Interval State Estimation
by Solving Data Association

Simon Rohou, Benoît Desrochers, Luc Jaulin

ENSTA Bretagne, Lab-STICC, UMR CNRS 6285, Brest, France

International Online Seminar
on Interval Methods in Control Engineering
July 2021

Section 1

Set membership tools for state estimation

Set membership state estimation

Classical state estimation problem:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) & \text{(evolution equation)} \\ \mathbf{y}^i = \mathbf{g}\big(\mathbf{x}(t_i)\big) & \text{(observation equation)} \end{cases}$$

Set membership state estimation

Classical state estimation problem:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) & \text{(evolution equation)} \\ \mathbf{y}^i = \mathbf{g}\big(\mathbf{x}(t_i)\big) & \text{(observation equation)} \end{cases}$$

with:

- $\mathbf{x}(\cdot)$, $\mathbf{u}(\cdot)$, uncertain trajectories
- $\mathbf{y}^i$, an input measurement vector

Set membership state estimation

Classical state estimation problem:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) & \text{(evolution equation)} \\ \mathbf{y}^i = \mathbf{g}\big(\mathbf{x}(t_i)\big) & \text{(observation equation)} \end{cases}$$

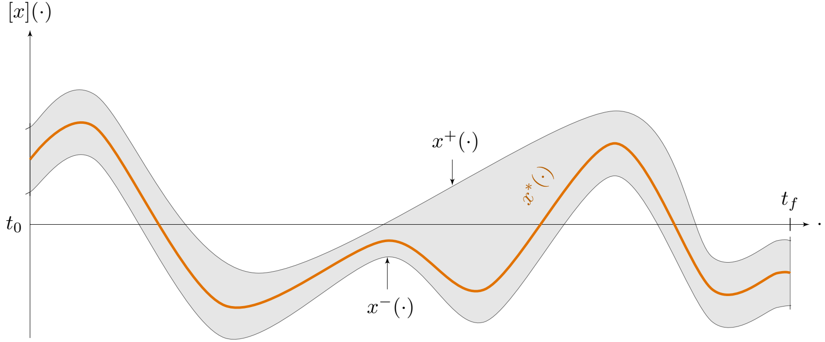
with:

- $\mathbf{x}(\cdot)$, $\mathbf{u}(\cdot)$, uncertain trajectories
- $\mathbf{y}^i$, an input measurement vector

and associated sets:

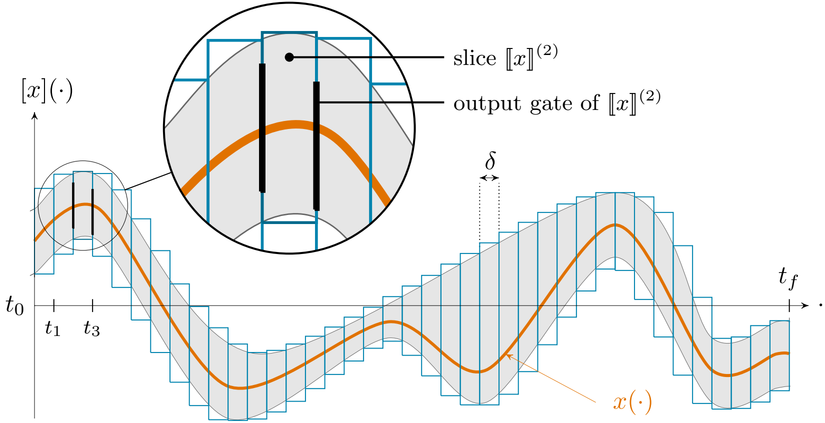
- $[\mathbf{x}](\cdot)$, $[\mathbf{u}](\cdot)$: tubes
- $[\mathbf{y}^i]$: an interval

Tubes



Example of scalar tube

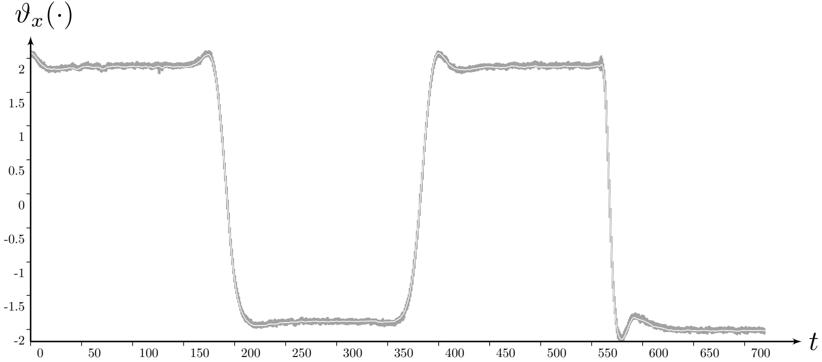
Tubes



Computer implementation (http://codac.io)

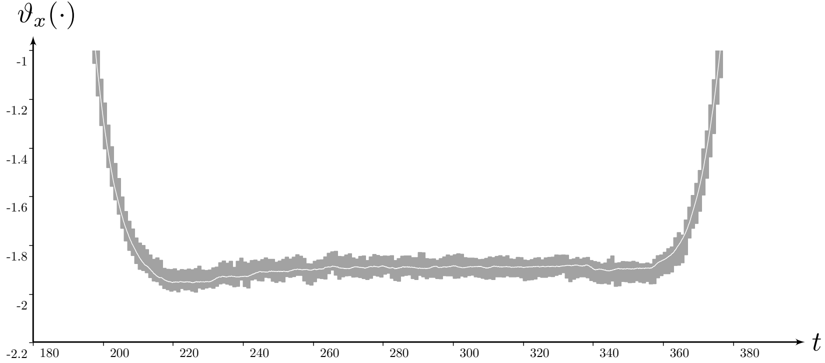
Example: velocity sensing

East velocity given by a DVL + IMU sensors:



Example: velocity sensing

East velocity given by a DVL + IMU sensors (zoom):

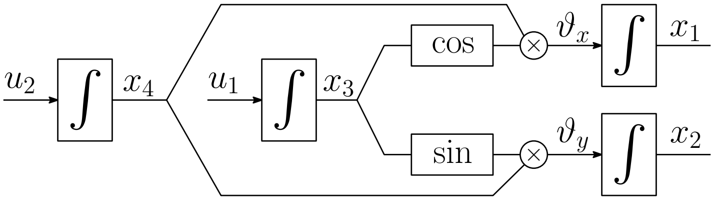


Decomposition of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

Decomposition of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

State equation:

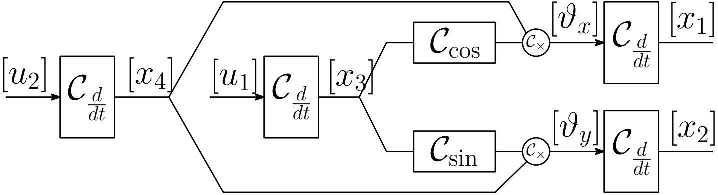
$$\left\{ \begin{array}{rccl} (i) & \dot{x}_1 & = & x_4 \cos(x_3) \\ (ii) & \dot{x}_2 & = & x_4 \sin(x_3) \\ (iii) & \dot{x}_3 & = & u_1 \\ (iv) & \dot{x}_4 & = & u_2 \end{array} \right.$$



Decomposition of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

State equation:

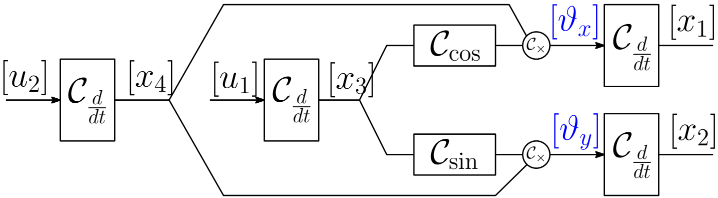
$$\begin{cases} (i) & \dot{x}_1 = x_4 \cos(x_3) \\ (ii) & \dot{x}_2 = x_4 \sin(x_3) \\ (iii) & \dot{x}_3 = u_1 \\ (iv) & \dot{x}_4 = u_2 \end{cases}$$



Decomposition of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

State equation:

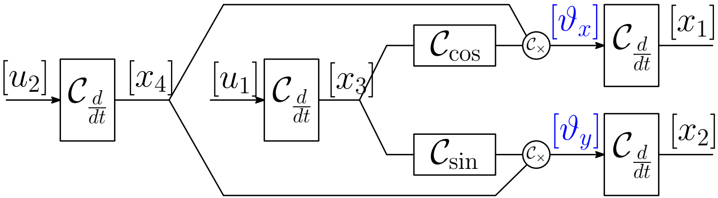
$$\begin{cases} (i) & \dot{x}_1 = x_4 \cos(x_3) & \Leftrightarrow \{\dot{x}_1 = \vartheta_x \; ; \; \vartheta_x = x_4 \cos(x_3)\} \\ (ii) & \dot{x}_2 = x_4 \sin(x_3) & \Leftrightarrow \{\dot{x}_2 = \vartheta_y \; ; \; \vartheta_y = x_4 \sin(x_3)\} \\ (iii) & \dot{x}_3 = u_1 \\ (iv) & \dot{x}_4 = u_2 \end{cases}$$



Decomposition of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

State equation:

$$\left\{\begin{array}{rlcl} (i) & \dot{x}_1 & = & x_4\cos(x_3) \\ (ii) & \dot{x}_2 & = & x_4\sin(x_3) \\ (iii) & \dot{x}_3 & = & u_1 \\ (iv) & \dot{x}_4 & = & u_2 \end{array}\right. \quad \begin{array}{l} \Leftrightarrow \{\dot{x}_1 = \vartheta_x \ ; \ \vartheta_x = x_4\cos(x_3)\} \\ \Leftrightarrow \{\dot{x}_2 = \vartheta_y \ ; \ \vartheta_y = x_4\sin(x_3)\} \\ \\ \\ \end{array}$$

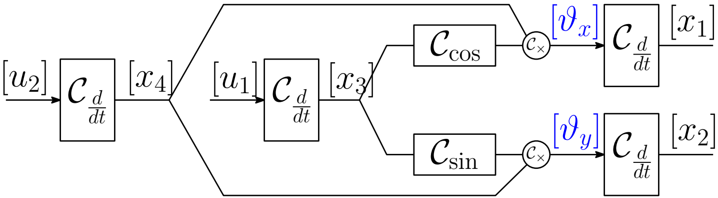


Involved operators: $\mathcal{C}_{\times}$, $\mathcal{C}_{\cos}$, $\mathcal{C}_{\sin}$, $\mathcal{C}_{\frac{d}{dt}}$

Decomposition of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

State equation:

$$\begin{cases} (i) & \dot{x}_1 = x_4 \cos(x_3) & \Leftrightarrow \{\dot{x}_1 = \vartheta_x \; ; \; \vartheta_x = x_4 \cos(x_3)\} \\ (ii) & \dot{x}_2 = x_4 \sin(x_3) & \Leftrightarrow \{\dot{x}_2 = \vartheta_y \; ; \; \vartheta_y = x_4 \sin(x_3)\} \\ (iii) & \dot{x}_3 = u_1 & \\ (iv) & \dot{x}_4 = u_2 & \end{cases}$$



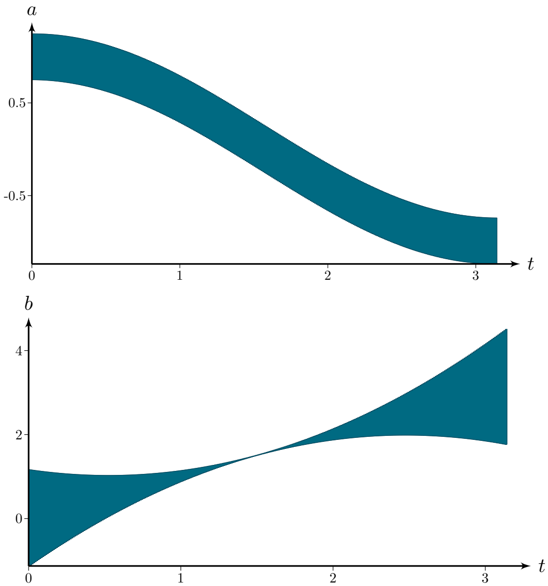
Involved operators: $\mathcal{C}_{\times}$, $\mathcal{C}_{\cos}$, $\mathcal{C}_{\sin}$, $\mathcal{C}_{\frac{d}{dt}}$

Involved sets: $[\mathbf{x}](\cdot)$, $[\mathbf{u}](\cdot)$, $[\vartheta_x](\cdot)$, $[\vartheta_y](\cdot)$

"Static" constraints

Static constraint:

- $\forall t,\ f(a(\cdot), b(\cdot), \dots) = 0$
- non differential (not in the form $\dot{a}(t) = b(t)$)
- non inter-temporal (not in the form $a(t+1) = b(t)$)



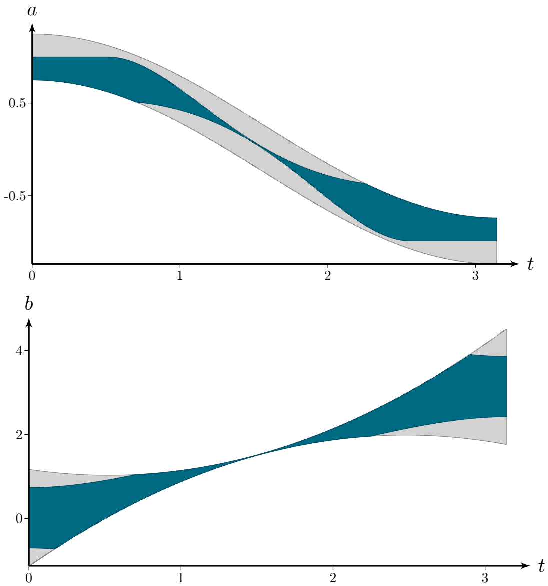
"Static" constraints

Static constraint:

- $\forall t,\ f(a(\cdot), b(\cdot), \dots) = 0$
- non differential (not in the form $\dot{a}(t) = b(t)$)
- non inter-temporal (not in the form $a(t+1) = b(t)$)

Example with $a(\cdot) = \cos\big(b(\cdot)\big)$:

$$\mathcal{C}_{\cos}\big([a](\cdot), [b](\cdot)\big)$$



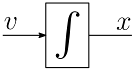
"Static" constraints

A definition of the $\mathcal{C}_+$ operator for tubes for the constraint $a(\cdot) = x(\cdot) + y(\cdot)$:

$$\mathcal{C}_+\big([a](\cdot), [x](\cdot), [y](\cdot)\big)$$

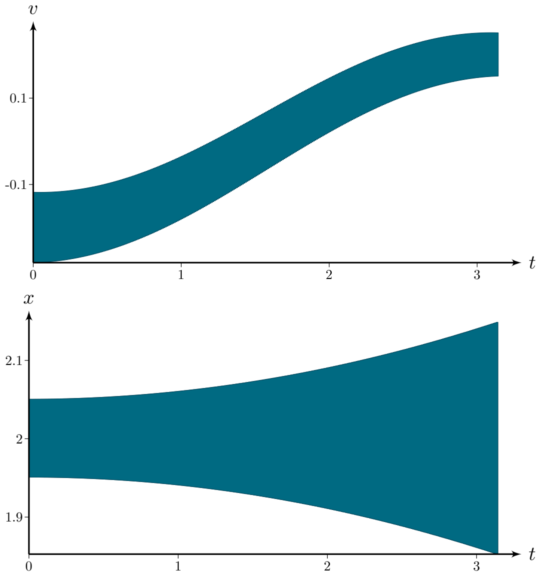
$$\begin{pmatrix} [a](t) \\ [x](t) \\ [y](t) \end{pmatrix} \xmapsto{\mathcal{C}_+} \begin{pmatrix} [a](t) \cap \big([x](\cdot) + [y](\cdot)\big) \\ [x](t) \cap \big([a](\cdot) - [y](\cdot)\big) \\ [y](t) \cap \big([a](\cdot) - [x](\cdot)\big) \end{pmatrix}$$

Derivative constraint

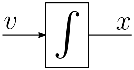


Differential constraint:

