

Solving geometrical constraints in space-time

L. Jaulin

First Robex day
2021, November 24



1. Secure a zone

INFO OBS. Un sous-marin nucléaire russe repéré dans le Golfe de Gascogne



Le navire a été repéré en janvier. Ce serait la première fois depuis la fin de la Guerre Froide qu'un tel sous-marin, doté de missiles nucléaires, se serait aventuré dans cette zone au large des côtes françaises.



Bay of Biscay 220 000 km²



An intruder

Several robots $\mathcal{R}_1, \dots, \mathcal{R}_n$ at positions $\mathbf{a}_1, \dots, \mathbf{a}_n$ are moving in the ocean.

If the intruder is in the visibility zone of one robot, it is detected.[5]

Complementary approach

We assume that a virtual intruder exists inside $\mathbb{G}$.

We localize it with a set-membership observer inside $\mathbb{X}(t)$.

The secure zone corresponds to the complementary of $\mathbb{X}(t)$.

Assumptions

The intruder satisfies

$$\dot{\mathbf{x}} \in \mathbb{F}(\mathbf{x}(t)).$$

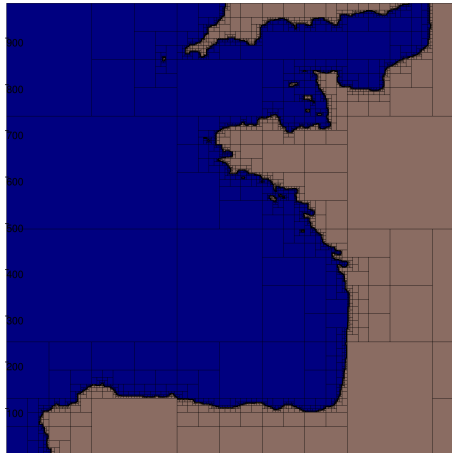
Each robot $\mathcal{R}_i$ has the visibility zone $g_{\mathbf{a}_i}^{-1}([0, d_i])$ where d_i is the scope.

Theorem. An (undetected) intruder has a state vector $\mathbf{x}(t)$ inside the set

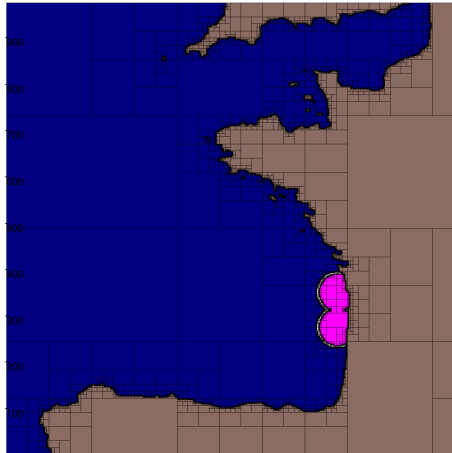
$$\mathbb{X}(t) = \mathbb{G} \cap (\mathbb{X}(t - dt) + dt \cdot \mathbb{F}(\mathbb{X}(t - dt))) \cap \bigcap_i g_{\mathbf{a}_i(t)}^{-1}([d_i(t), \infty]),$$

where $\mathbb{X}(0) = \mathbb{G}$. The secure zone is

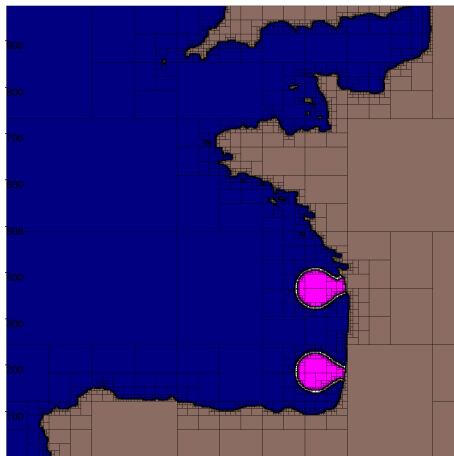
$$\mathbb{S}(t) = \overline{\mathbb{X}(t)}.$$



