_{\sharp\text{missed}} + \underbrace{|\hat{\mathbf{x}}| - |\gamma|}_{\sharp \textit{false}} \right) \right)$$

where $\boldsymbol{\Gamma}$ is the set of possible assignment sets.

Example: GOSPA, missed and false object

- Suppose $p = 1$ and $c = 40$.
- Optimal assignment: $\gamma^* = \{(2, 1)\}$.
- GOSPA is

$$d_1^{(40,2)}(\mathbf{x}, \hat{\mathbf{x}}) = d(x^2, \hat{x}^1) + 2 \times \frac{c}{2}$$
$$= 10 + 40 = 50.$$



CONCLUSIONS FROM EXAMPLES

- We used GOSPA to compare three estimates for the same set $\mathbf{x}$.
- The true set $\mathbf{x}$ contained two objects.
- We obtained the smallest metric when both objects were properly detected.
- GOSPA took a larger value when one object was missed and an even larger value when we also had a false object.

GOSPA for RFSs

Multi-Object Tracking

Lennart Svensson

GOSPA FOR RFSs (1)

- In tracking, the set of objects and estimates are (often) RFSs.
- To evaluate tracking algorithms we need metrics between RFSs!

Key result: GOSPA metrics for RFSs

- For $1 \leq p, p' < \infty$

$$\sqrt[p']{\mathbb{E}\left[d_p^{(c,2)}(\mathbf{x}, \hat{\mathbf{x}})^{p'}\right]},$$

where $\mathbf{x}$ and $\hat{\mathbf{x}}$ are RFSs, is a metric.

- We are particularly interested in cases where $p = p'$.

GOSPA FOR RFSs (2)

- We know that $\sqrt[p']{\mathbb{E}\left[d_p^{(c,2)}(\mathbf{x},\hat{\mathbf{x}})^{p'}\right]}$ is a metric for $1 \leq p, p' < \infty$.

Mean GOSPA

- Setting $p = p' = 1$ gives that **mean GOSPA**

$$\mathbb{E}\left[d_1^{(c,2)}(\mathbf{x},\hat{\mathbf{x}})\right]$$

is a metric.

Root mean squared GOSPA (RMS-GOSPA)

- Setting $p = p' = 2$ gives that **root mean squared GOSPA**

$$\sqrt{\mathbb{E}\left[d_2^{(c,2)}(\mathbf{x},\hat{\mathbf{x}})^2\right]}$$

is a metric. **Note:** mean squared GOSPA is not a metric.

DECOMPOSING GOSPA FOR RFSs

- Let γ^* denote the optimal assignment in the GOSPA metric (a RFS).

Decomposing GOSPA

- For any $1 \leq p < \infty$, the following is a metric

$$\sqrt[p]{\mathbb{E}\left[d_p^{(c,2)}(\mathbf{x},\hat{\mathbf{x}})^p\right]} = \sqrt[p]{\underbrace{\mathbb{E}\left[\sum_{(i,j)\in\gamma^*} d(x^i,\hat{x}^j)^p\right]}_{\text{localisation}^p} + \underbrace{\frac{c^p}{2}\mathbb{E}\left[|\mathbf{x}| - |\gamma^*|\right]}_{\text{missed}^p} + \underbrace{\frac{c^p}{2}\mathbb{E}\left[|\hat{\mathbf{x}}| - |\gamma^*|\right]}_{\text{false}^p}}.$$

- **Proof:** Setting $p = p'$ $\Rightarrow$ the left hand side is a metric.
- The result then follows from

$$d_p^{(c,2)}(\mathbf{x},\hat{\mathbf{x}})^p = \sum_{(i,j)\in\gamma^*} d(x^i,\hat{x}^j)^p + \frac{c^p}{2}\left(|\mathbf{x}| - |\gamma^*|\right) + \frac{c^p}{2}\left(|\hat{\mathbf{x}}| - |\gamma^*|\right)$$

DECOMPOSING GOSPA FOR RFSs

- Let γ^* denote the optimal assignment in the GOSPA metric (a RFS).

Decomposing GOSPA

- For any $1 \leq p < \infty$, the following is a metric

$$\sqrt[p]{\mathbb{E}\left[d_p^{(c,2)}(\mathbf{x}, \hat{\mathbf{x}})^p\right]} = \sqrt[p]{\underbrace{\mathbb{E}\left[\sum_{(i,j)\in\gamma^*} d(x^i, \hat{x}^j)^p\right]}_{\text{localisation}^p} + \underbrace{\frac{c^p}{2}\mathbb{E}\left[|\mathbf{x}| - |\gamma^*|\right]}_{\text{missed}^p} + \underbrace{\frac{c^p}{2}\mathbb{E}\left[|\hat{\mathbf{x}}| - |\gamma^*|\right]}_{\text{false}^p}}.$$

- In particular, both **mean GOSPA and RMS-GOSPA decompose** as above.
- This enables us to **analyse error sources**!

GOSPA FOR RFSs: SIMULATION EXAMPLE

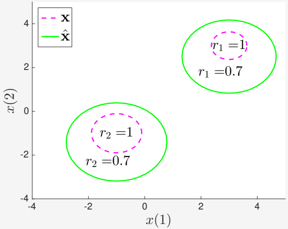
RMS-GOSPA for two MBs

- Suppose $\mathbf{x}$ is a MB RFS with

$$r_1 = r_2 = 1$$
$$p_1(x) = \mathcal{N}(x; [3,\ 3]^T, 0.1\, I)$$
$$p_2(x) = \mathcal{N}(x; [-1,\ -1]^T, 0.2\, I)$$

- Also, suppose $\hat{\mathbf{x}}$ is a MB RFS with

$$\hat{r}_1 = \hat{r}_2 = 0.7$$
$$\hat{p}_1(x) = \mathcal{N}(x; [2.5,\ 2.5]^T, 0.7\, I)$$
$$\hat{p}_2(x) = \mathcal{N}(x; [-1.5,\ -1.4]^T, 0.8\, I)$$



- Using $p = 2$ and $c = 3$, we get

$$\text{RMS-GOSPA} \approx 2.4, \quad \text{false} \approx 0.3,$$
$$\text{localisation} \approx 1.7, \quad \text{missed} \approx 1.7.$$

- **Note:** $\text{RMS-GOSPA} = \sqrt{\text{localisation}^2 + \text{missed}^2 + \text{false}^2}$.

SUMMARY

- GOSPA is a metric between sets of points. For $p = 1$,

$$\text{GOSPA} = \text{localisation error} + \frac{c}{2}\left(\sharp\text{missed objects} + \sharp\text{false objects}\right)$$

- GOSPA penalises false and missed object estimates.
- Efficiently computed using Hungarian/auction algorithms.
- Both mean GOSPA and RMS-GOSPA are metrics on RFSs.