- $\dot{\mathbf{x}}(\cdot) = \mathbf{v}(\cdot)$
- one trajectory and its derivative



Derivative constraint

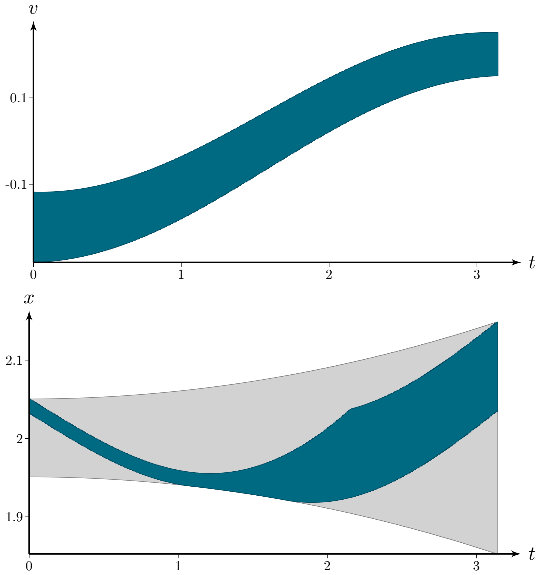


Differential constraint:

- $\dot{\mathbf{x}}(\cdot) = \mathbf{v}(\cdot)$
- one trajectory and its derivative

Contractor on tubes:

$$\mathcal{C}_{\frac{d}{dt}}\big([\mathbf{x}](\cdot), [\mathbf{v}](\cdot)\big)$$



Guaranteed computation of robot trajectories

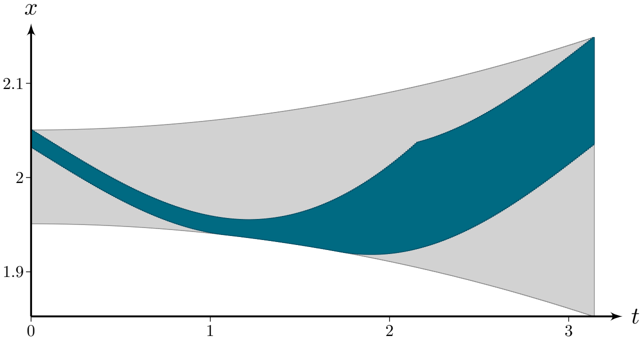
Rohou, Jaulin, Mihaylova, Le Bars, Veres

Robotics and Autonomous Systems, 2017

Derivative constraint

Definition of the $\mathcal{C}_{\frac{d}{dt}}$ operator:

$$\begin{pmatrix} [x](t) \\ [v](t) \end{pmatrix} \xmapsto{\mathcal{C}_{\frac{d}{dt}}} \begin{pmatrix} \bigcap_{t_1=t_0}^{t_f} \left([x](t_1) + \int_{t_1}^{t} [v](\tau) d\tau \right) \\ [v](t) \end{pmatrix} \quad (1)$$



Propagations using Contractor Networks

Contractor Programming

- $\mathcal{C}_+\big([z^j],[c],[b]\big)$
- $\mathcal{C}_+\big([\mathbf{x}],[v],[w]\big)$
- $\mathcal{C}_{\cos}\big([\mathbf{x}],[b]\big)$
- $\mathcal{C}_{\mathbf{h}}\big([c],[\mathbf{u}],[y^i],[w]\big)$
- $\mathcal{C}_{\frac{d}{dt}}\big([a],[b]\big)$
- $\mathcal{C}_{\frac{d}{dt}}\big([c],[v]\big)$

(abstract example)

Propagations using Contractor Networks

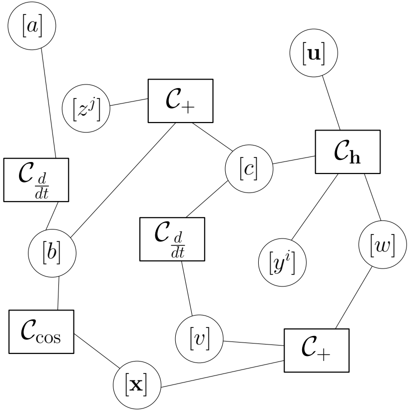
Contractor Programming

- $\mathcal{C}_{+}\big([z^j],[c],[b]\big)$
- $\mathcal{C}_{+}\big([\mathbf{x}],[v],[w]\big)$
- $\mathcal{C}_{\cos}\big([\mathbf{x}],[b]\big)$
- $\mathcal{C}_{\mathbf{h}}\big([c],[\mathbf{u}],[y^i],[w]\big)$
- $\mathcal{C}_{\frac{d}{dt}}\big([a],[b]\big)$
- $\mathcal{C}_{\frac{d}{dt}}\big([c],[v]\big)$

$\Leftrightarrow$

Contractor Networks

(http://codac.io)



(abstract example)

Decomposition and wrapping effects

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \Longleftrightarrow \left\{ \begin{array}{l} \mathbf{v} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \dot{\mathbf{x}} = \mathbf{v} \end{array} \right. \Longrightarrow \left\{ \begin{array}{l} \mathcal{C}_{\mathbf{f}}\big([\mathbf{v}](\cdot), [\mathbf{x}](\cdot), [\mathbf{u}](\cdot)\big) \\ \mathcal{C}_{\frac{d}{dt}}\big([\mathbf{x}](\cdot), [\mathbf{v}](\cdot)\big) \end{array} \right.$$

Decomposition and wrapping effects

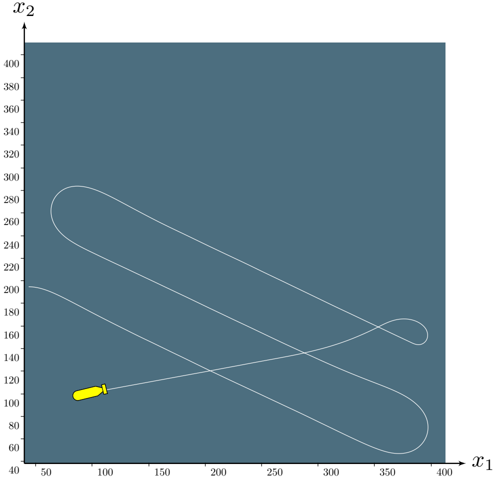
$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u}) \Longleftrightarrow \begin{cases} \mathbf{v} = \mathbf{f}(\mathbf{x},\mathbf{u}) \\ \dot{\mathbf{x}} = \mathbf{v} \end{cases} \Longrightarrow \begin{cases} \mathcal{C}_{\mathbf{f}}\big([\mathbf{v}](\cdot),[\mathbf{x}](\cdot),[\mathbf{u}](\cdot)\big) \\ \mathcal{C}_{\frac{d}{dt}}\big([\mathbf{x}](\cdot),[\mathbf{v}](\cdot)\big) \end{cases}$$

See also: more efficient contractors without decomposition,
e.g. $\mathcal{C}_{\text{Lohner}}$ for dealing with $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$

▪ Safe and collaborative autonomous underwater docking

Auguste Bourgois *PhD thesis*, 2021

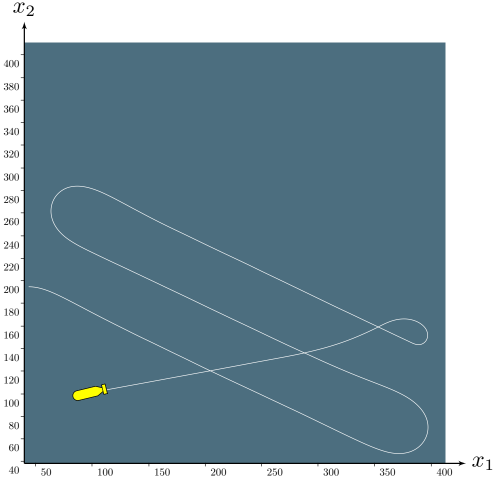
Dynamic state estimation



State estimation:

$$\left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \\ \end{array} \right.$$

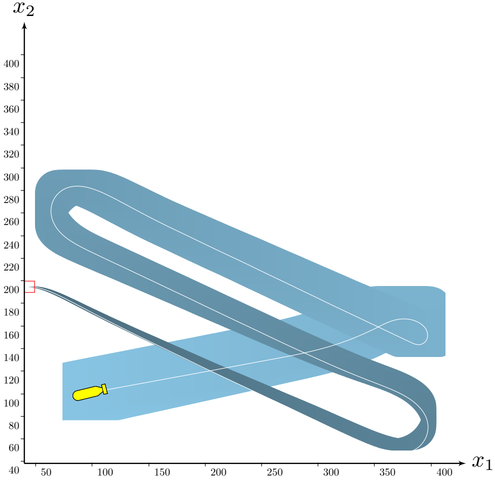
Dynamic state estimation



State estimation:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \dot{\mathbf{x}}(t) = \mathbf{v}(t) \end{cases}$$

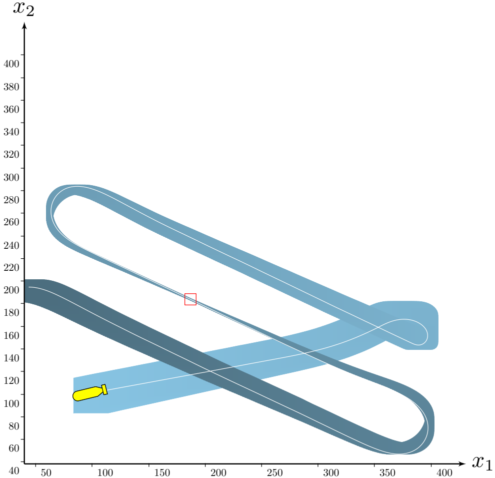
Dynamic state estimation



State estimation:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \dot{\mathbf{x}}(t) = \mathbf{v}(t) \\ \mathbf{x}(t_0) \in [\mathbf{x}_0] \end{cases}$$

Dynamic state estimation

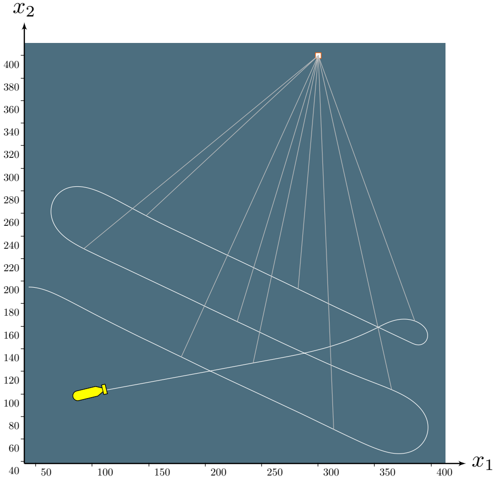


State estimation:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \dot{\mathbf{x}}(t) = \mathbf{v}(t) \\ \mathbf{x}(t_1) \in [\mathbf{x}_1] \end{cases}$$

Dynamic state estimation

Considering **range-only measurements** from a known beacon.

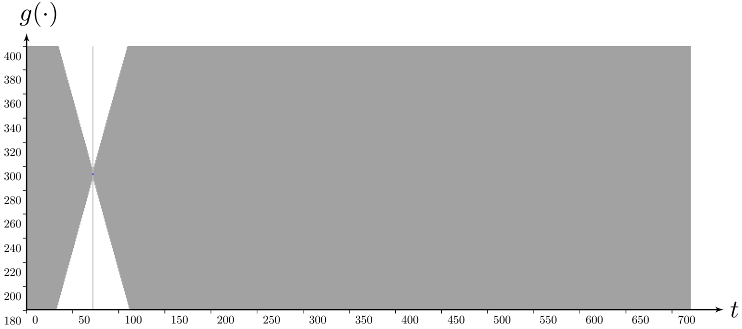


Non-linear state estimation:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ y_i = g\big(\mathbf{x}(t_i)\big) \end{cases}$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



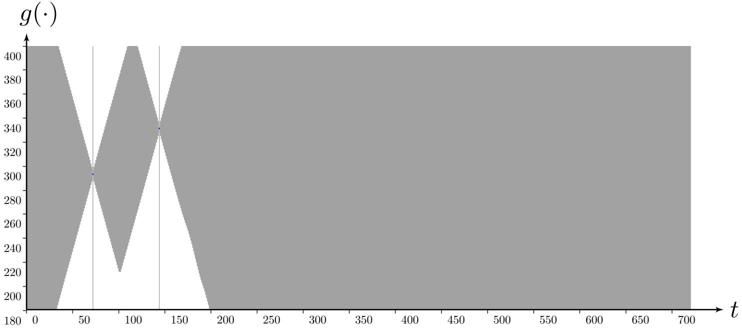
Observation tube, considering 1 range-only measurement from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g: \quad g(\cdot) = \sqrt{\big(x_1(\cdot) - \mathcal{B}_1\big)^2 + \big(x_2(\cdot) - \mathcal{B}_2\big)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



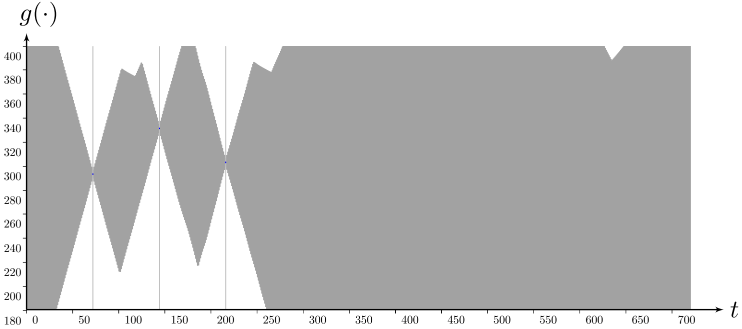
Observation tube, considering 2 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g : \quad g(\cdot) = \sqrt{\big(x_1(\cdot) - \mathcal{B}_1\big)^2 + \big(x_2(\cdot) - \mathcal{B}_2\big)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



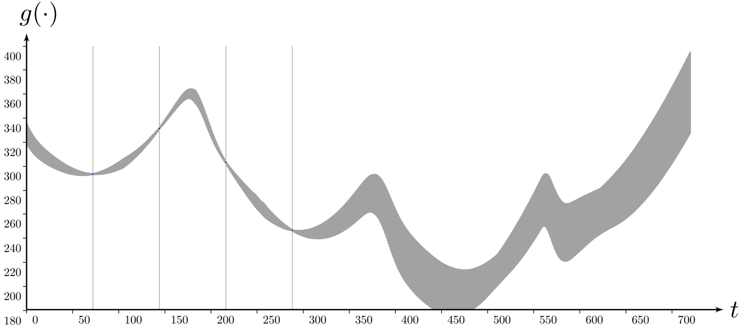
Observation tube, considering 3 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g : \quad g(\cdot) = \sqrt{\big(x_1(\cdot) - \mathcal{B}_1\big)^2 + \big(x_2(\cdot) - \mathcal{B}_2\big)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



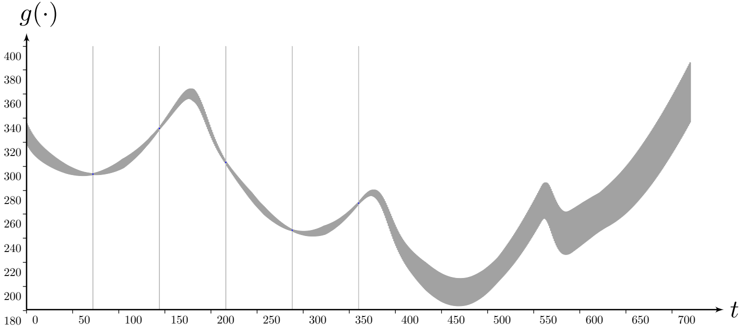
Observation tube, considering 4 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g: \quad g(\cdot) = \sqrt{\left(x_1(\cdot) - \mathcal{B}_1\right)^2 + \left(x_2(\cdot) - \mathcal{B}_2\right)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