Set $\mathbb{G}$ in blue

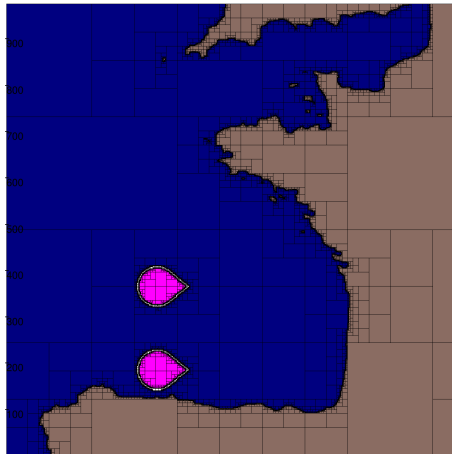


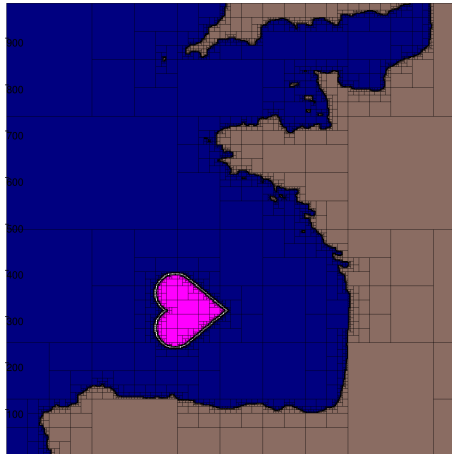
Magenta: $\mathbb{G} \cap \bigcup_i g_{\mathbf{a}_i(t)}^{-1}([0, d_i(t)])$ Blue: $\mathbb{G} \cap \bigcap_i g_{\mathbf{a}_i(t)}^{-1}([d_i(t), \infty])$

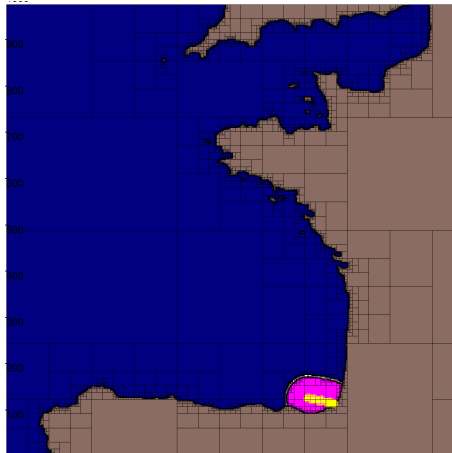


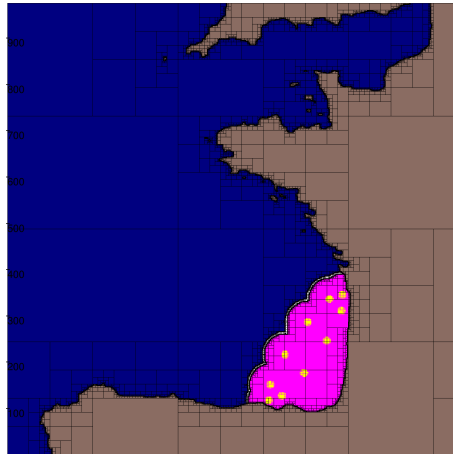
Blue:

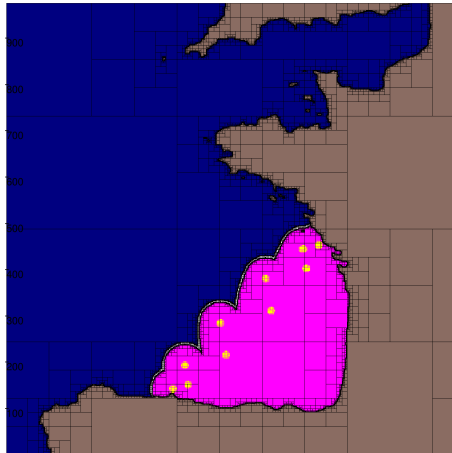
$$\mathbb{X}(t) = \mathbb{G} \cap (\mathbb{X}(t - dt) + dt \cdot \mathbb{F}(\mathbb{X}(t - dt))) \cap \bigcap_i g_{\mathbf{a}_i(t)}^{-1}([d_i(t), \infty]).$$

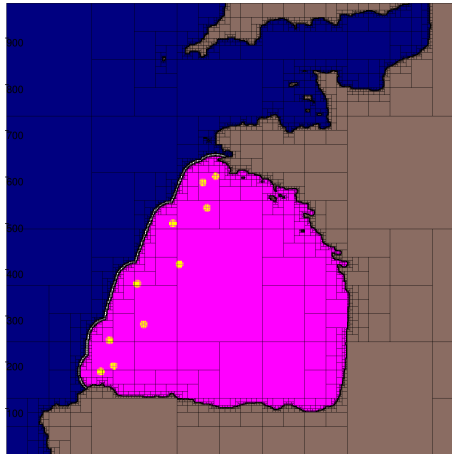












Video : youtube

Idea: Take into account the future.

