Set-membership method for proving stability of extended Kalman filters

Auguste Bourgois

ENSTA Bretagne

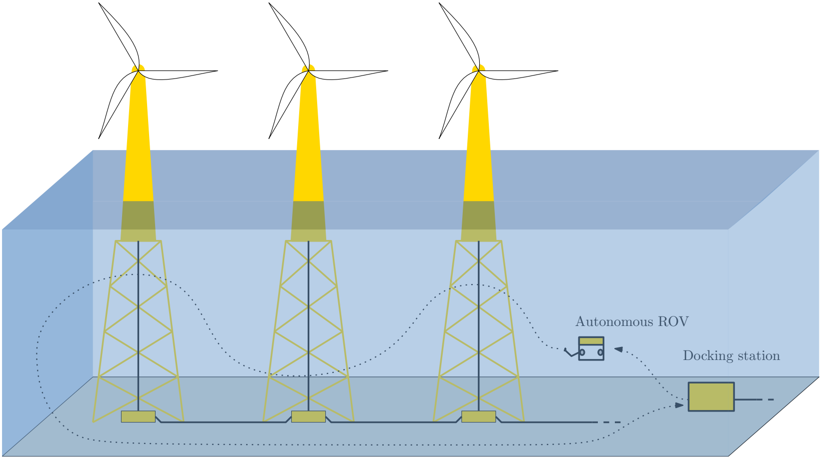
auguste.bourgois@ensta-bretagne.org

International Online Seminar on Interval Methods in Control Engineering
December 4^{th}, 2020