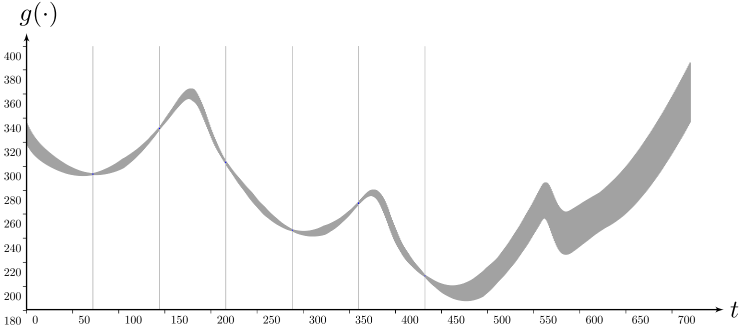
Observation tube, considering 5 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g : \quad g(\cdot) = \sqrt{\left(x_1(\cdot) - \mathcal{B}_1\right)^2 + \left(x_2(\cdot) - \mathcal{B}_2\right)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



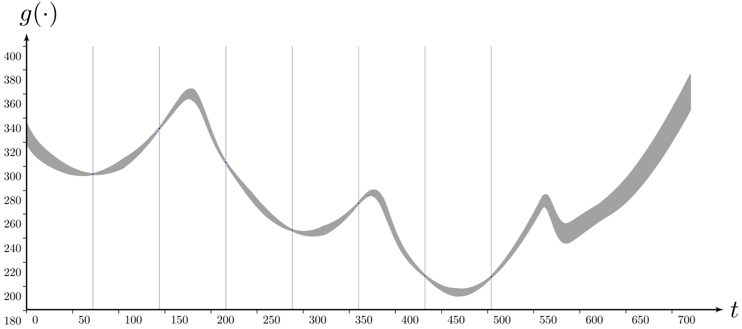
Observation tube, considering 6 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g : \quad g(\cdot) = \sqrt{\big(x_1(\cdot) - \mathcal{B}_1\big)^2 + \big(x_2(\cdot) - \mathcal{B}_2\big)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



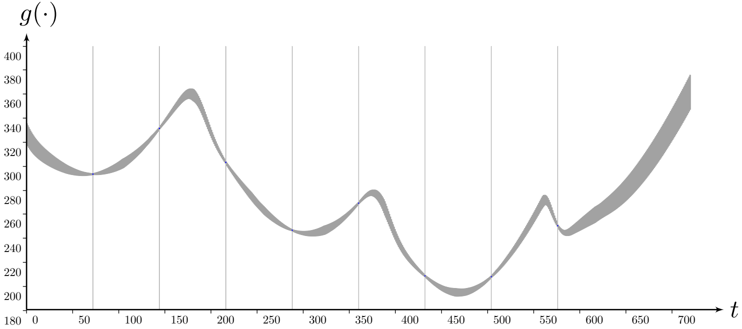
Observation tube, considering 7 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g : \quad g(\cdot) = \sqrt{\big(x_1(\cdot) - \mathcal{B}_1\big)^2 + \big(x_2(\cdot) - \mathcal{B}_2\big)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



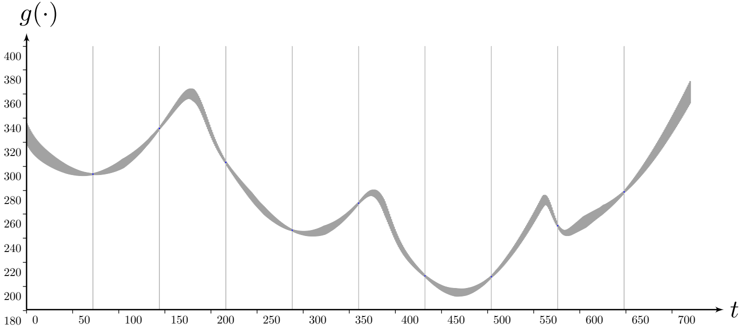
Observation tube, considering 8 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g : \quad g(\cdot) = \sqrt{\left(x_1(\cdot) - \mathcal{B}_1\right)^2 + \left(x_2(\cdot) - \mathcal{B}_2\right)^2}.$$

Exteroceptive measurements

Creating another tube $[g](\cdot)$ that will be **constrained by measurements**.



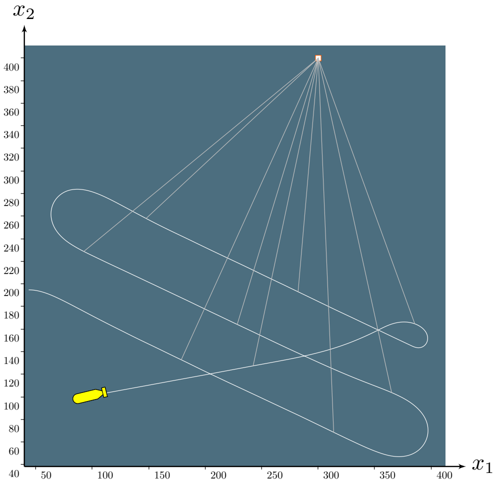
Observation tube, considering 9 range-only measurements from the beacon.

The state tube $[\mathbf{x}](\cdot)$ and $[g](\cdot)$ are constrained by

$$\mathcal{L}_g: \quad g(\cdot) = \sqrt{\left(x_1(\cdot) - \mathcal{B}_1\right)^2 + \left(x_2(\cdot) - \mathcal{B}_2\right)^2}.$$

Dynamic state estimation

Considering **range-only measurements** from a known beacon.

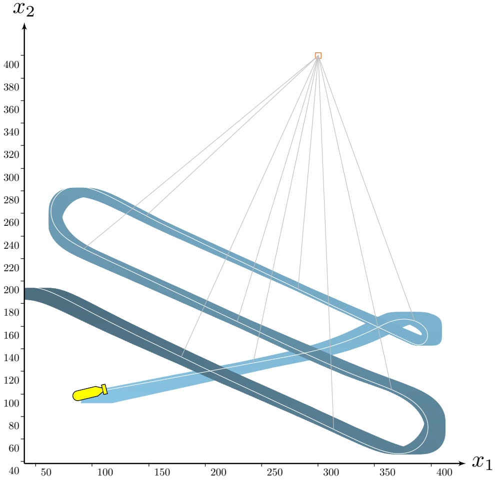


State estimation:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ y_i = g\big(\mathbf{x}(t_i)\big) \end{cases}$$

Dynamic state estimation

Considering **range-only measurements** from a known beacon.



State estimation:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ y_i = g\big(\mathbf{x}(t_i)\big) \end{cases}$$

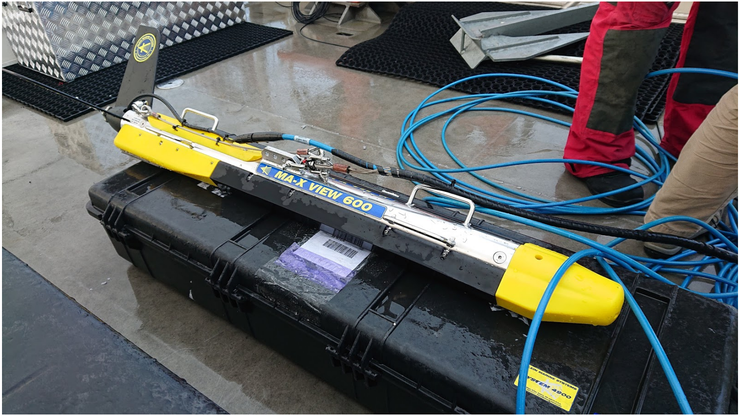
Section 2

Application to underwater robotics with indistinguishable landmarks

Video

Underwater robotics: sonar sensors

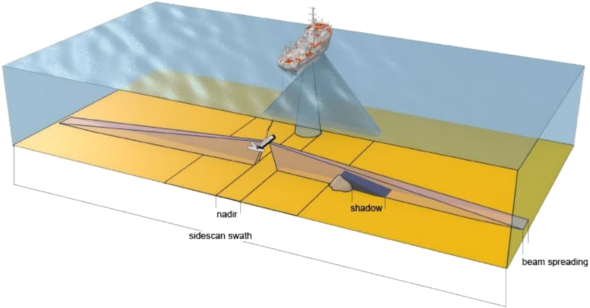
Side-scan sonars: to perceive objects on the seabed



A side-scan sonar Klein Max-View 600 during a demo in Brest, France.

Underwater robotics: sonar sensors

Side-scan sonars: to perceive objects on the seabed

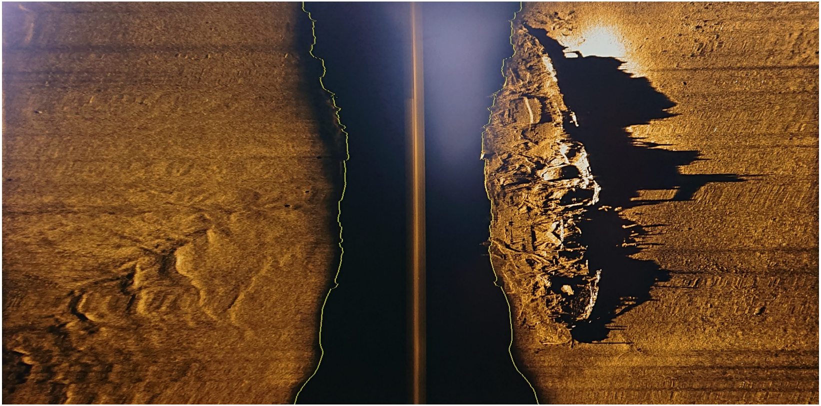


Schematic drawing illustrating the principles of a side-scan sonar.

Image from www.ga.gov.au

Underwater robotics: sonar sensors

Side-scan sonars: to perceive objects on the seabed



Perception of a wreck with the Klein Max-View 600.

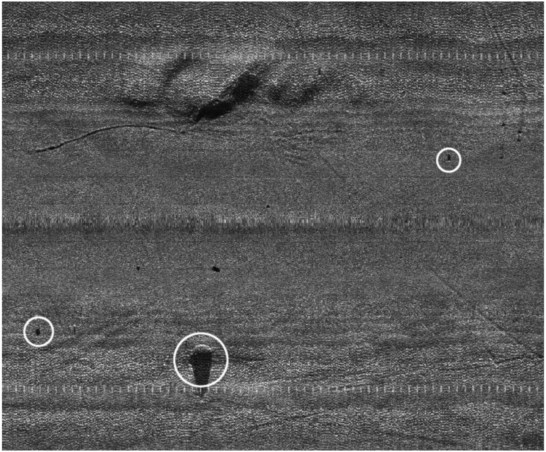
Underwater robotics: sonar sensors

Now, onboard of an **Autonomous Underwater Vehicle** (AUV):

Daurade Autonomous Underwater Vehicle (AUV).

Underwater robotics: sonar sensors

Now, onboard of an **Autonomous Underwater Vehicle** (AUV):



Detection of unidentifiable/indistinguishable rocks on the seabed.

Localization with data association: assumptions

1. **The map is static and made of point landmarks (rocks)**

Localization with data association: assumptions

1. **The map is static and made of point landmarks (rocks)**

2. **The landmarks are indistinguishable**
   All the rocks on the seabed look alike:

Localization with data association: assumptions

1. **The map is static and made of point landmarks (rocks)**

2. **The landmarks are indistinguishable**
   All the rocks on the seabed look alike:

3. **The position of each landmark is known (bounded)**

Localization with data association: assumptions

1. **The map is static and made of point landmarks (rocks)**

2. **The landmarks are indistinguishable**
   All the rocks on the seabed look alike:

3. **The position of each landmark is known (bounded)**

4. **The initial pose of the robot is not known**

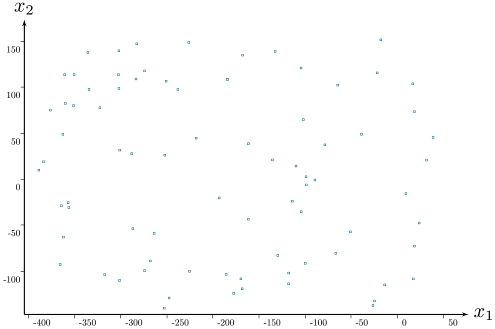
Localization with data association: assumptions

1. **The map is static and made of point landmarks (rocks)**
2. **The landmarks are indistinguishable**
   All the rocks on the seabed look alike:

3. **The position of each landmark is known (bounded)**
4. **The initial pose of the robot is not known**

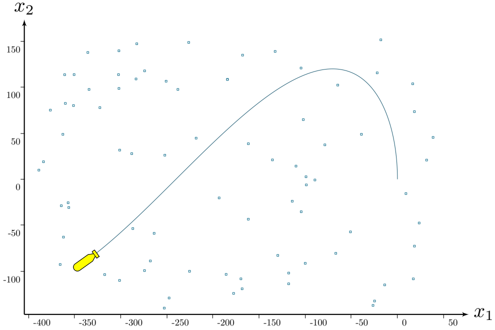
state estimation ⇄ data association

Localization with data association: formalization



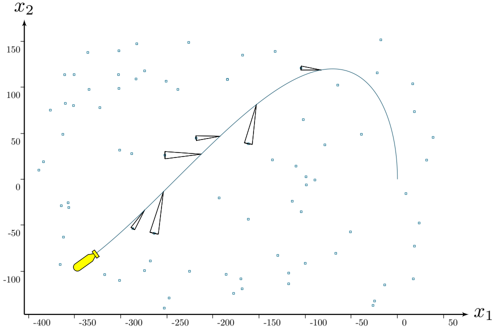
$$\left\{ \begin{array}{l} \mathbf{m}^i \in \mathbb{M} \\ \text{(map constraint)} \end{array} \right.$$

Localization with data association: formalization



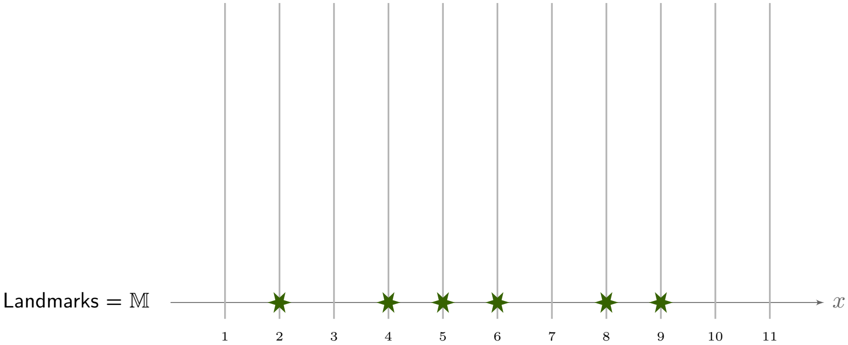
$$\begin{cases} \mathbf{m}^i \in \mathbb{M} \\ \text{(map constraint)} \\ \\ \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \text{(evolution equation)} \end{cases}$$

Localization with data association: formalization



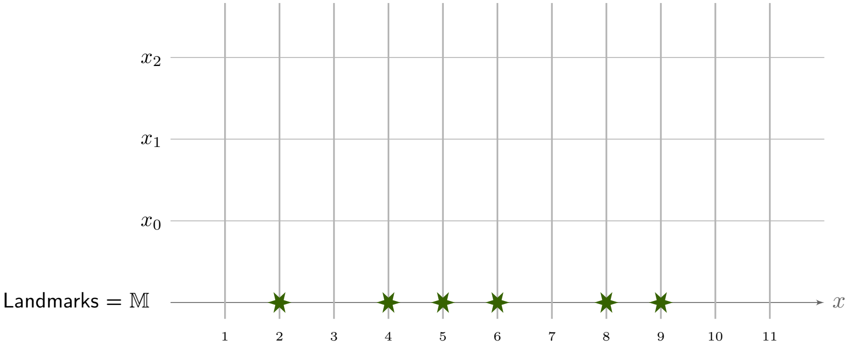
$$\begin{cases} \mathbf{m}^i \in \mathbb{M} \\ \text{(map constraint)} \\ \\ \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \text{(evolution equation)} \\ \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0} \\ \text{(observation equation)} \end{cases}$$