The feasible set can be obtained by the following contractions

$$\begin{aligned}\overrightarrow{\mathbb{X}}(t) &= \overrightarrow{\mathbb{X}}(t) \cap (\mathbb{X}(t-dt) + dt \cdot \mathbb{F}(\mathbb{X}(t-dt))) \\ \overleftarrow{\mathbb{X}}(t) &= \overleftarrow{\mathbb{X}}(t) \cap (\mathbb{X}(t+dt) - dt \cdot \mathbb{F}(\mathbb{X}(t+dt))) \\ \mathbb{X}(t) &= \overrightarrow{\mathbb{X}}(t) \cap \overleftarrow{\mathbb{X}}(t)\end{aligned}$$

with the initialization

$$\mathbb{X}(t) = \overrightarrow{\mathbb{X}}(t) = \overleftarrow{\mathbb{X}}(t) = \mathbb{G}.$$

2. Intervals

Problem. Given $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a box $[\mathbf{x}] \subset \mathbb{R}^n$, prove that

$$\forall \mathbf{x} \in [\mathbf{x}], f(\mathbf{x}) \geq 0.$$

Interval arithmetic can solve efficiently this problem.

Example. Is the function

$$f(\mathbf{x}) = x_1 x_2 - (x_1 + x_2) \cos x_2 + \sin x_1 \cdot \sin x_2 + 2$$

always positive for $x_1, x_2 \in [-1, 1]$?

Interval arithmetic

$$\begin{aligned} [-1, 3] + [2, 5] &= [1, 8], \\ [-1, 3] \cdot [2, 5] &= [-5, 15], \\ \text{abs}([-7, 1]) &= [0, 7] \end{aligned}$$

The interval extension of

$$f(x_1, x_2) = x_1 \cdot x_2 - (x_1 + x_2) \cdot \cos x_2 + \sin x_1 \cdot \sin x_2 + 2$$

is

$$\begin{aligned} [f]([x_1], [x_2]) &= [x_1] \cdot [x_2] - ([x_1] + [x_2]) \cdot \cos [x_2] \\ &\quad + \sin [x_1] \cdot \sin [x_2] + 2. \end{aligned}$$

Theorem (Moore, 1970)

$$[f]([x]) \subset \mathbb{R}^+ \Rightarrow \forall \mathbf{x} \in [x], f(\mathbf{x}) \geq 0.$$

Contractors

The operator $\mathcal{C} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a *contractor* [2] for the equation $f(\mathbf{x}) = 0$, if

$$\begin{cases} \mathcal{C}([x]) \subset [x] & \text{(contractance)} \\ \mathbf{x} \in [x] \text{ and } f(\mathbf{x}) = 0 \Rightarrow \mathbf{x} \in \mathcal{C}([x]) & \text{(consistence)} \end{cases}$$

Tubes

A trajectory is a function $\mathbf{f} : \mathbb{R} \rightarrow \mathbb{R}^n$. [4, 3]. For instance

$$\mathbf{f}(t) = \begin{pmatrix} \cos t \\ \sin t \end{pmatrix}$$

is a trajectory.

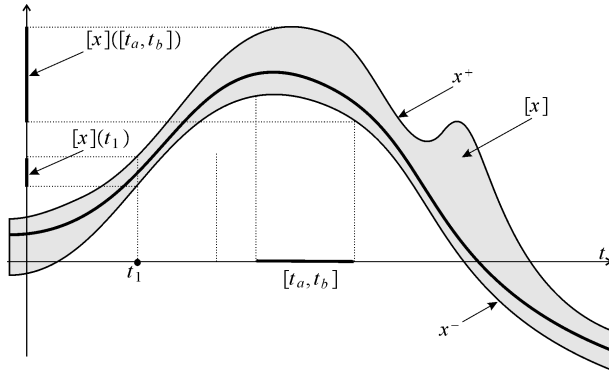
Order relation

$$\mathbf{f} \leq \mathbf{g} \Leftrightarrow \forall t, \forall i, f_i(t) \leq g_i(t).$$

We have

$$\mathbf{h} = \mathbf{f} \wedge \mathbf{g} \Leftrightarrow \forall t, \forall i, h_i(t) = \min(f_i(t), g_i(t)),$$

$$\mathbf{h} = \mathbf{f} \vee \mathbf{g} \Leftrightarrow \forall t, \forall i, h_i(t) = \max(f_i(t), g_i(t)).$$



The set of trajectories is a lattice. Interval of trajectories (tubes) can be defined.