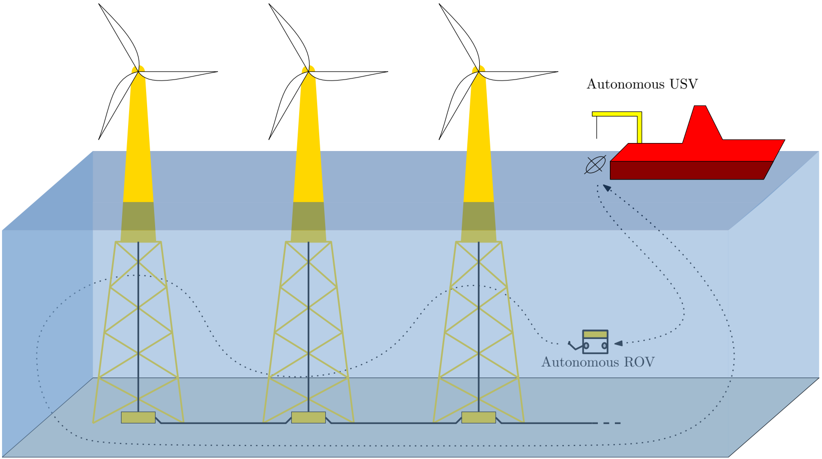
Overview

Introduction

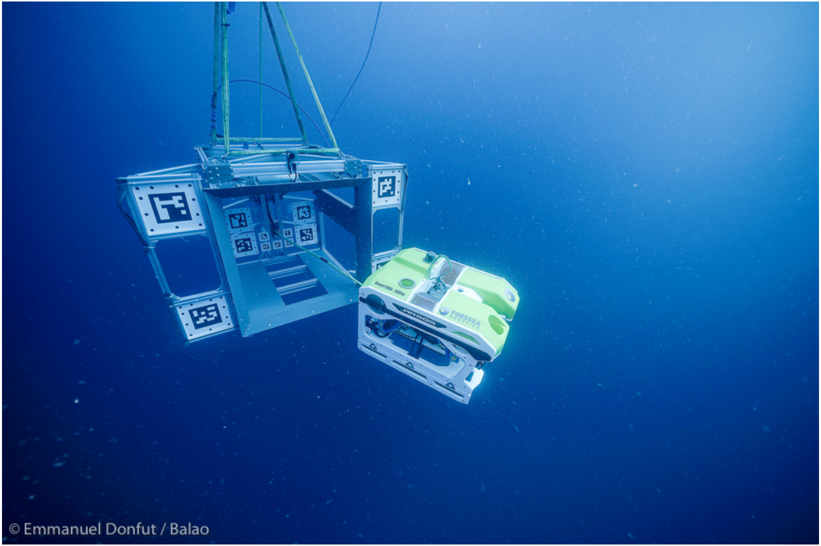
Context I



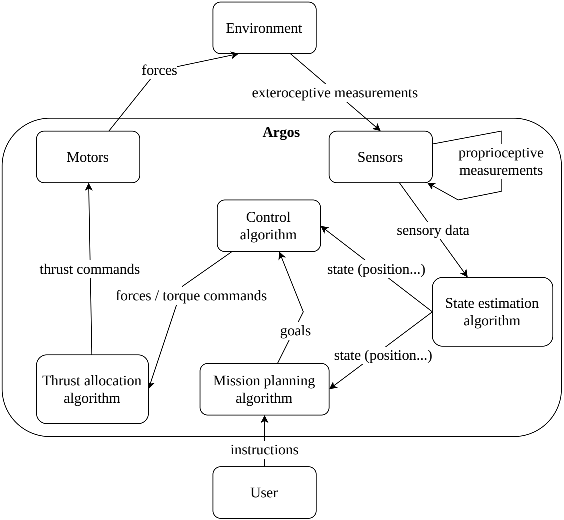
Context II



Underwater docking I



Underwater docking II

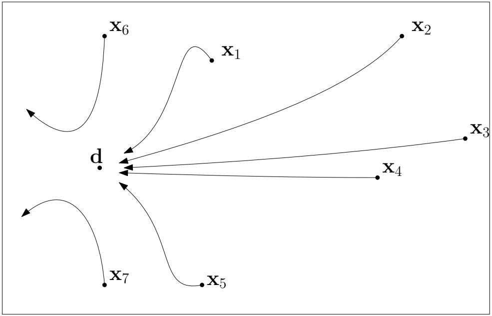


Approach of this thesis

- Model the robot, its environment and its target as a *dynamical system*;

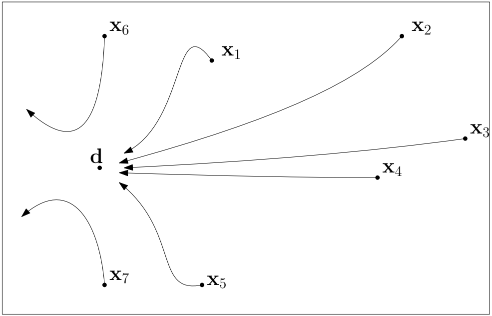
Approach of this thesis

- Model the robot, its environment and its target as a *dynamical system*;
- Model the docking mission as a *stability problem*;



Approach of this thesis

- Model the robot, its environment and its target as a *dynamical system*;
- Model the docking mission as a *stability problem*;
- Prove, a priori, the *feasibility* of the mission



Stability of discrete-time systems

2. Stability of discrete-time systems

- Discrete systems
- Uncertain systems
- Stability of discrete systems

Discrete systems

Definition (Discrete-time system)

A discrete dynamical system is a function $\phi : \mathbb{N} \times \mathbb{R}^n \to \mathbb{R}^n$ such that

1. $\phi(0, .)$ is the identity function : for any $\mathbf{x} \in \mathbb{R}^n$, $\phi(0, \mathbf{x}) = \mathbf{x}$
2. $\phi(p, \phi(q, \mathbf{x})) = \phi(p + q, \mathbf{x})$ for any $p, q \in \mathbb{N}$ and for any $\mathbf{x} \in \mathbb{R}^n$.

Discrete systems

Definition (Discrete-time system)

A discrete dynamical system is a function $\phi : \mathbb{N} \times \mathbb{R}^n \to \mathbb{R}^n$ such that

1. $\phi(0, .)$ is the identity function : for any $\mathbf{x} \in \mathbb{R}^n$, $\phi(0, \mathbf{x}) = \mathbf{x}$
2. $\phi(p, \phi(q, \mathbf{x})) = \phi(p + q, \mathbf{x})$ for any $p, q \in \mathbb{N}$ and for any $\mathbf{x} \in \mathbb{R}^n$.

Relationship with mathematical sequence

The successive states of the system form a mathematical sequence

$$\mathbf{x}_n = \mathbf{f}(\mathbf{x}_0) \quad (1)$$

2. Stability of discrete-time systems

- Discrete systems
- Uncertain systems
- Stability of discrete systems

Uncertain systems [4]

Consider the system

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k\right) \tag{2}$$

Uncertain systems [4]

Consider the system

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k\right) \tag{2}$$

Uncertainties can originate from :

- the initial condition $\mathbf{x}_0 \in [\mathbf{x}_0]$;

Uncertain systems [4]

Consider the system

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k\right) \tag{2}$$

Uncertainties can originate from :

- the initial condition $\mathbf{x}_0 \in [\mathbf{x}_0]$;
- the function $\mathbf{f}$, which can depend of uncertain parameters

Uncertain state

Definition (Interval [2])

An *interval* $[x]$ is a connected subset of $\mathbb{R}$: $[x^-, x^+]$

Uncertain state

Definition (Interval [2])

An *interval* $[x]$ is a connected subset of $\mathbb{R}$: $[x^-, x^+]$
An *interval vector*, or *box*, $[\mathbf{x}]$ is a vector, the members of which are intervals.