State estimation with indistinguishable landmarks



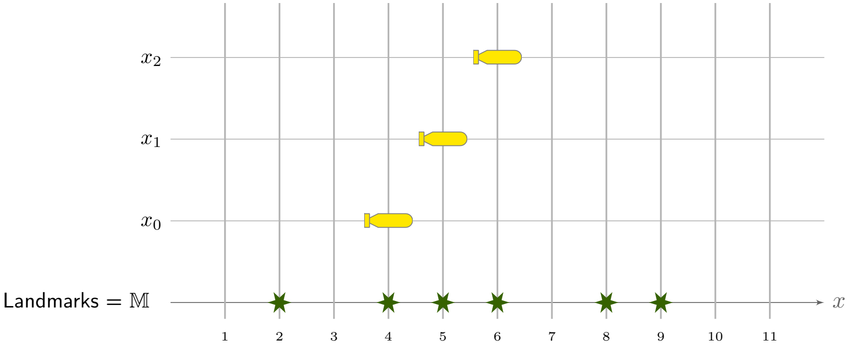
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



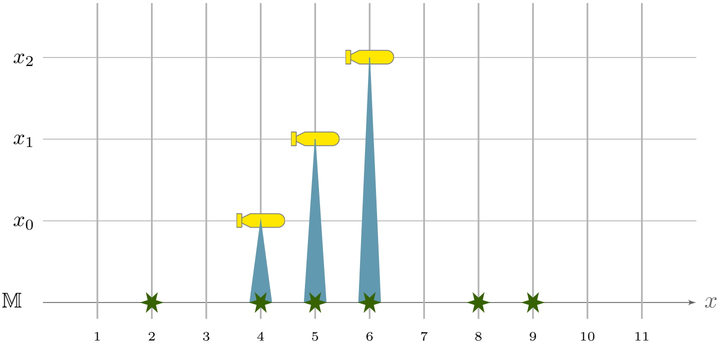
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



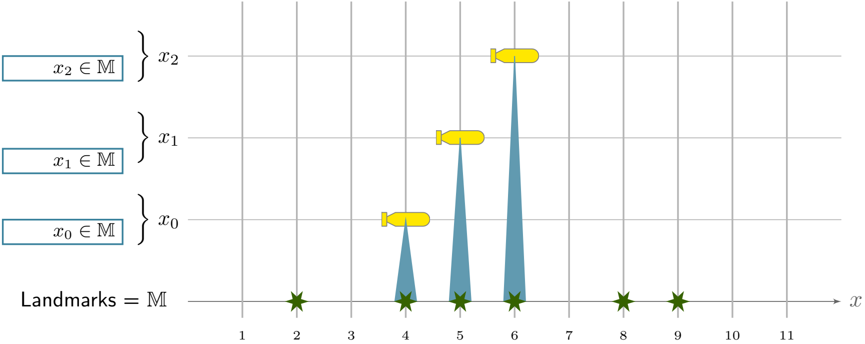
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



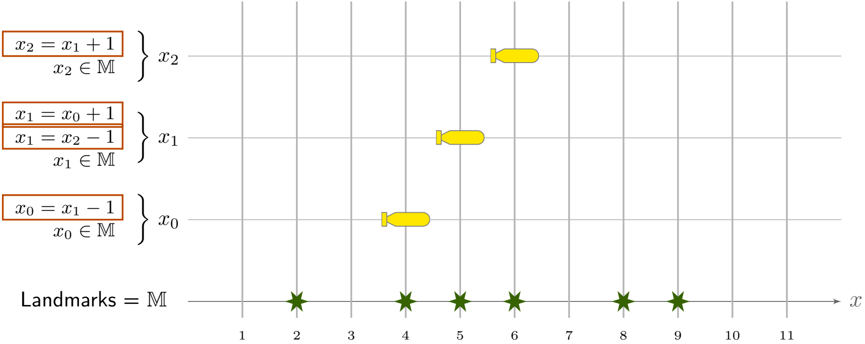
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



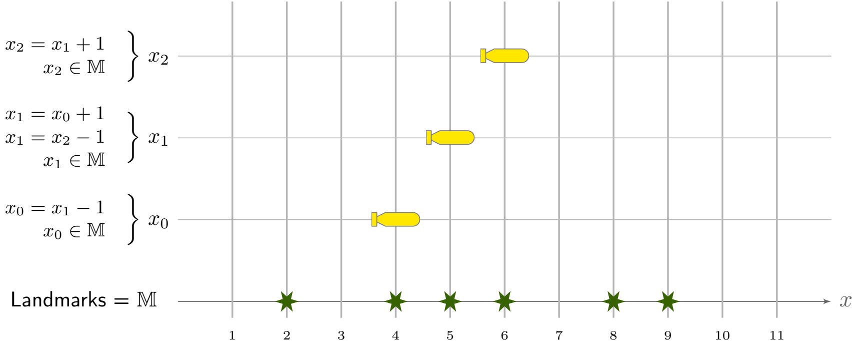
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



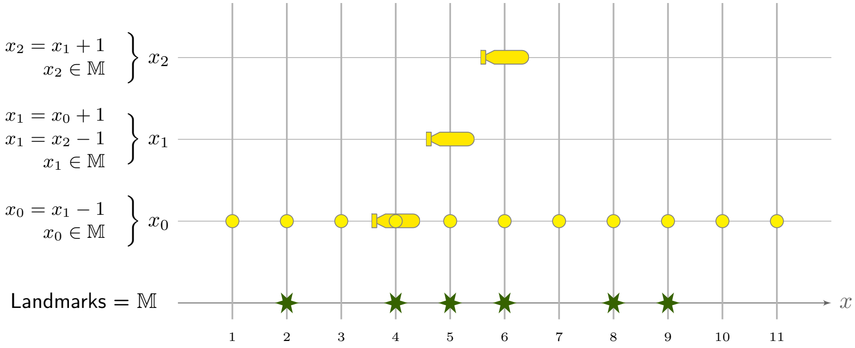
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



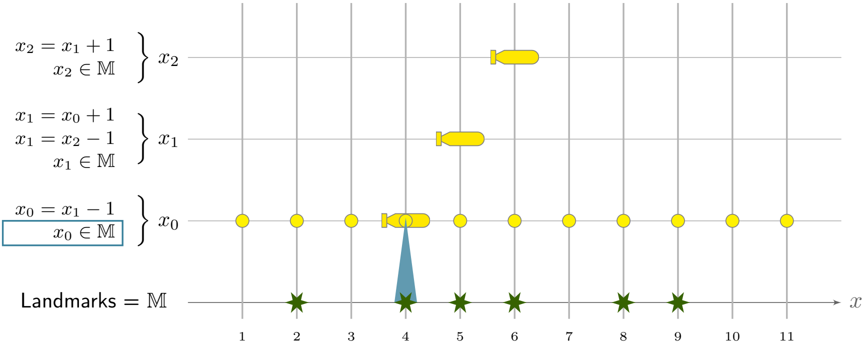
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



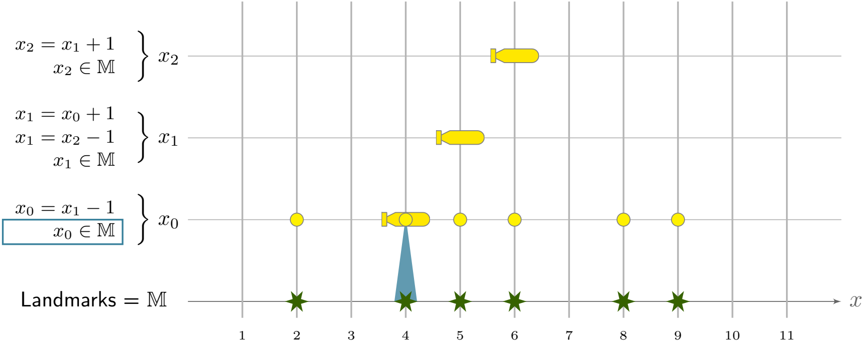
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



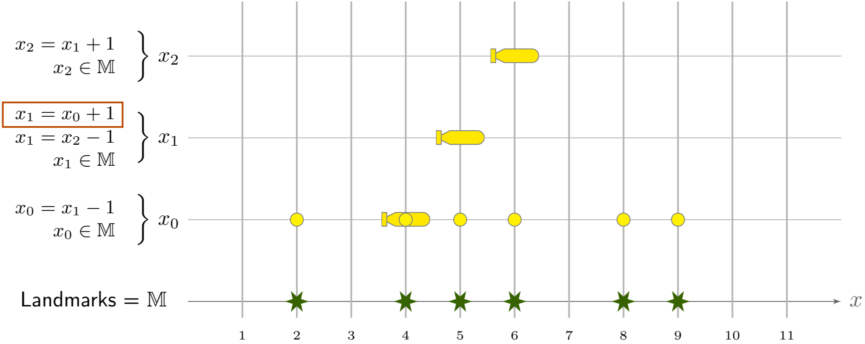
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



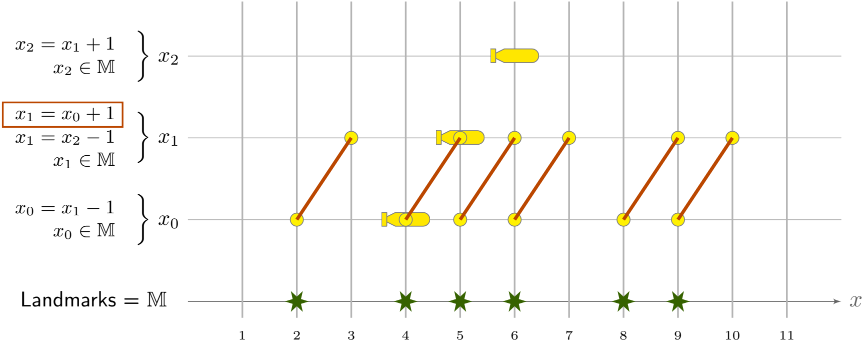
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



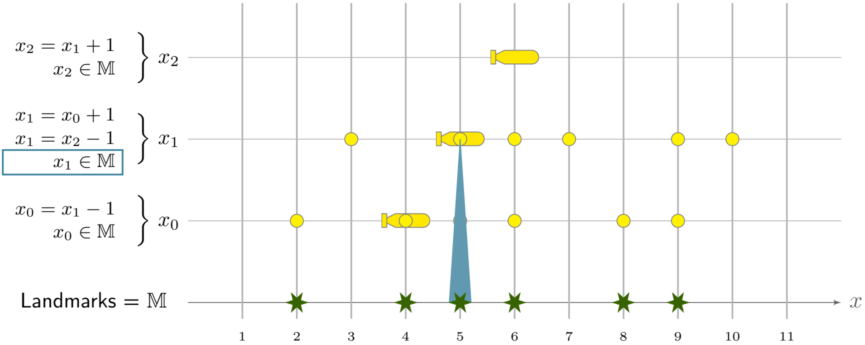
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



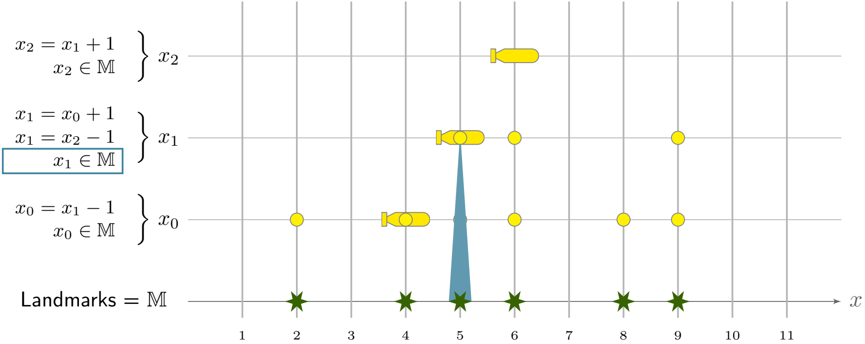
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



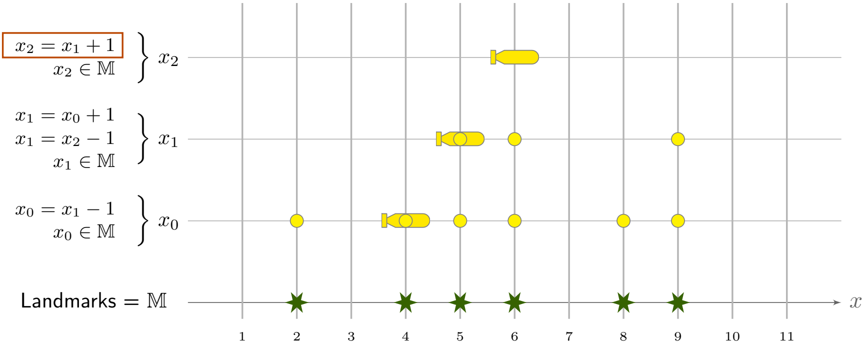
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



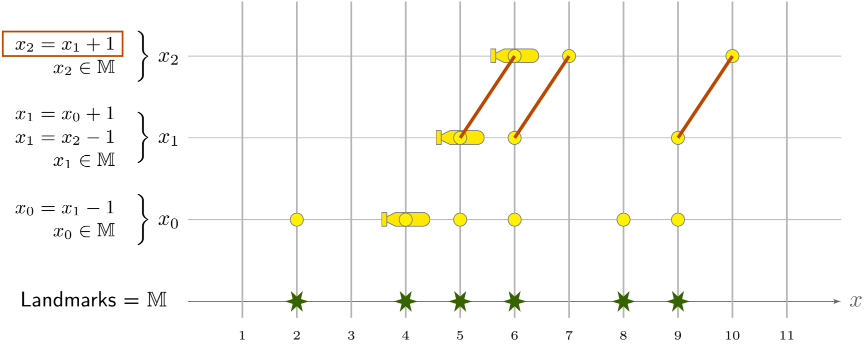
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



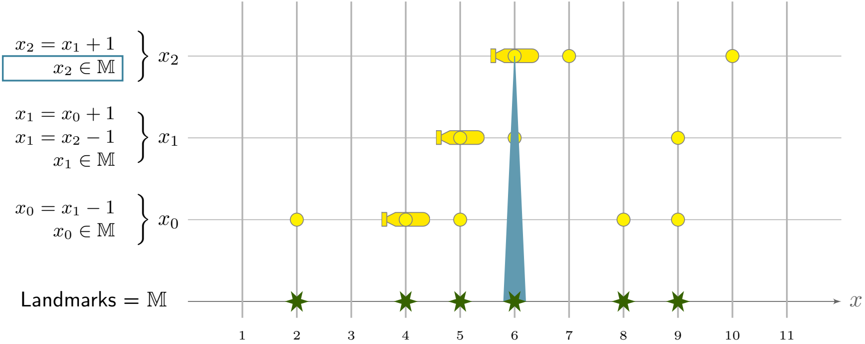
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



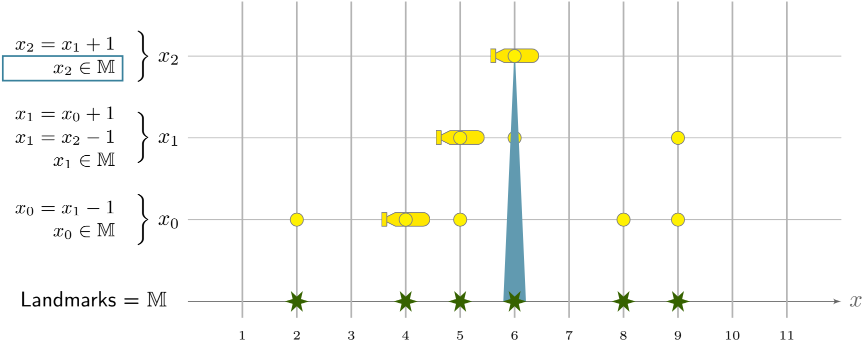
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



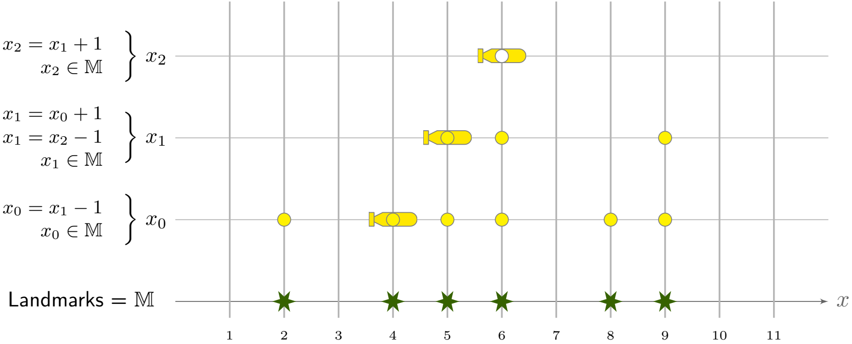
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