Propagation

Example. Consider $x(t) \in [x](t)$ with the constraint

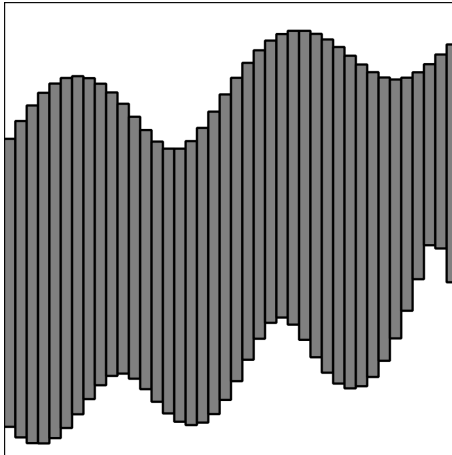
$$\forall t, x(t) = x(t+1)$$

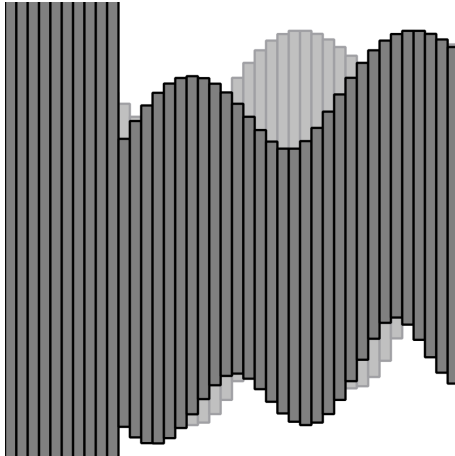
Contract the tube $[x](t)$.

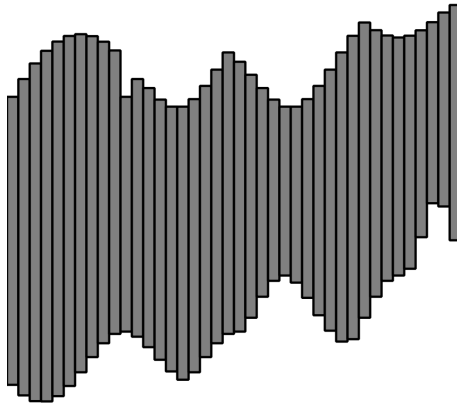
We first decompose into primitive trajectory constraints