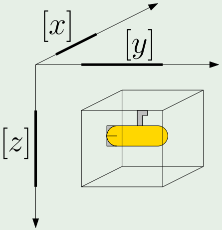
Uncertain state

Definition (Interval [2])

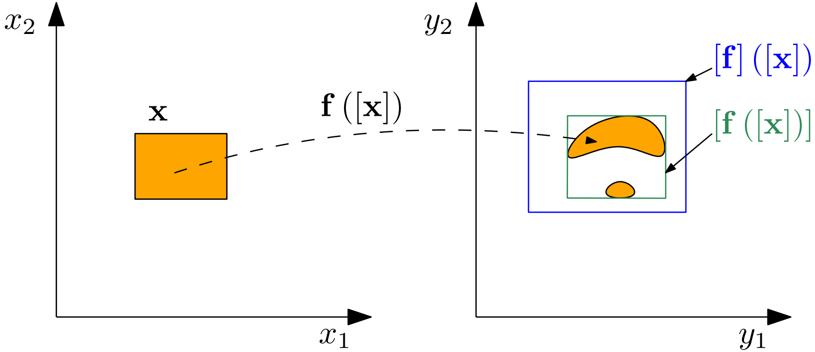
An *interval* $[x]$ is a connected subset of $\mathbb{R}$: $[x^-, x^+]$
An *interval vector*, or *box*, $[\mathbf{x}]$ is a vector, the members of which are intervals.

Example

Consider a robot in the ocean. Its position $\mathbf{x}$ can be represented with an interval vector $[\mathbf{x}]$.



Functions, inclusion functions



Centred form I

Definition (Centred form [3])

Let $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$, with Jacobian $\mathbf{J}$ and $\mathbf{f}(\mathbf{0}) = \mathbf{0}$.
The *centred form* of $\mathbf{f}$ is given by

$$[\mathbf{f}_c]([\mathbf{x}]) = [\mathbf{J}]([\mathbf{x}]) \cdot [\mathbf{x}] \tag{3}$$

Centred form I

Definition (Centred form [3])

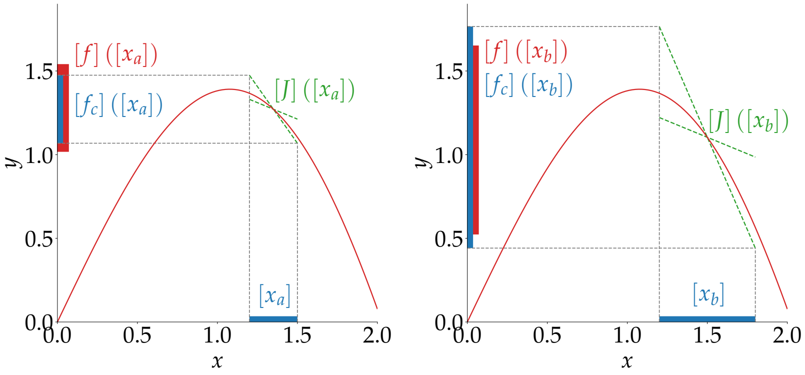
Let $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$, with Jacobian $\mathbf{J}$ and $\mathbf{f}(\mathbf{0}) = \mathbf{0}$.
The *centred form* of $\mathbf{f}$ is given by

$$[\mathbf{f}_c]([\mathbf{x}]) = [\mathbf{J}]([\mathbf{x}]) \cdot [\mathbf{x}] \tag{3}$$

Remark

$[\mathbf{f}_c]$ is an inclusion function for $\mathbf{f}$

Centred form II



2. Stability of discrete-time systems

- Discrete systems
- Uncertain systems
- Stability of discrete systems

Lyapunov's definitions of stability (discrete)

Consider the discrete system described by

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k\right) \tag{4}$$

such that $\mathbf{f}\left(\bar{\mathbf{x}}\right) = \bar{\mathbf{x}}$

Definition (Stability)

The system is *stable* around $\bar{\mathbf{x}}$ if

$$\forall \varepsilon > 0,\ \exists \delta > 0,\ \left\|\mathbf{x}_0 - \bar{\mathbf{x}}\right\| < \delta \implies \forall k \geq 0,\ \left\|\mathbf{x}_k - \bar{\mathbf{x}}\right\| < \varepsilon \tag{5}$$

Definition (Asymptotic stability)

The system is *asymptotically stable* around $\bar{\mathbf{x}}$ if it is *stable* and

$$\exists \delta > 0,\ \left\|\mathbf{x}_0 - \bar{\mathbf{x}}\right\| < \delta \implies \lim_{k \to \infty} \mathbf{x}_k = \bar{\mathbf{x}} \tag{6}$$

Proving stability of uncertain discrete systems

Some methods do exist :

Proving stability of uncertain discrete systems

Some methods do exist :

- using the Jury criterion [1];

Proving stability of uncertain discrete systems

Some methods do exist :

- using the Jury criterion [1];
- using the eigenvalues [5]

Proving stability of uncertain discrete systems

Some methods do exist :

- using the Jury criterion [1];
- using the eigenvalues [5]

Problem

How do we find the stability neighbourhoods ε and δ ?

Stability contractor

Consider the system

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k)$$

such that $\mathbf{f}(\mathbf{0}) = \mathbf{0}$ with Jacobian $\mathbf{J}_\mathbf{f}$ and initial condition $[\mathbf{x}_0]$.