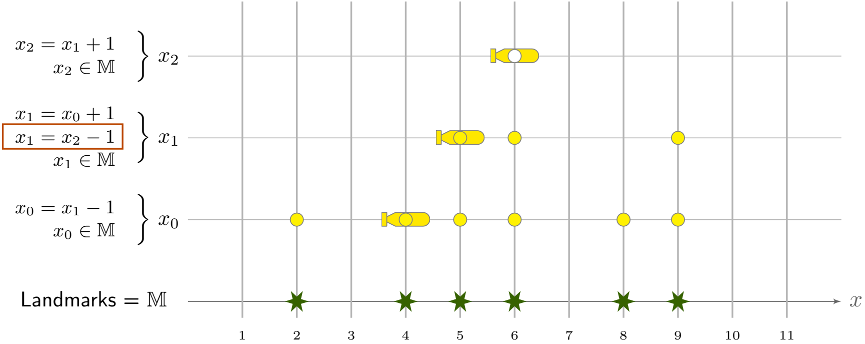
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



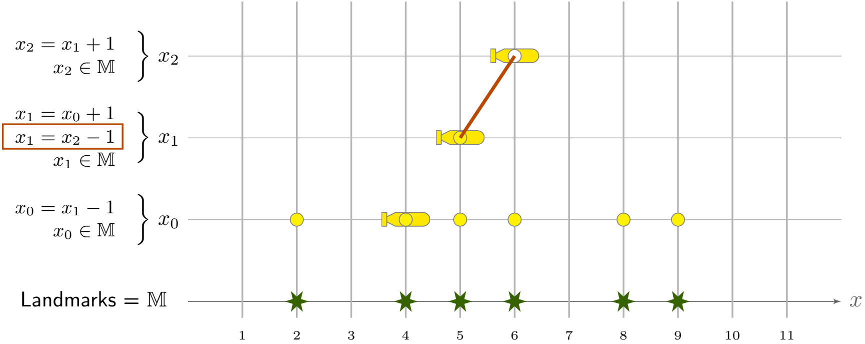
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



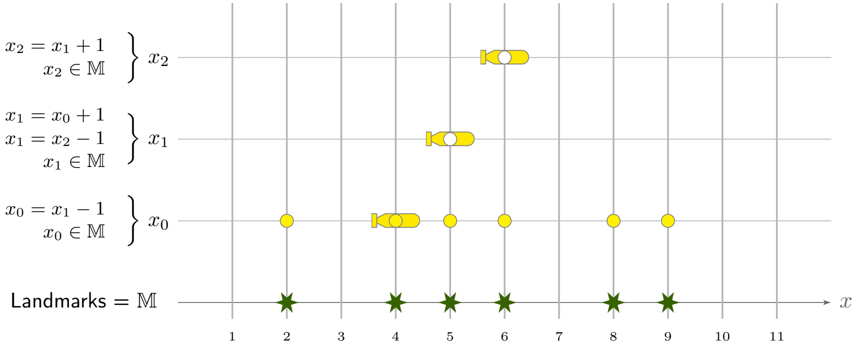
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



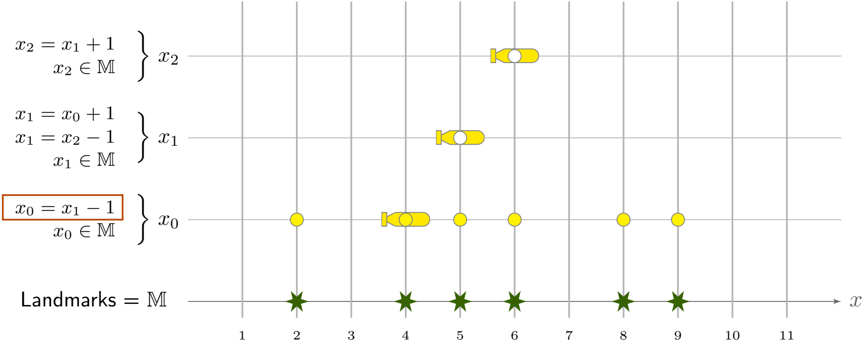
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



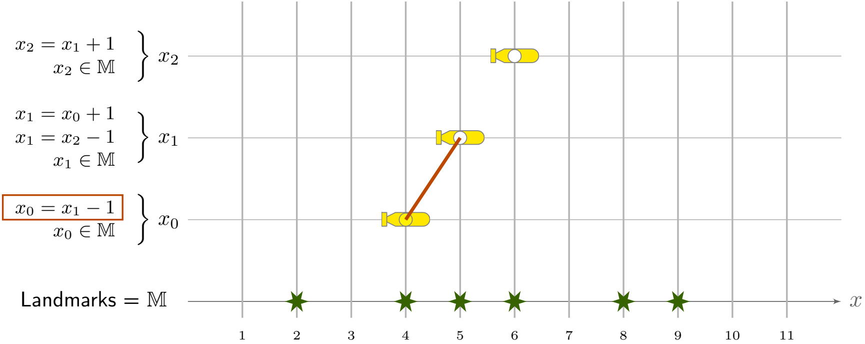
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



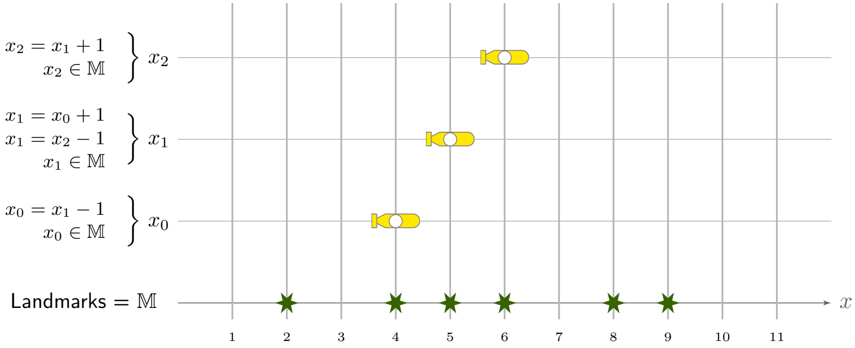
$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

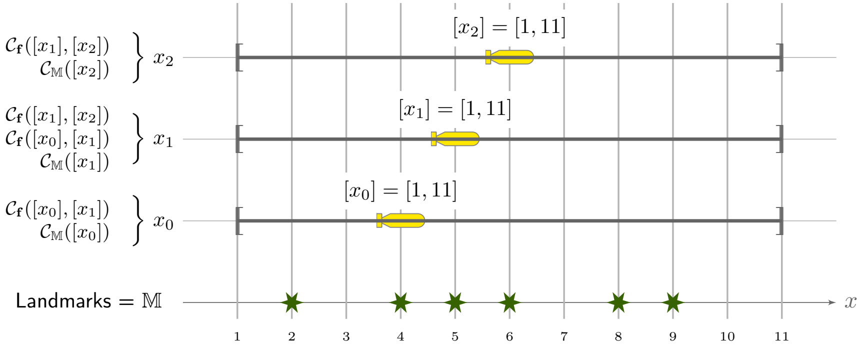
State estimation with indistinguishable landmarks



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks

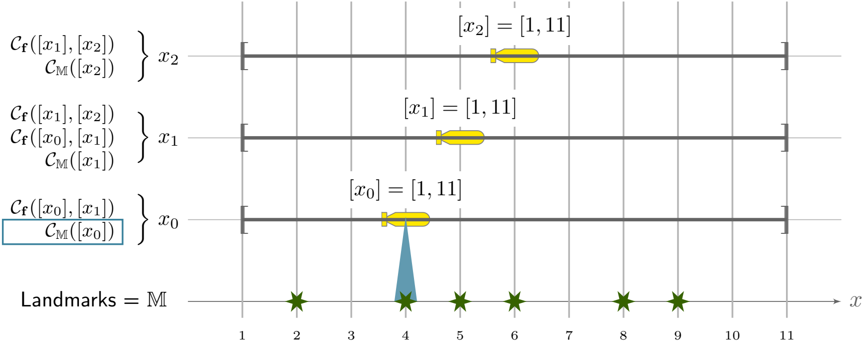
Set-membership approach



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks

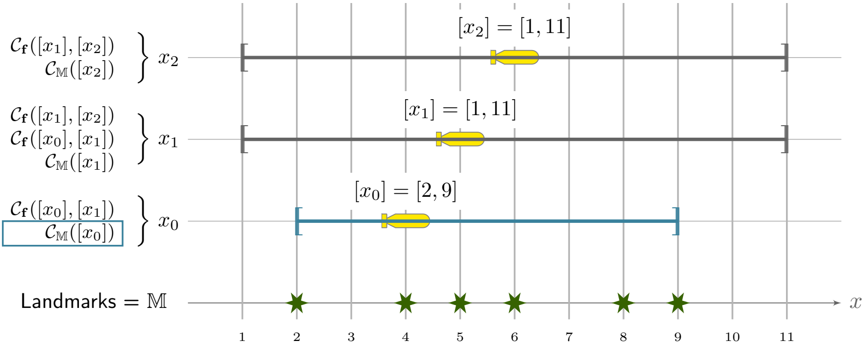
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

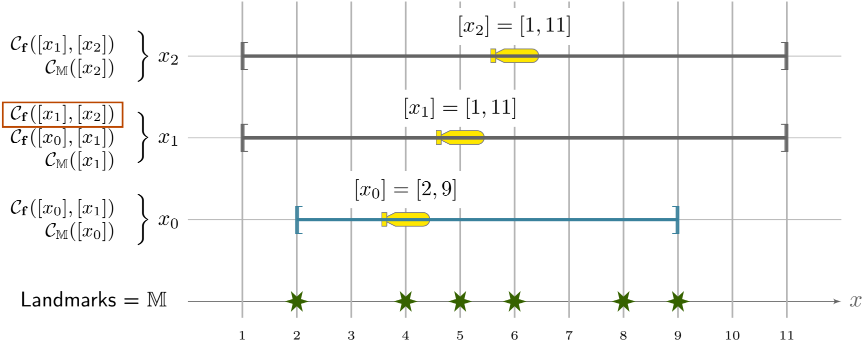
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

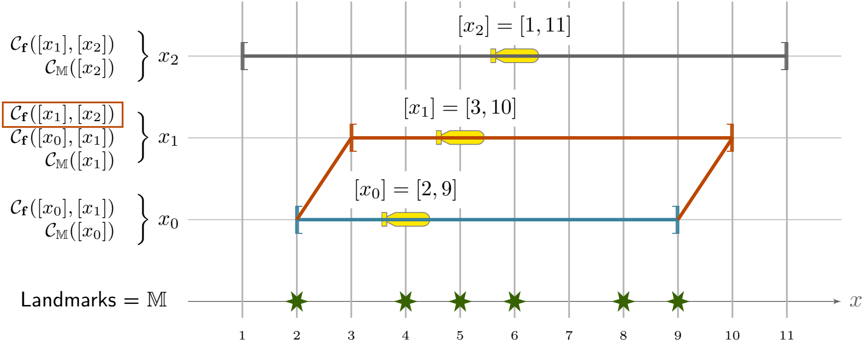
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

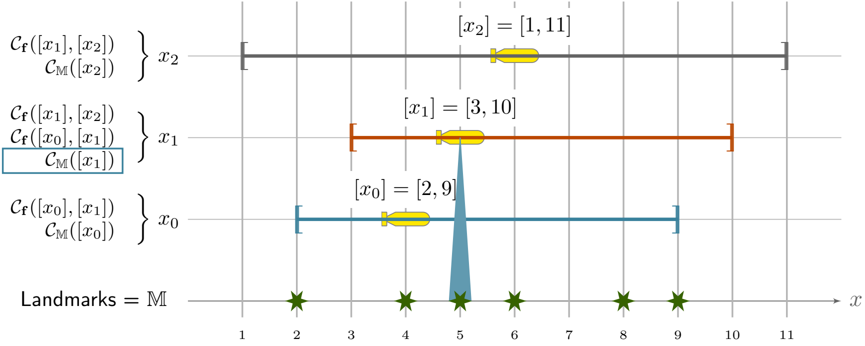
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

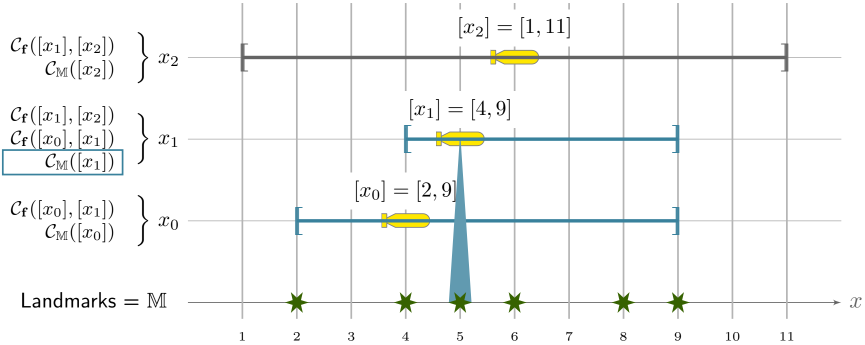
Set-membership approach



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks

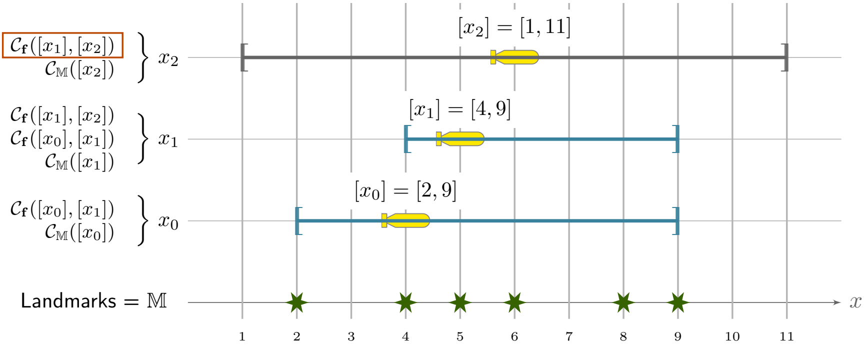
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

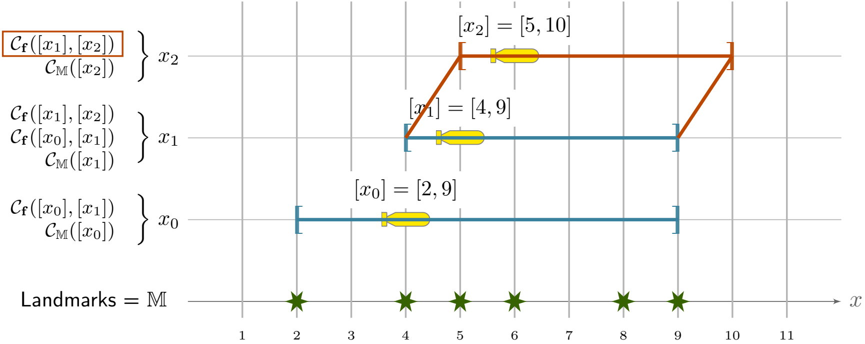
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

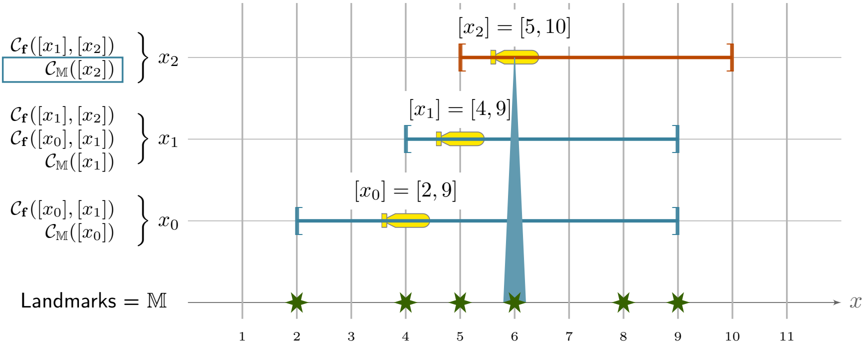
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