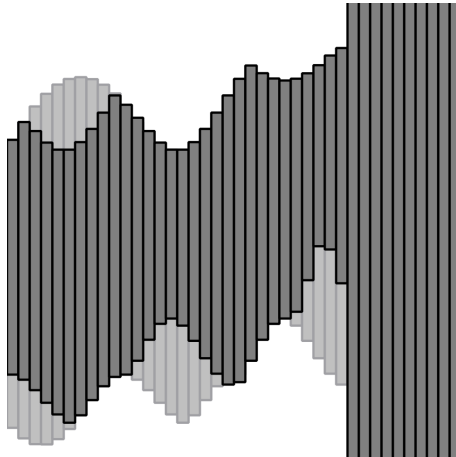
$$x(t) = a(t+1)$$

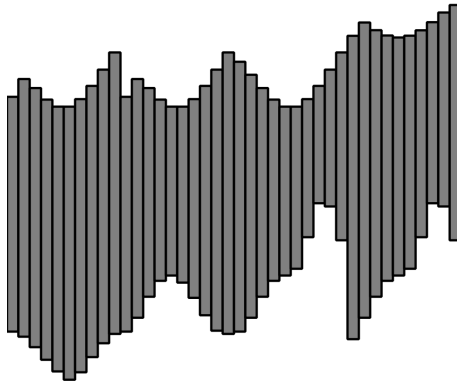
$$x(t) = a(t).$$

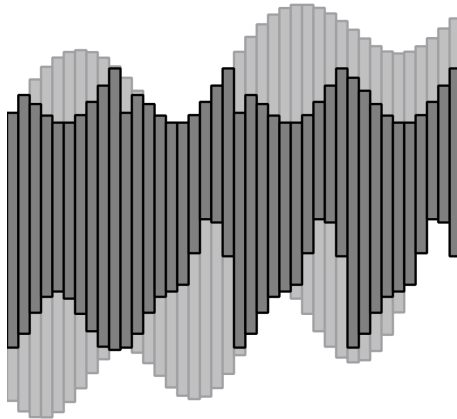


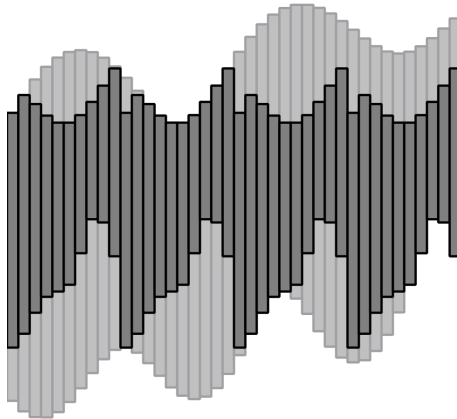












tubex-lib

Search docs

☐ Tubes; basics

Definition

Arithmetics on tubes

Integrals of tubes

Set-inversion

Contractors for tubes

Implementation

Installing the Tubex library

© 1996-2000 by The McGraw-Hill Companies, Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage and retrieval system, without prior written permission from The McGraw-Hill Companies, Inc.

A tube $[x](\cdot)$ is defined as an envelope enclosing an uncertain trajectory $x(\cdot) : \mathbb{R} \rightarrow \mathbb{R}^n$. It is built as an interval of two functions $[x^-(\cdot), x^+(\cdot)]$ such that $\forall t, x^-(t) \leq x^+(t)$. A trajectory $x(\cdot)$ belongs to the tube $[x](\cdot)$ if $\forall t, x(t) \in [x](t)$. Fig. 1 illustrates a tube implemented with a set of boxes. This sliced implementation is detailed hereinafter.

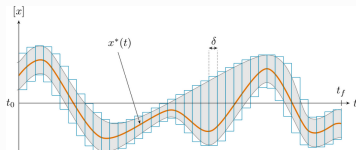


Fig. 1 A tube $[x](\cdot)$ represented by a set of slices. This representation can be used to enclose signals such as $x^*(\cdot)$.

Code example:

```
float timestep = 0.1;
Interval domain(0,10);
Tube x(domain, timestep, Function("t", "{t-5}^2 + [-0.5,0.5]"));
```

<http://codac.io/> [4]

3. Cooperative localization

Classical state estimation

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) & t \in \mathbb{R} \\ \mathbf{0} &= \mathbf{g}(\mathbf{x}(t), t) & t \in \mathbb{T} \subset \mathbb{R}. \end{cases}$$

Space constraint $\mathbf{g}(\mathbf{x}(t), t) = 0$.

Example.

$$\left\{ \begin{array}{l} \dot{x}_1 = x_3 \cos x_4 \\ \dot{x}_2 = x_3 \cos x_4 \\ \dot{x}_3 = u_1 \\ \dot{x}_4 = u_2 \\ (x_1(5) - 1)^2 + (x_2(5) - 2)^2 - 4 = 0 \\ (x_1(7) - 1)^2 + (x_2(7) - 2)^2 - 9 = 0 \end{array} \right.$$

With time-space constraints

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) & t \in \mathbb{R} \\ \mathbf{0} &= \mathbf{g}(\mathbf{x}(t), \mathbf{x}(t'), t, t') & (t, t') \in \mathbb{T} \subset \mathbb{R} \times \mathbb{R}. \end{cases}$$

Example. An ultrasonic underwater robot with state

$$\mathbf{x} = (x_1, x_2, \dots) = (x, y, \theta, v, \dots)$$

At time t the robot emits an omni-directional sound. At time t' it receives it

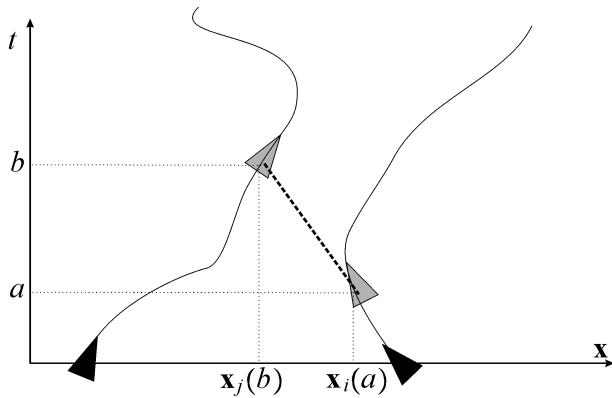
$$\left(x_1 - x'_1\right)^2 + \left(x_2 - x'_2\right)^2 - c \left(t - t'\right)^2 = 0.$$

Consider n robots $\mathcal{R}_1, \dots, \mathcal{R}_n$ described by

$$\dot{\mathbf{x}}_i = \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \mathbf{u}_i \in [\mathbf{u}_i].$$

Omnidirectional sounds are emitted and received.