Definition

Let $[\mathbf{a}] \subset [\mathbf{x}_0]$ and $[\mathbf{b}] \subset [\mathbf{x}_0]$ such that $\mathbf{0} \in [\mathbf{a}]$ and $\mathbf{0} \in [\mathbf{b}]$. A *stability contractor* of rate $\alpha < 1$ is an operator $\Psi : \mathbb{IR}^n \to \mathbb{IR}^n$ which satisfies

$$\begin{array}{lr}
[\mathbf{a}] \subset [\mathbf{b}] \implies \Psi([\mathbf{a}]) \subset \Psi([\mathbf{b}]) & \text{(monotonicity)} \\
\Psi([\mathbf{a}]) \subset [\mathbf{a}] & \text{(contractance)} \\
\Psi(\mathbf{0}) = \mathbf{0} & \text{(equilibrium)} \\
\Psi([\mathbf{a}]) \subset \alpha \cdot [\mathbf{a}] \implies \forall k \geq 1,\ \Psi^k([\mathbf{a}]) \subset \alpha^k \cdot [\mathbf{a}] & \text{(convergence)}
\end{array}$$

Iterative centred form as a stability contractor I

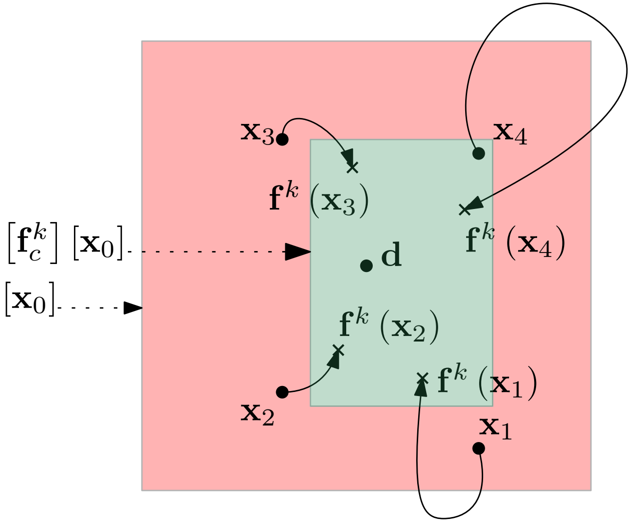
Theorem

The centred form $\left[\mathbf{f}_c^k\right]([\mathbf{x}_0])$ *enclosing* $\mathbf{f}^k([\mathbf{x}_0])$ *is given by*

$$\begin{aligned} [\mathbf{z}_0] &= [\mathbf{x}_0] \\ [\mathbf{A}_0] &= \mathbf{I}_n \\ [\mathbf{z}_{k+1}] &= [\mathbf{f}]([\mathbf{z}_k]) \\ [\mathbf{A}_{k+1}] &= [\mathbf{J}_\mathbf{f}]([\mathbf{z}_k]) \cdot [\mathbf{A}_k] \\ \left[\mathbf{f}_c^k\right]([\mathbf{x}_0]) &= [\mathbf{A}_k] \cdot [\mathbf{x}_0] \end{aligned} \tag{7}$$

Furthermore, if $\exists k > 0$ *such that* $\left[\mathbf{f}_c^k\right]([\mathbf{x}_0]) \subset [\mathbf{x}_0]$, *then* $\left[\mathbf{f}_c^k\right]$ *is a stability contractor and the system is asymptotically stable inside* $[\mathbf{x}_0]$.

Iterative centred form as a stability contractor II

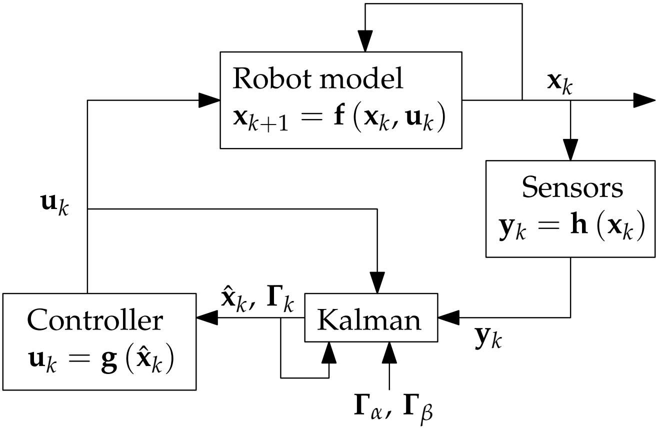


Proving stability of a discrete Kalman filter

3. Proving stability of a discrete Kalman filter

- Reminders about the Kalman filter
- Stability of a localisation system

Kalman filters in robotics



Equations [6]

Two steps :

Equations [6]

Two steps :

- Prediction

$$\hat{\mathbf{x}}_{k+1\,|\,k} = \mathbf{f}\left(\hat{\mathbf{x}}_{k\,|\,k}, \mathbf{u}_k\right) \qquad \text{(predicted estimation)}$$

$$\mathbf{\Gamma}_{k+1\,|\,k} = \mathbf{F}_k \cdot \mathbf{\Gamma}_{k\,|\,k} \cdot \mathbf{F}_k^{\mathrm{T}} + \mathbf{\Gamma}_\alpha \qquad \text{(predicted covariance)}$$