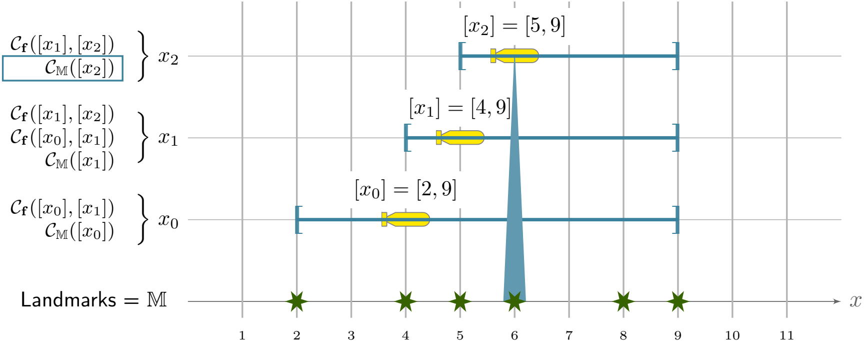
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

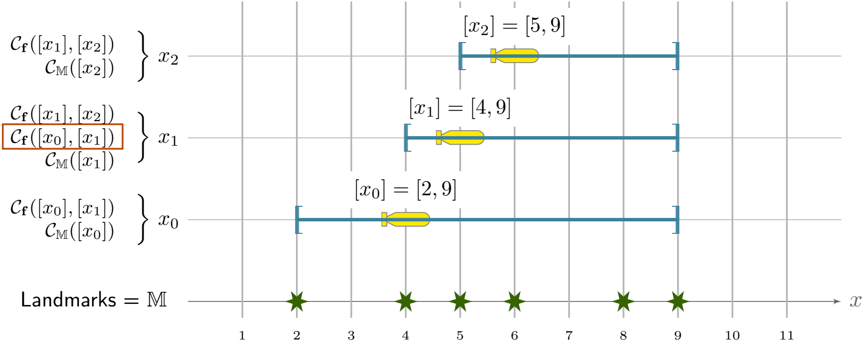
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

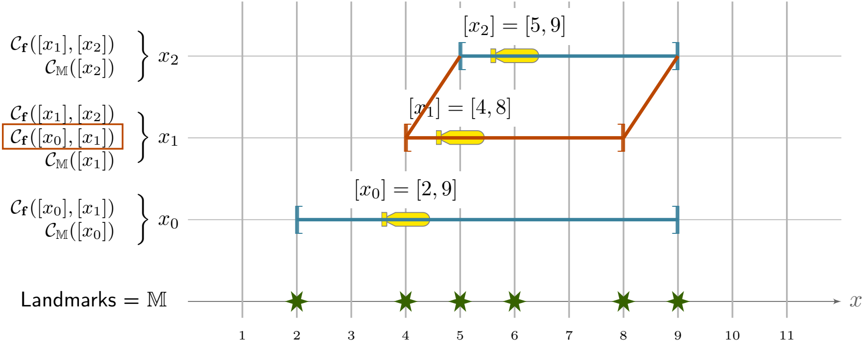
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

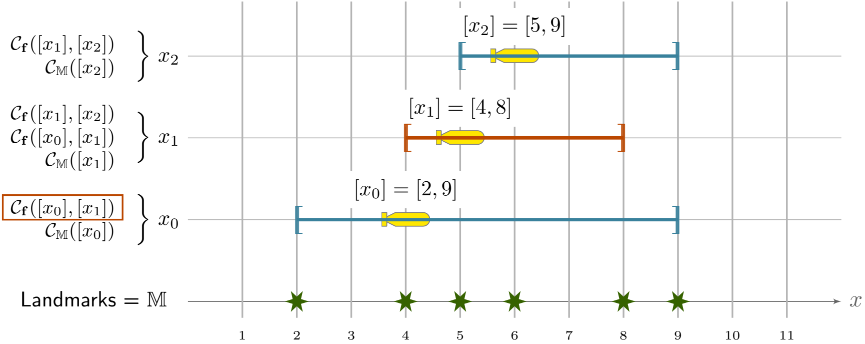
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

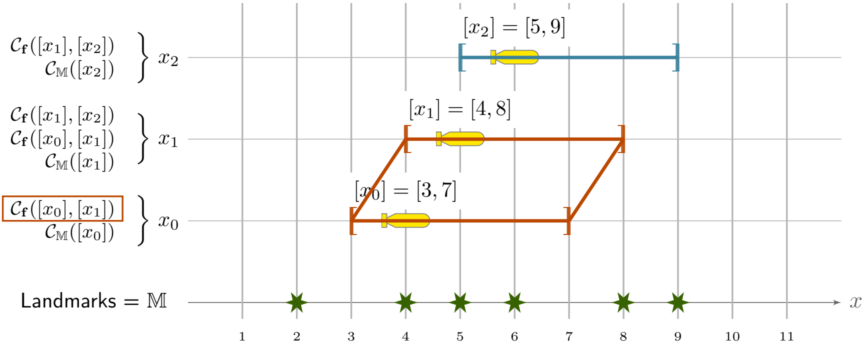
Set-membership approach



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks

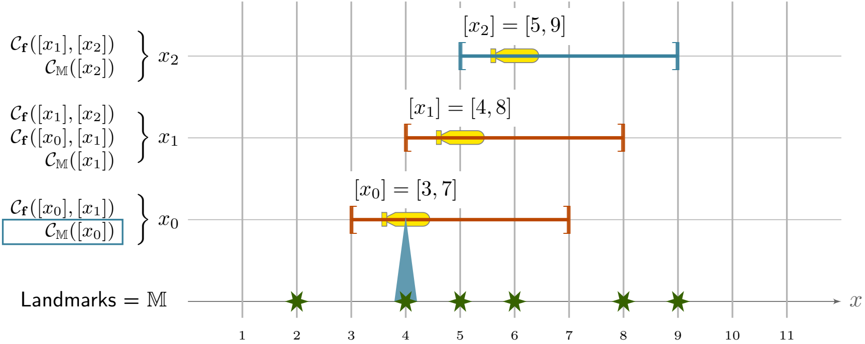
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

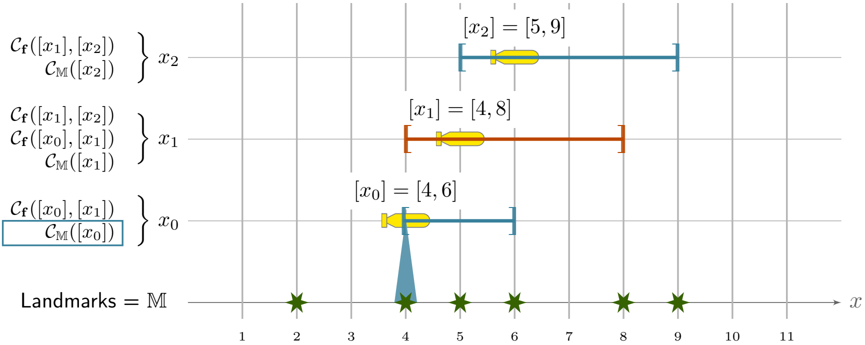
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

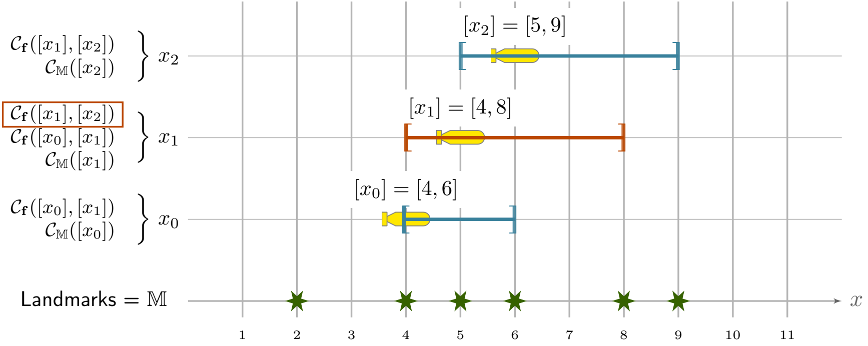
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

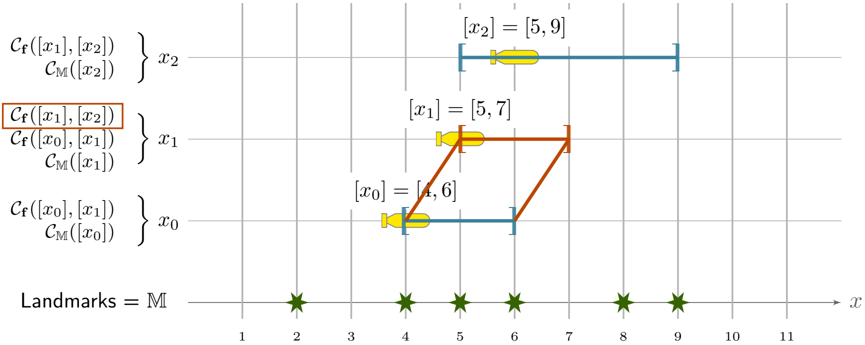
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

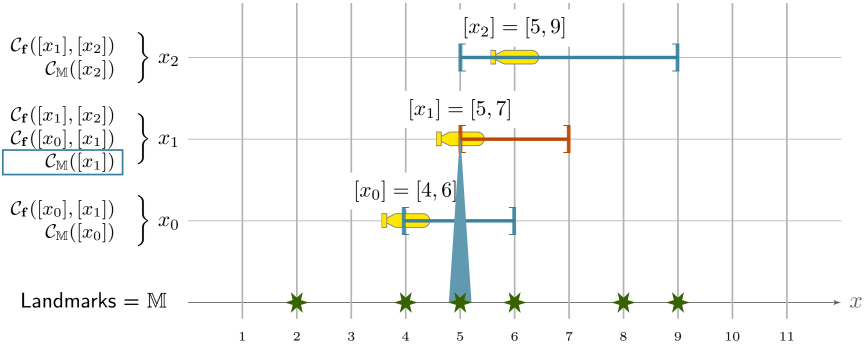
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

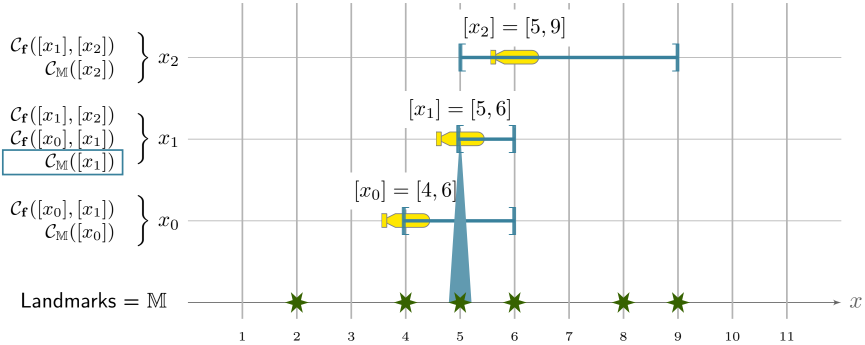
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

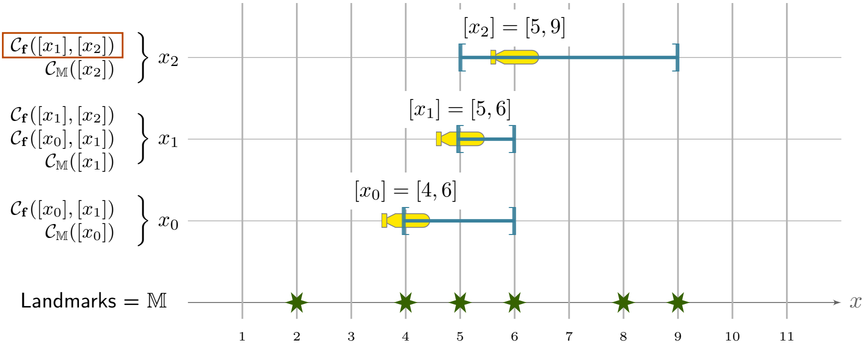
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

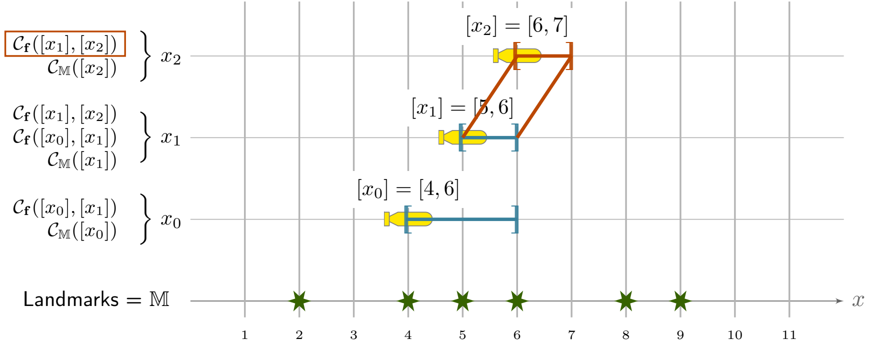
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

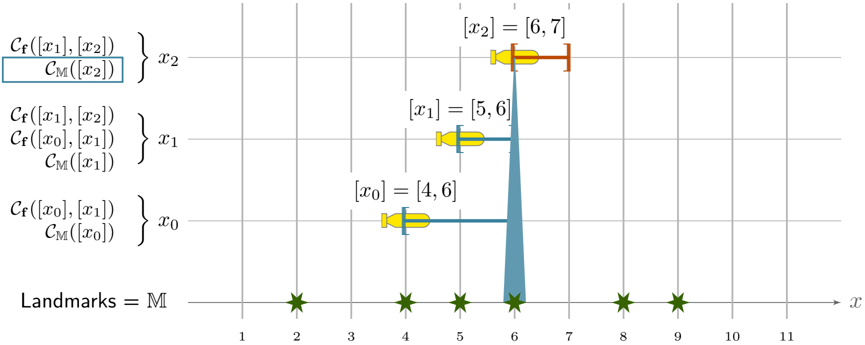
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

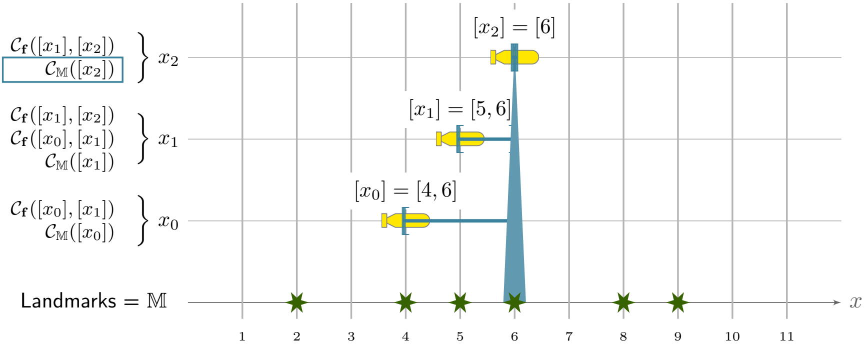
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