A *ping* is a 4-uple (a, b, i, j) where a is the emission time, b is the reception time, i is the emitting robot and j the receiver.



With the time space constraint

$$\begin{aligned}\dot{\mathbf{x}}_i &= \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \mathbf{u}_i \in [\mathbf{u}_i]. \\ g(\mathbf{x}_{i(k)}(a(k)), \mathbf{x}_{j(k)}(b(k)), a(k), b(k)) &= 0\end{aligned}$$

where

$$g(\mathbf{x}_i, \mathbf{x}_j, a, b) = \|\mathbf{x}_1 - \mathbf{x}_2\| - c(b - a).$$

Clocks are uncertain. We only have measurements $\tilde{a}(k), \tilde{b}(k)$ of $a(k), b(k)$ thanks to clocks h_i . Thus

$$\dot{\mathbf{x}}_i = \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \mathbf{u}_i \in [\mathbf{u}_i].$$

$$g(\mathbf{x}_{i(k)}(a(k)), \mathbf{x}_{j(k)}(b(k)), a(k), b(k)) = 0$$

$$\tilde{a}(k) = h_{i(k)}(a(k))$$

$$\tilde{b}(k) = h_{j(k)}(b(k))$$

The drift of the clocks is bounded

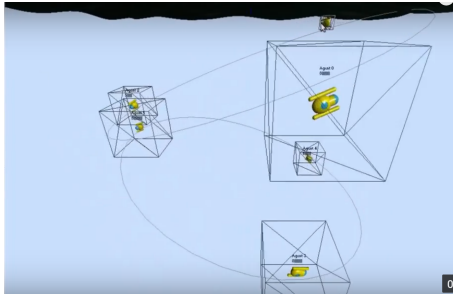
$$\dot{\mathbf{x}}_i = \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \mathbf{u}_i \in [\mathbf{u}_i].$$

$$g(\mathbf{x}_{i(k)}(a(k)), \mathbf{x}_{j(k)}(b(k)), a(k), b(k)) = 0$$

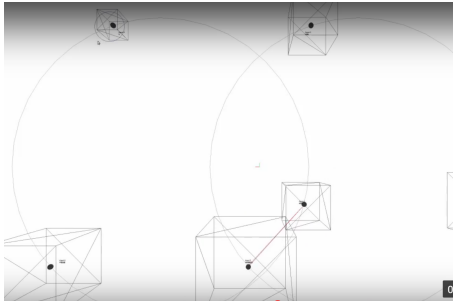
$$\tilde{a}(k) = h_{i(k)}(a(k))$$

$$\tilde{b}(k) = h_{j(k)}(b(k))$$

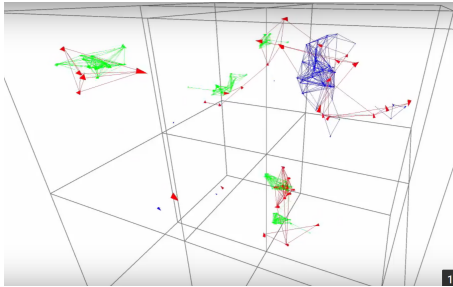
$$\dot{h}_i = 1 + n_h, n_h \in [n_h]$$



[Youtube \[1\]](#)



[Youtube](#)



[Youtube](#)



A. Bethencourt and L. Jaulin.

Cooperative localization of underwater robots with unsynchronized clocks.

Journal of Behavioral Robotics, 4(4):233–244, 2013.



G. Chabert and L. Jaulin.

Contractor Programming.

Artificial Intelligence, 173:1079–1100, 2009.



S. Rohou.

Reliable robot localization: a constraint programming approach over dynamical systems.

PhD dissertation, Université de Bretagne Occidentale, ENSTA-Bretagne, France, december 2017.



S. Rohou, L. Jaulin, M. Mihaylova, F. L. Bars, and S. Veres.

Guaranteed Computation of Robots Trajectories.

Robotics and Autonomous Systems, 93:76–84, 2017.



K. Vencatasamy, L. Jaulin, and B. Zerr.

Secure the zone from intruders with a group robots.

Special Issue on Ocean Engineering and Oceanography,
Springer, 2018.