Equations [6]

Two steps :

- Prediction

$$\hat{\mathbf{x}}_{k+1\,|\,k} = \mathbf{f}\left(\hat{\mathbf{x}}_{k\,|\,k}, \mathbf{u}_k\right) \qquad \text{(predicted estimation)}$$

$$\boldsymbol{\Gamma}_{k+1\,|\,k} = \mathbf{F}_k \cdot \boldsymbol{\Gamma}_{k\,|\,k} \cdot \mathbf{F}_k^{\mathrm{T}} + \boldsymbol{\Gamma}_\alpha \qquad \text{(predicted covariance)}$$

- Correction

$$\hat{\mathbf{x}}_{k\,|\,k} = \hat{\mathbf{x}}_{k\,|\,k-1} + \mathbf{K}_k \cdot \tilde{\mathbf{z}}_k \qquad \text{(corrected estimation)}$$

$$\boldsymbol{\Gamma}_{k\,|\,k} = \left(\mathbf{I} - \mathbf{K}_k \cdot \mathbf{H}_k\right) \cdot \boldsymbol{\Gamma}_{k\,|\,k-1} \qquad \text{(corrected covariance)}$$

$$\tilde{\mathbf{z}}_k = \mathbf{y}_k - \mathbf{h}\left(\hat{\mathbf{x}}_{k\,|\,k-1}\right) \qquad \text{(innovation)}$$

$$\mathbf{S}_k = \mathbf{H}_k \cdot \boldsymbol{\Gamma}_{k\,|\,k-1} \cdot \mathbf{H}_k^{\mathrm{T}} + \boldsymbol{\Gamma}_\beta \qquad \text{(innovation's covariance)}$$

$$\mathbf{K}_k = \boldsymbol{\Gamma}_{k\,|\,k-1} \cdot \mathbf{H}_k^{\mathrm{T}} \cdot \mathbf{S}_k^{-1} \qquad \text{(Kalman gain)}$$

Choosing the parameters I

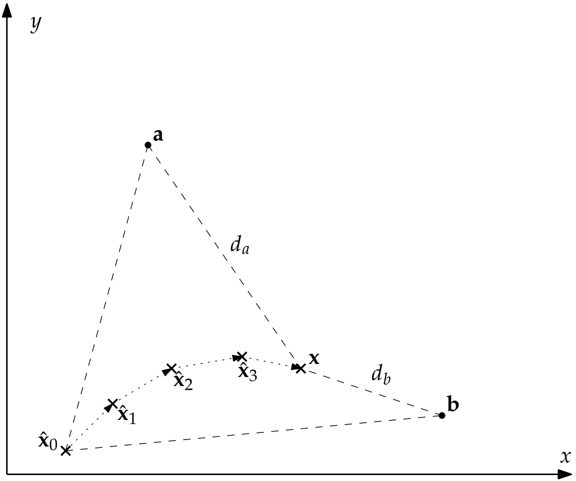
Problem

$\mathbf{\Gamma}_\alpha$, $\mathbf{\Gamma}_\beta$, $\hat{\mathbf{x}}_{0\,|\,0}$, $\mathbf{\Gamma}_{0\,|\,0}$ must be chosen carefully.

Choosing the parameters I

Problem

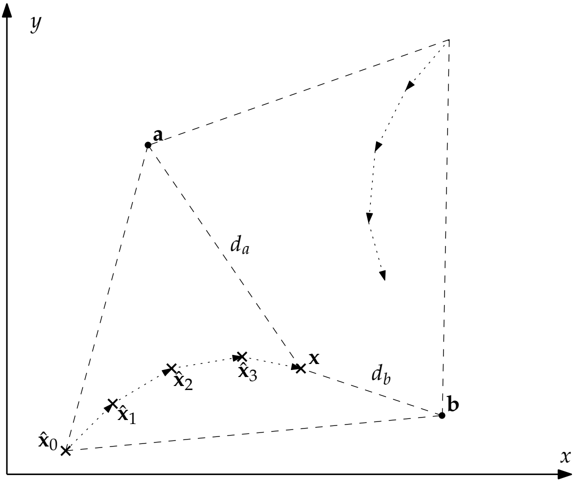
$\boldsymbol{\Gamma}_\alpha$, $\boldsymbol{\Gamma}_\beta$, $\hat{\mathbf{x}}_{0\,|\,0}$, $\boldsymbol{\Gamma}_{0\,|\,0}$ must be chosen carefully.