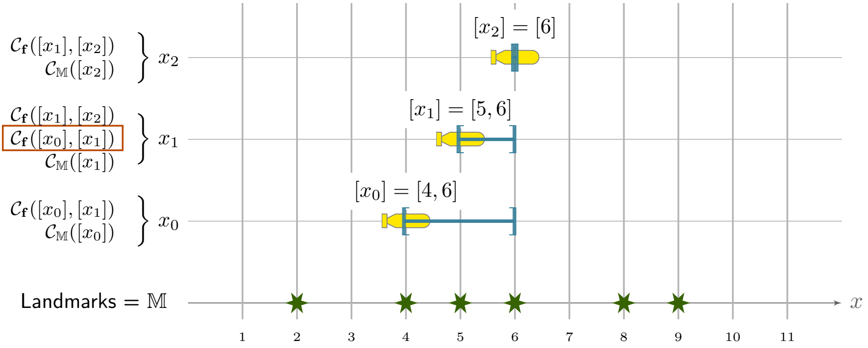
Set-membership approach



$$\mathbb{M} = \{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

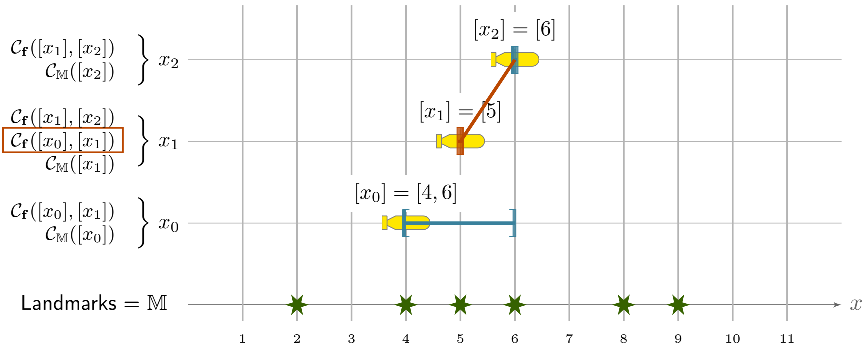
Set-membership approach



$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with indistinguishable landmarks

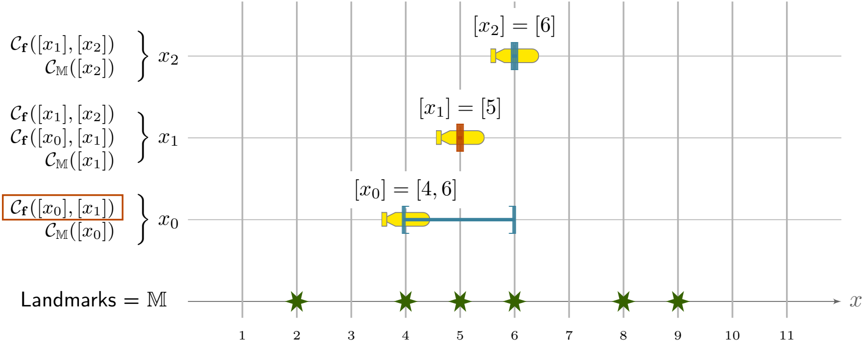
Set-membership approach



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks

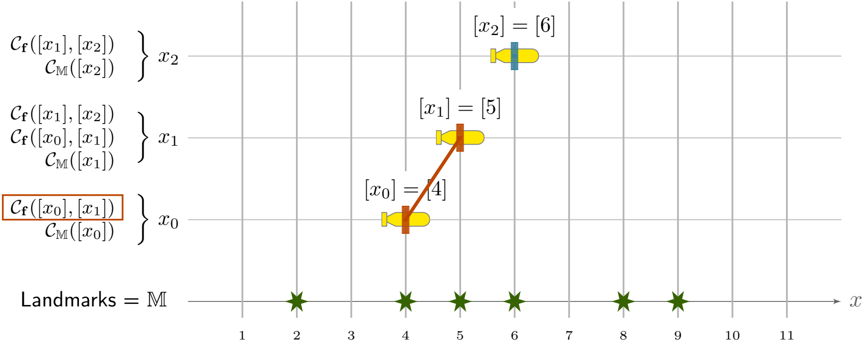
Set-membership approach



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks

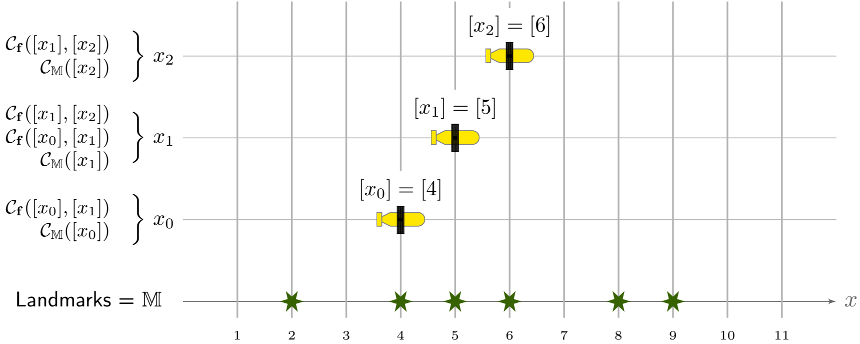
Set-membership approach



$$\mathbb{M} = \{2, 4, 5, 6, 8, 9\}$$

State estimation with indistinguishable landmarks

Set-membership approach

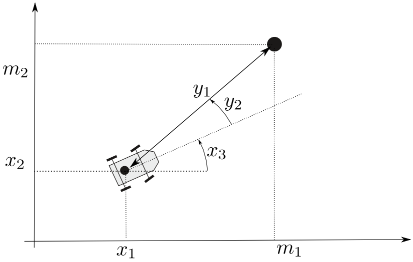


$$\mathbb{M}=\{2,4,5,6,8,9\}$$

State estimation with landmark perception: example

Example:

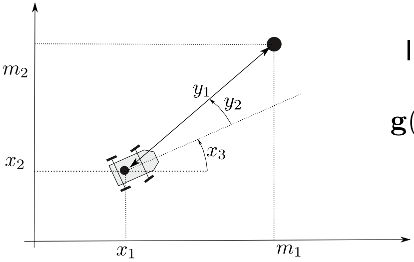
- a robot at position $(x_1, x_2)^\intercal$ with a heading x_3
- a landmark $\mathbf{m}$ located at $(m_1, m_2)^\intercal$
- the corresponding measurement vector is composed of
  - the distance y_1
  - the bearing y_2



State estimation with landmark perception: example

Example:

- a robot at position $(x_1, x_2)^\intercal$ with a heading x_3
- a landmark $\mathbf{m}$ located at $(m_1, m_2)^\intercal$
- the corresponding measurement vector is composed of
  - the distance y_1
  - the bearing y_2



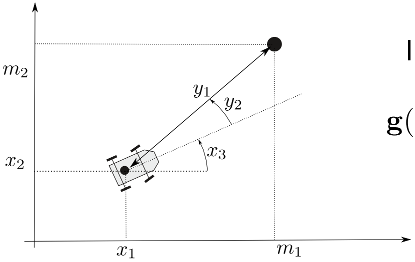
In such case, we have:

$$\mathbf{g}(\mathbf{x}, \mathbf{y}) = \begin{pmatrix} x_1 + y_1 \cdot \cos(x_3 + y_2) - m_1 \\ x_2 + y_1 \cdot \sin(x_3 + y_2) - m_2 \end{pmatrix}$$

State estimation with landmark perception: example

Example:

- a robot at position $(x_1, x_2)^\intercal$ with a heading x_3
- a landmark $\mathbf{m}$ located at $(m_1, m_2)^\intercal$
- the corresponding measurement vector is composed of
  - the distance y_1
  - the bearing y_2



In such case, we have:

$$\mathbf{g}(\mathbf{x}, \mathbf{y}, \mathbf{m}^i) = \begin{pmatrix} x_1 + y_1 \cdot \cos(x_3 + y_2) - m_1^i \\ x_2 + y_1 \cdot \sin(x_3 + y_2) - m_2^i \end{pmatrix}$$

State estimation with landmark perception

In the general case we have:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0} \\ \mathbf{m} \in [\mathbf{m}] \end{cases}$$

State estimation with landmark perception

In the general case we have:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0} \\ \mathbf{m} \in [\mathbf{m}] \end{cases}$$

Problem: when several landmarks $\mathbf{m}_1, \dots, \mathbf{m}_l$ can be observed,

- data may not be associated,
- we do not know to which landmark $\mathbf{m}^i$ the measurement $\mathbf{y}^i$ refers.

State estimation with landmark perception

In the general case we have:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0} \\ \big(\mathbf{m}^i \in [\mathbf{m}_1]\big) \vee \dots \vee \big(\mathbf{m}^i \in [\mathbf{m}_\ell]\big) \end{cases}$$

with $\mathbf{m}^i$ the identity of the beacon perceived at time t_i.

Problem: when several landmarks $\mathbf{m}_1$,...,$\mathbf{m}_l$ can be observed,

- data may not be associated,
- we do not know to which landmark $\mathbf{m}^i$ the measurement $\mathbf{y}^i$ refers.

State estimation with landmark perception

In the general case we have:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0} \\ \mathbf{m}^i \in \mathbb{M} = \{[\mathbf{m}_1], \dots, [\mathbf{m}_\ell]\} \end{cases}$$

with $\mathbf{m}^i$ the identity of the beacon perceived at time t_i.

Problem: when several landmarks $\mathbf{m}_1$,...,$\mathbf{m}_l$ can be observed,

- data may not be associated,
- we do not know to which landmark $\mathbf{m}^i$ the measurement $\mathbf{y}^i$ refers.

State estimation with landmark perception

Interesting test case with **heterogeneous constraints**:

$$\left\{ \begin{array}{ll} \mathbf{m}^i \in \mathbb{M} & \rightarrow \text{discrete constraint} \\ & \\ & \end{array} \right.$$

State estimation with landmark perception

Interesting test case with **heterogeneous constraints**:

$$\begin{cases} \mathbf{m}^i \in \mathbb{M} & \rightarrow \text{discrete constraint} \\ \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) & \rightarrow \text{differential constraint} \\ \end{cases}$$

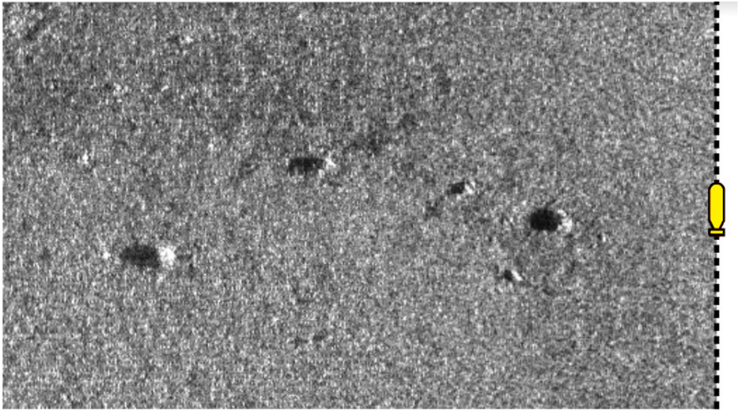
State estimation with landmark perception

Interesting test case with **heterogeneous constraints**:

$$\begin{cases} \mathbf{m}^i \in \mathbb{M} & \rightarrow \text{discrete constraint} \\ \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big) & \rightarrow \text{differential constraint} \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0} & \rightarrow \text{continuous constraint} \end{cases}$$

Constellation contractor: illustration

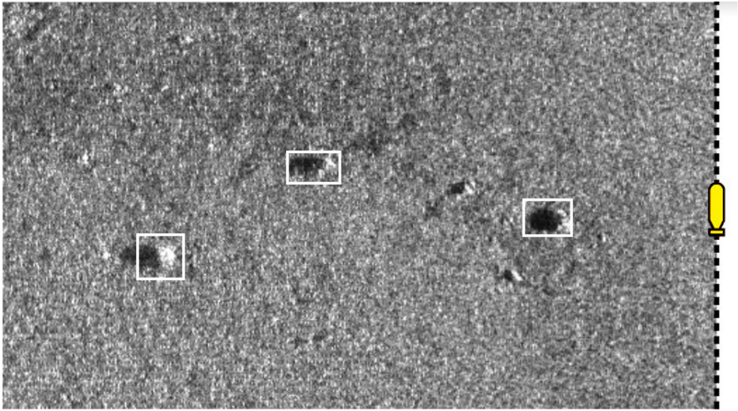
Constraint $\mathbf{m}^i \in \mathbb{M}$:
An observation $\mathbf{y}^i$ is related to one $\mathbf{m}^i$ of the known landmarks $\mathbb{M}$.



Perception of the seabed with a side-scan sonar.

Constellation contractor: illustration

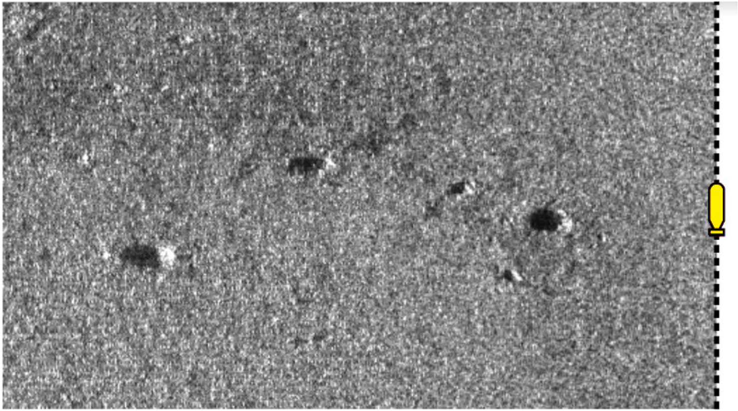
Constraint $\mathbf{m}^i \in \mathbb{M}$:
An observation $\mathbf{y}^i$ is related to one $\mathbf{m}^i$ of the known landmarks $\mathbb{M}$.



Seamarks are already known with some uncertainty.

Constellation contractor: illustration

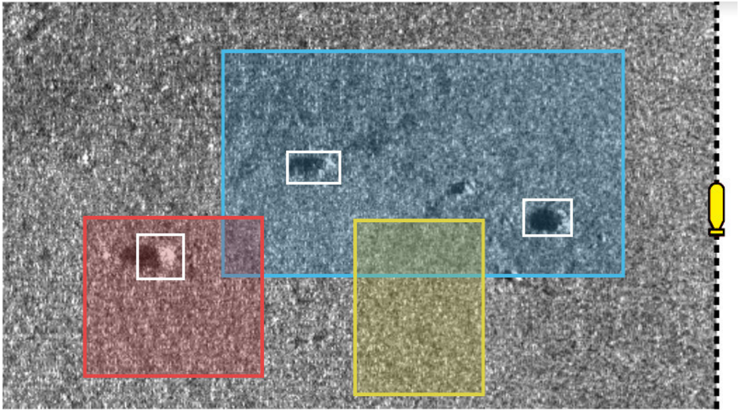
Constraint $\mathbf{m}^i \in \mathbb{M}$:
An observation $\mathbf{y}^i$ is related to one $\mathbf{m}^i$ of the known landmarks $\mathbb{M}$.



Some of the rocks may be observed by the robot with its sonar.

Constellation contractor: illustration

Constraint $\mathbf{m}^i \in \mathbb{M}$:
An observation $\mathbf{y}^i$ is related to one $\mathbf{m}^i$ of the known landmarks $\mathbb{M}$.

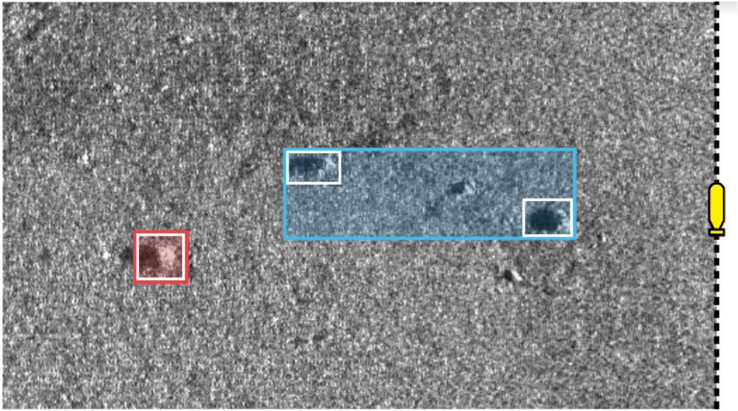


The position of the rock is first estimated from the position estimate.

$$\mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0}$$

Constellation contractor: illustration

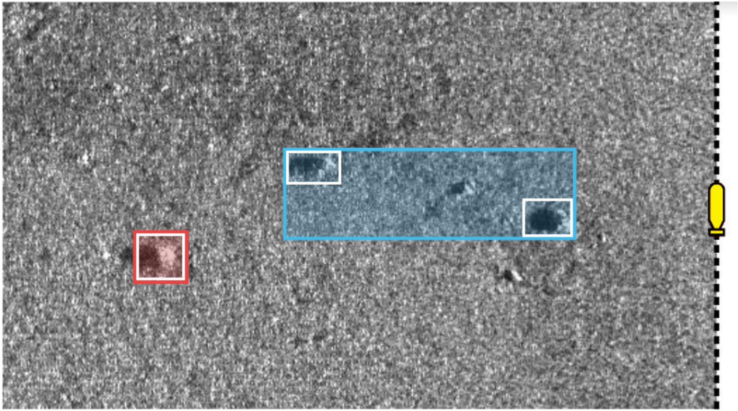
Constraint $\mathbf{m}^i \in \mathbb{M}$:
An observation $\mathbf{y}^i$ is related to one $\mathbf{m}^i$ of the known landmarks $\mathbb{M}$.