Choosing the parameters II

Problem

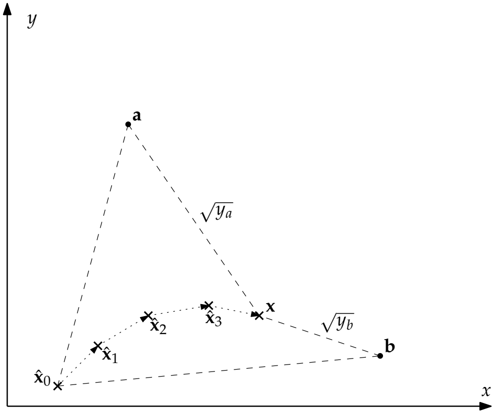
$\mathbf{\Gamma}_\alpha$, $\mathbf{\Gamma}_\beta$, $\hat{\mathbf{x}}_{0\,|\,0}$, $\mathbf{\Gamma}_{0\,|\,0}$ must be chosen carefully.



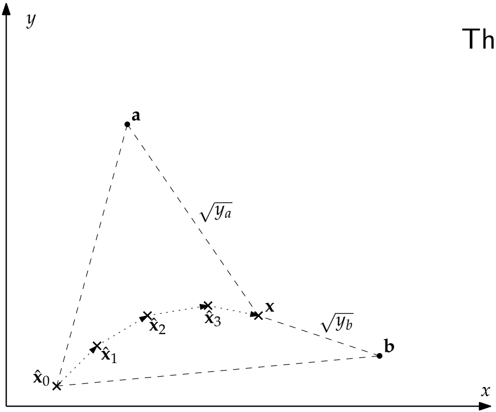
3. Proving stability of a discrete Kalman filter

- Reminders about the Kalman filter
- Stability of a localisation system

Localisation system I



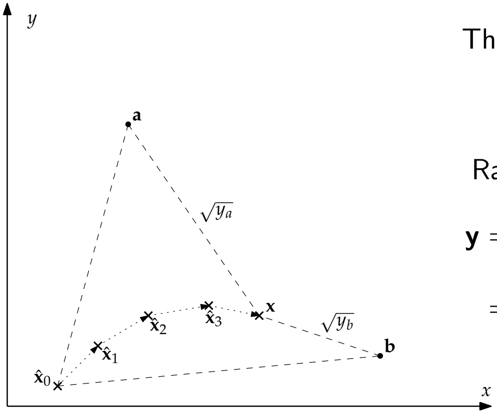
Localisation system I



The robot is not moving :

$$\forall k > 0,\ \mathbf{x}_k = \mathbf{x}_0$$

Localisation system I



The robot is not moving :

$$\forall k > 0,\ \mathbf{x}_k = \mathbf{x}_0$$

Range measurements :

$$\mathbf{y} = \mathbf{h}(\mathbf{x}) = \begin{pmatrix} (x - a_x)^2 + (y - a_y)^2 \\ (x - b_x)^2 + (y - b_y)^2 \end{pmatrix}$$

Localisation system II

Note :

$$\mathbf{u}_k = \mathbf{0}$$
$$\mathbf{f}\left(\mathbf{x}_{k\,|\,k}, \mathbf{u}_k\right) = \mathbf{x}_{k\,|\,k}$$
$$\mathbf{F}_k = \mathbf{I}$$

Localisation system II

Note :

$$\mathbf{u}_k = \mathbf{0}$$
$$\mathbf{f}\left(\mathbf{x}_{k\,|\,k}, \mathbf{u}_k\right) = \mathbf{x}_{k\,|\,k}$$
$$\mathbf{F}_k = \mathbf{I}$$

Denote :

$$\hat{\mathbf{x}}_{k\,|\,k} \leftrightarrow \hat{\mathbf{x}}_k$$
$$\mathbf{\Gamma}_{k+1\,|\,k} \leftrightarrow \mathbf{G}_k$$

Problem

Goal

Find a stability neighbourhood δ around $\mathbf{x}$ such that $\hat{\mathbf{x}}_k \to \mathbf{x}$

Problem

Goal

Find a stability neighbourhood δ around $\mathbf{x}$ such that $\hat{\mathbf{x}}_k \to \mathbf{x}$

Method

- Transform the localisation system into a discrete system;

Problem

Goal

Find a stability neighbourhood δ around $\mathbf{x}$ such that $\hat{\mathbf{x}}_k \to \mathbf{x}$

Method

- Transform the localisation system into a discrete system;
- Identify a fixed point of the system $\bar{\mathbf{v}}$;

Problem

Goal

Find a stability neighbourhood δ around $\mathbf{x}$ such that $\hat{\mathbf{x}}_k \to \mathbf{x}$

Method

- Transform the localisation system into a discrete system;
- Identify a fixed point of the system $\bar{\mathbf{v}}$;
- Choose the parameters and the initial state of the system $[\mathbf{v}_0] \ni \bar{\mathbf{v}}$;

Problem

Goal

Find a stability neighbourhood δ around $\mathbf{x}$ such that $\hat{\mathbf{x}}_k \to \mathbf{x}$

Method

- Transform the localisation system into a discrete system;
- Identify a fixed point of the system $\bar{\mathbf{v}}$;
- Choose the parameters and the initial state of the system $[\mathbf{v}_0] \ni \bar{\mathbf{v}}$;
- Use the stability contractor to check stability.

Formalisation I

The Kalman filter can be seen as a discrete system with state

$$\mathbf{v}_k = (\hat{x}_k, \hat{y}_k, \sigma_{x_k}, \sigma_{y_k}, \sigma_{xy_k}) \tag{8}$$

where

$$\mathbf{\Gamma}_k = \begin{bmatrix} \sigma_{x_k}^2 & \sigma_{xy_k} \\ \sigma_{xy_k} & \sigma_{y_k}^2 \end{bmatrix}$$

Formalisation II

$$\begin{pmatrix} \hat{x}_{k+1} \\ \hat{y}_{k+1} \end{pmatrix} = \begin{pmatrix} \hat{x}_{k} \\ \hat{y}_{k} \end{pmatrix} + \mathbf{K}_k \cdot \tilde{\mathbf{z}}_k \qquad \text{(corrected position)}$$

$$\boldsymbol{\Gamma}_{k+1} = (\mathbf{I} - \mathbf{K}_k \cdot \mathbf{H}_k) \cdot \mathbf{G}_k \qquad \text{(corrected covariance)}$$

$$\tilde{\mathbf{z}}_k = \mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_k) \qquad \text{(innovation)}$$

$$\mathbf{G}_k = \boldsymbol{\Gamma}_k + \boldsymbol{\Gamma}_\alpha \qquad \text{(predicted covariance)}$$

$$\mathbf{K}_k = \mathbf{G}_k \cdot \mathbf{H}_k^{\mathrm{T}} \cdot \mathbf{S}_k^{-1} \qquad \text{(Kalman gain)}$$

$$\mathbf{S}_k = \mathbf{H}_k \cdot \mathbf{G}_k \cdot \mathbf{H}_k^{\mathrm{T}} + \boldsymbol{\Gamma}_\beta \qquad \text{(innovation's covariance)}$$

$$\mathbf{H}_k = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\hat{\mathbf{x}}_k) \qquad \text{(observation matrix)}$$

Formalisation II

$$\begin{pmatrix} \hat{x}_{k+1} \\ \hat{y}_{k+1} \end{pmatrix} = \begin{pmatrix} \hat{x}_{k} \\ \hat{y}_{k} \end{pmatrix} + \mathbf{K}_k \cdot \tilde{\mathbf{z}}_k \qquad \text{(corrected position)}$$

$$\boldsymbol{\Gamma}_{k+1} = (\mathbf{I} - \mathbf{K}_k \cdot \mathbf{H}_k) \cdot \mathbf{G}_k \qquad \text{(corrected covariance)}$$

$$\tilde{\mathbf{z}}_k = \mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_k) \qquad \text{(innovation)}$$

$$\mathbf{G}_k = \boldsymbol{\Gamma}_k + \boldsymbol{\Gamma}_\alpha \qquad \text{(predicted covariance)}$$

$$\mathbf{K}_k = \mathbf{G}_k \cdot \mathbf{H}_k^{\mathrm{T}} \cdot \mathbf{S}_k^{-1} \qquad \text{(Kalman gain)}$$

$$\mathbf{S}_k = \mathbf{H}_k \cdot \mathbf{G}_k \cdot \mathbf{H}_k^{\mathrm{T}} + \boldsymbol{\Gamma}_\beta \qquad \text{(innovation's covariance)}$$

$$\mathbf{H}_k = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\hat{\mathbf{x}}_k) \qquad \text{(observation matrix)}$$

$\Longrightarrow$ we obtain a system of the form

$$\mathbf{v}_{k+1} = \mathbf{p}(\mathbf{v}_k)$$

Formalisation III

Why keeping the covariance prediction equation ?

$$\mathbf{G}_k = \mathbf{\Gamma}_k + \mathbf{\Gamma}_\alpha$$

- to add a Brownian motion...

Formalisation III

Why keeping the covariance prediction equation ?

$$\mathbf{G}_k = \mathbf{\Gamma}_k + \mathbf{\Gamma}_\alpha$$

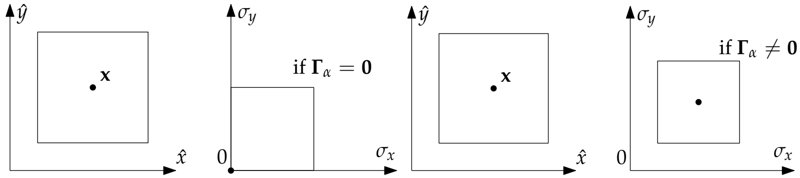
- to add a Brownian motion...
- ... so as to avoid $\mathbf{\Gamma}_k \to \mathbf{0}$

Formalisation III

Why keeping the covariance prediction equation ?

$$\mathbf{G}_k = \mathbf{\Gamma}_k + \mathbf{\Gamma}_\alpha$$

- to add a Brownian motion...
- ... so as to avoid $\mathbf{\Gamma}_k \to \mathbf{0}$