Then the position of the rock is contracted from the known map.

Constellation contractor: illustration

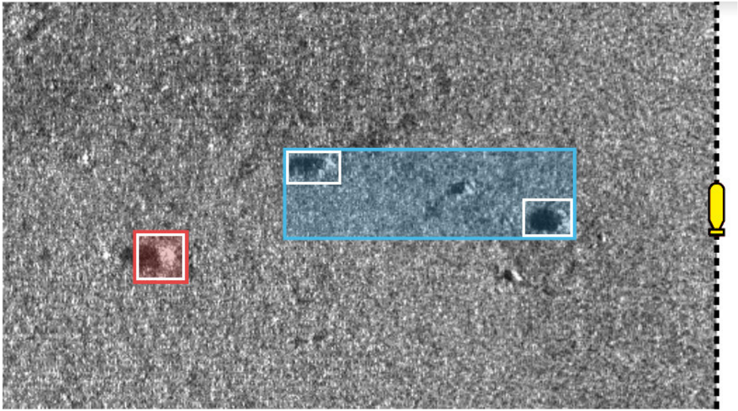
Constraint $\mathbf{m}^i \in \mathbb{M}$:
An observation $\mathbf{y}^i$ is related to one $\mathbf{m}^i$ of the known landmarks $\mathbb{M}$.



If the boxed-position is a singleton, then the rock is *identified*.

Constellation contractor: illustration

Constraint $\mathbf{m}^i \in \mathbb{M}$:
An observation $\mathbf{y}^i$ is related to one $\mathbf{m}^i$ of the known landmarks $\mathbb{M}$.



In any cases, the boxed-positions of the rocks allow localization updates.

$$\mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0}$$

Association constraint: constellation contractor

Let us consider a constellation of ℓ points $\mathbb{M} = \{[\mathbf{m}_1], \dots, [\mathbf{m}_\ell]\}$ of $\mathbb{IR}^d$ and a box $[\mathbf{x}] \in \mathbb{IR}^d$. We want to compute the smallest box $\mathcal{C}_{\text{constell}}([\mathbf{x}])$ containing $\mathbb{M} \cap [\mathbf{x}]$, or equivalently:

$$\mathcal{C}_{\text{constell}}([\mathbf{x}]) = \bigsqcup_{j=1}^{\ell} \big([\mathbf{x}] \cap [\mathbf{m}_j]\big), \tag{2}$$

where $\bigsqcup$, called *squared union*, returns the smallest box enclosing the union of its arguments.

Decomposition

We recall the problem:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big), \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0}, \\ \mathbf{m}^i \in \mathbb{M}, \end{cases} \tag{3}$$

Decomposition

We recall the problem:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big), \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0}, \\ \mathbf{m}^i \in \mathbb{M}, \end{cases} \tag{3}$$

with:

$$\mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \begin{pmatrix} x_1(t_i) \\ x_2(t_i) \end{pmatrix} + y_1^i \cdot \begin{pmatrix} \cos(x_3(t_i) + y_2^i) \\ \sin(x_3(t_i) + y_2^i) \end{pmatrix} - \begin{pmatrix} m_1^i \\ m_2^i \end{pmatrix}$$

Decomposition

We recall the problem:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big), \\ \mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \mathbf{0}, \\ \mathbf{m}^i \in \mathbb{M}, \end{cases} \tag{3}$$

with:

$$\mathbf{g}\big(\mathbf{x}(t_i), \mathbf{y}^i, \mathbf{m}^i\big) = \begin{pmatrix} x_1(t_i) \\ x_2(t_i) \end{pmatrix} + y_1^i \cdot \begin{pmatrix} \cos(x_3(t_i) + y_2^i) \\ \sin(x_3(t_i) + y_2^i) \end{pmatrix} - \begin{pmatrix} m_1^i \\ m_2^i \end{pmatrix}$$

These equations can be broken down into:

$$\left\{ \begin{array}{rl} (i) & \mathbf{v}(\cdot) = \mathbf{f}\big(\mathbf{x}(\cdot), \mathbf{u}(\cdot)\big) \\ (ii) & \dot{\mathbf{x}}(\cdot) = \mathbf{v}(\cdot) \\ (iii) & \mathbf{p}^i = \mathbf{x}(t_i) \\ (iv) & \mathbf{d}^i = \mathbf{m}^i - \mathbf{p}^i_{1,2} \\ (v) & a^i = p_3^i + y_2^i \\ (vi) & \mathbf{d}^i = y_1^i \cdot \begin{pmatrix} \cos(a^i) \\ \sin(a^i) \end{pmatrix} \\ (vii) & \mathbf{m}^i \in \mathbb{M} \end{array} \right. \tag{4}$$

Applying contractors

$$\begin{cases} (i) & \mathbf{v}(\cdot) = \mathbf{f}\big(\mathbf{x}(\cdot), \mathbf{u}(\cdot)\big) \\ (ii) & \dot{\mathbf{x}}(\cdot) = \mathbf{v}(\cdot) \\ (iii) & \mathbf{p}^i = \mathbf{x}(t_i) \\ (iv) & \mathbf{d}^i = \mathbf{m}^i - \mathbf{p}^i_{1,2} \\ (v) & a^i = p^i_3 + y^i_2 \\ (vi) & \mathbf{d}^i = y^i_1 \cdot \begin{pmatrix} \cos(a^i) \\ \sin(a^i) \end{pmatrix} \\ (vii) & \mathbf{m}^i \in \mathbb{M} \end{cases} \implies \begin{cases} (i) & \mathcal{C}_{\mathbf{f}}\big([\mathbf{v}](\cdot), [\mathbf{x}](\cdot), [\mathbf{u}](\cdot)\big) \\ (ii) & \mathcal{C}_{\frac{d}{dt}}\big([\mathbf{x}](\cdot), [\mathbf{v}](\cdot)\big) \\ (iii) & \mathcal{C}_{\text{eval}}\big([t_i], [\mathbf{p}^i], [\mathbf{x}](\cdot)\big) \\ (iv) & \mathcal{C}_{-}\big([\mathbf{d}^i], [\mathbf{m}^i], [\mathbf{p}^i_{1,2}]\big) \\ (v) & \mathcal{C}_{+}\big([a^i], [p^i_3], [y^i_2]\big) \\ (vi) & \mathcal{C}_{\text{polar}}\big([d^i_1], [d^i_2], [y^i_1], [a^i]\big) \\ (vii) & \mathcal{C}_{\text{constell}}\big([\mathbf{m}^i]\big) \end{cases}$$

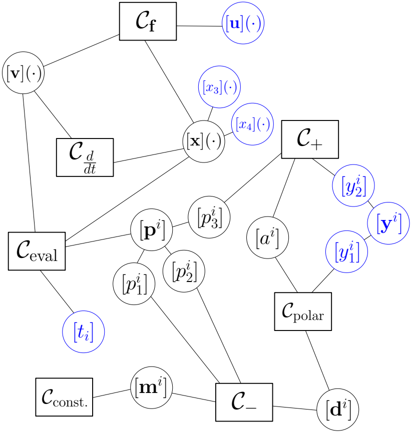
Applying contractors

$$\left\{\begin{array}{rl} (i) & \mathbf{v}(\cdot) = \mathbf{f}\big(\mathbf{x}(\cdot), \mathbf{u}(\cdot)\big) \\ (ii) & \dot{\mathbf{x}}(\cdot) = \mathbf{v}(\cdot) \\ (iii) & \mathbf{p}^i = \mathbf{x}(t_i) \\ (iv) & \mathbf{d}^i = \mathbf{m}^i - \mathbf{p}^i_{1,2} \\ (v) & a^i = p^i_3 + y^i_2 \\ (vi) & \mathbf{d}^i = y^i_1 \cdot \begin{pmatrix} \cos(a^i) \\ \sin(a^i) \end{pmatrix} \\ (vii) & \mathbf{m}^i \in \mathbb{M} \end{array}\right. \implies \left\{\begin{array}{rl} (i) & \mathcal{C}_{\mathbf{f}}\big([\mathbf{v}](\cdot), [\mathbf{x}](\cdot), [\mathbf{u}](\cdot)\big) \\ (ii) & \mathcal{C}_{\frac{d}{dt}}\big([\mathbf{x}](\cdot), [\mathbf{v}](\cdot)\big) \\ (iii) & \mathcal{C}_{\text{eval}}\big([t_i], [\mathbf{p}^i], [\mathbf{x}](\cdot)\big) \\ (iv) & \mathcal{C}_{-}\big([\mathbf{d}^i], [\mathbf{m}^i], [\mathbf{p}^i_{1,2}]\big) \\ (v) & \mathcal{C}_{+}\big([a^i], [p^i_3], [y^i_2]\big) \\ (vi) & \mathcal{C}_{\text{polar}}\big([d^i_1], [d^i_2], [y^i_1], [a^i]\big) \\ (vii) & \mathcal{C}_{\text{constell}}\big([\mathbf{m}^i]\big) \end{array}\right.$$

A Minimal contractor for the Polar equation: application to robot localization
Desrochers, Jaulin. *Engineering Applications of Artificial Intelligence, 55(Supplement C):83–92*, 2016

Reliable non-linear state estimation involving time uncertainties
Rohou, Jaulin, Mihaylova, Le Bars, Veres. *Automatica, 93:379–388*, 2018

Applying contractors with Codac



(inputs in blue)

Python code of the solver:

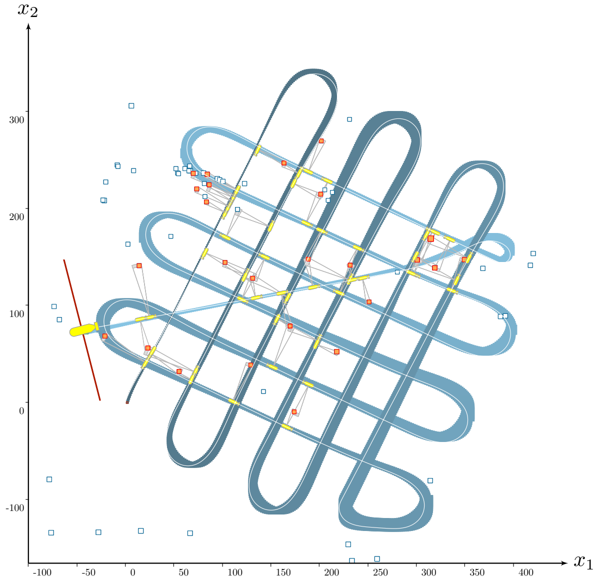
```
cn = ContractorNetwork()
cn.add(ctc_f, [x,u,v])
cn.add(ctc.deriv, [x,v])

for i in range(0,len(v_obs)):

  t  = Interval(v_obs[i][0])
  y1 = Interval(v_obs[i][1])
  y2 = Interval(v_obs[i][2])

  a = cn.create_dom(Interval())
  d = cn.create_dom(IntervalVector(2))
  p = cn.create_dom(IntervalVector(3))

  cn.add(ctc_constell, [m[i]])
  cn.add(ctc_minus, [d,m[i],p[0],p[1]])
  cn.add(ctc_plus, [a,p[2],y2])
  cn.add(ctc.polar, [d,y1,a])
  cn.add(ctc.eval, [t,p,x,v])

cn.contract(True)
```

Application

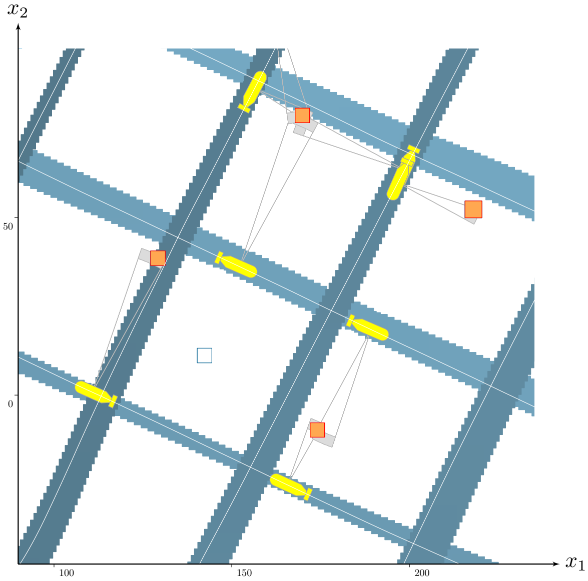
- Daurade: Autonomous Underwater Vehicle (AUV)
- weight: 1010kg – length: 5m – max depth: 300m

Special thanks to DGA-TN Brest (formerly GESMA)

Application



Application



Results on actual data

Map: 133 objects. 54 detections in sonar images.

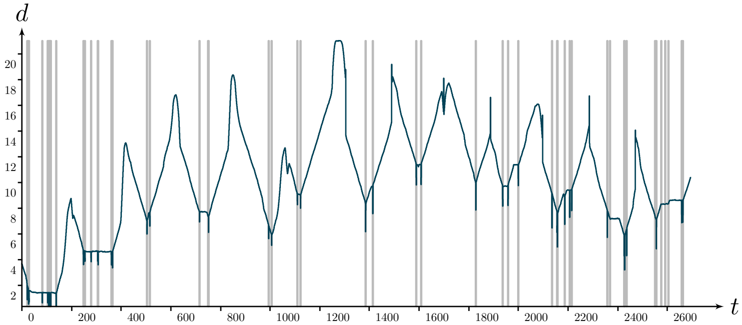
Table: Numerical results of the iterative localization algorithm.

#	time(s)	#min	#max	#ok
1	0.278	133	133	0
2	0.271	14	64	0
3	0.268	5	52	0
4	0.266	1	34	2
5	0.271	1	16	39
6	0.267	1	4	48
7	0.266	1	3	49
8	0.266	1	3	50
9	0.266	1	2	51

- #min: minimal number of objects included in the $[\mathbf{m}^i]$
- #max: maximal number of objects included in the $[\mathbf{m}^i]$
- #ok: number of correct associations

Results on actual data

The initial position of the robot is **not known before the contractions**, and is finally estimated with an **error of 3.6m** in the worst case:



$d = w([\mathbf{x}_{1,2}])$ when reaching a contracting fixed point. Computation time < 2.5s.

d: diameter of each box $[x_1](t) \times [x_2](t)$, *i.e.* localization error in the very worst case.

Assets of constraint programming

- **simplicity** of the approach
  transparent application of contractors on elementary constraints

Assets of constraint programming

- **simplicity** of the approach
  transparent application of contractors on elementary constraints
- **reliability** of the results: no solution can be lost
  useful for proof purposes and the safety of systems

Assets of constraint programming

- **simplicity** of the approach
  transparent application of contractors on elementary constraints
- **reliability** of the results: no solution can be lost
  useful for proof purposes and the safety of systems
- focus on **the *what* instead of the *how***
  no expertise required on how to solve a problem

Assets of constraint programming

- **simplicity** of the approach
  transparent application of contractors on elementary constraints
- **reliability** of the results: no solution can be lost
  useful for proof purposes and the safety of systems
- focus on **the *what* instead of the *how***
  no expertise required on how to solve a problem
- **complex systems** easily handled
  non-linearities, differential equations, values from datasets

Assets of constraint programming

- **simplicity** of the approach
  transparent application of contractors on elementary constraints
- **reliability** of the results: no solution can be lost
  useful for proof purposes and the safety of systems
- focus on **the *what* instead of the *how***
  no expertise required on how to solve a problem
- **complex systems** easily handled
  non-linearities, differential equations, values from datasets

Codac library: open-source library providing tools for constraint programming over reals, trajectories and sets

http://codac.io