Initialisation

$$\bar{\mathbf{v}} = (0, 0, \sigma_0, \sigma_0, 0)$$

$$[\mathbf{v}_0] = \begin{pmatrix} [-\varepsilon_1, \varepsilon_1] \\ [-\varepsilon_1, \varepsilon_1] \\ [\sigma_0 - \varepsilon_2, \sigma_0 + \varepsilon_2] \\ [\sigma_0 - \varepsilon_2, \sigma_0 + \varepsilon_2] \\ [-\varepsilon_2, \varepsilon_2] \end{pmatrix} \ni \bar{\mathbf{v}}$$

$$\varepsilon_1 = 1 \times 10^{-3}$$

$$\varepsilon_2 = 5 \times 10^{-5}$$

$$\sigma_0 = 3.66 \times 10^{-3}$$

$$\mathbf{\Gamma}_\alpha = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

$$\mathbf{\Gamma}_\beta = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{a} = (-5, 5)$$

$$\mathbf{b} = (5, 5)$$

Results

Results

$$[\mathbf{f}_c]\left([\mathbf{v}_0]-\bar{\mathbf{v}}\right)=\begin{pmatrix}\left[-3.5\times 10^{-4},3.6\times 10^{-4}\right]\\ \left[-3.5\times 10^{-4},3.6\times 10^{-4}\right]\\ \left[-4.63\times 10^{-5},4.68\times 10^{-5}\right]\\ \left[-4.73\times 10^{-5},4.79\times 10^{-5}\right]\\ \left[-1.48\times 10^{-5},1.49\times 10^{-5}\right]\end{pmatrix}\subset[\mathbf{v}_0]-\bar{\mathbf{v}}$$

Results

Results

$$[\mathbf{f}_c]\left([\mathbf{v}_0]-\bar{\mathbf{v}}\right)=\begin{pmatrix}\left[-3.5\times10^{-4},3.6\times10^{-4}\right]\\ \left[-3.5\times10^{-4},3.6\times10^{-4}\right]\\ \left[-4.63\times10^{-5},4.68\times10^{-5}\right]\\ \left[-4.73\times10^{-5},4.79\times10^{-5}\right]\\ \left[-1.48\times10^{-5},1.49\times10^{-5}\right]\end{pmatrix}\subset[\mathbf{v}_0]-\bar{\mathbf{v}}$$

$\Longrightarrow$ the EKF is stable inside $[\mathbf{v}_0]$ and will converge towards $\bar{\mathbf{v}}$

$\Longrightarrow$ $\hat{\mathbf{x}}\rightarrow\mathbf{x}$

Conclusion

Conclusion I

Our method allows to

- Prove stability of discrete-time systems at an equilibrium point;

Conclusion I

Our method allows to

- Prove stability of discrete-time systems at an equilibrium point;
- Find a stability neighbourhood for the system...

Conclusion I

Our method allows to

- Prove stability of discrete-time systems at an equilibrium point;
- Find a stability neighbourhood for the system...
- ... to approximate basins of attraction.

Conclusion I

Our method allows to

- Prove stability of discrete-time systems at an equilibrium point;
- Find a stability neighbourhood for the system...
- ... to approximate basins of attraction.

However

- Limited to small neighbourhoods (centred form);

Conclusion I

Our method allows to

- Prove stability of discrete-time systems at an equilibrium point;
- Find a stability neighbourhood for the system...
- ... to approximate basins of attraction.

However

- Limited to small neighbourhoods (centred form);
- $\Longrightarrow$ requires additional bisection algorithms to be fully usable on a real life system

Conclusion II

Applied to an EKF, our method :

- allows to validate a priori the EKF parameters...

Future prospects :

- Apply to a moving robot (discrete-time);
- and apply to a hybrid system : continuous-time model for the robot, and discrete-time for the EKF & the controller.

Conclusion II

Applied to an EKF, our method :

- allows to validate a priori the EKF parameters...
- ... to ensure convergence towards the actual state of the system.

Future prospects :

- Apply to a moving robot (discrete-time);
- and apply to a hybrid system : continuous-time model for the robot, and discrete-time for the EKF & the controller.

References I

[1] L. Jaulin and J. Burger.
Proving stability of uncertain parametric models.
Automatica, page 627, 1999.

[2] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter.
Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics.
Springer-Verlag, London, 2001.

[3] R. E. Moore, R. B. Kearfott, and M. J. Cloud.
Introduction to interval analysis.
SIAM, 2009.

References II

[4] A. Rauh, J. Kersten, and H. Aschemann.
Interval methods and contractor-based branch-and-bound procedures for verified parameter identification of quasi-linear cooperative system models.
J. Comput. Appl. Math., 367, 2020.

[5] J. Rohn.
An algorithm for checking stability of symmetric interval matrices.
IEEE Transactions on Automatic Control, 41(1):133–136, 1996.

[6] S. Thrun, W. Burgard, and D. Fox.
Probabilistic robotics.
The MIT Press, Cambridge, Massachusetts, 2006.

Thank you for your attention !

Questions ?