UNIVERSITÉ LAVAL

GLO-4001/7021 INTRODUCTION À LA ROBOTIQUE MOBILE

Filtres à particules

(Automne 2017)

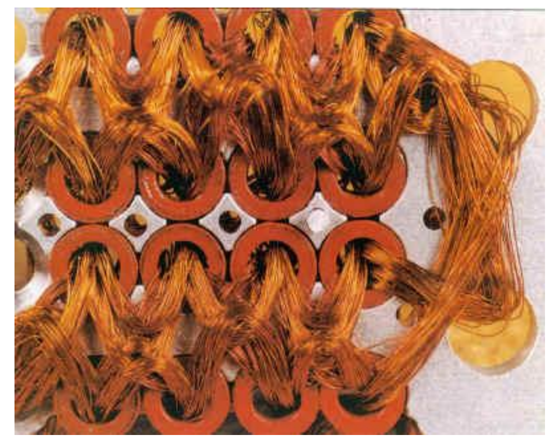
Philippe Giguère

Estimation d'état : filtres à particules

Estimation d'état : filtre Kalman

- Bruit gaussien
- Distribution unimodale
- Système linéaire :
  - filtre Kalman
- Système non linéaire :
  - filtre Kalman étendu (EKF)
  - filtre Kalman non-parfumé (UKF)
- Calcul rapide à effectuer
  - Apollo Guidance Computer (AGC)
  - 2.048 MHz
  - 2 048 x 16 RAM
  - 36 864 x 16 ROM

Interface DSKY du AGC



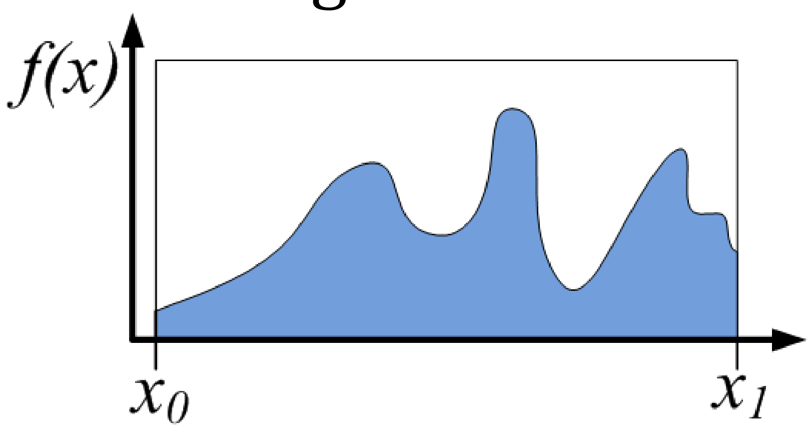
ROM tressé à la main: core rope memory

Méthodes Monte-Carlo

- Estimation de fonctions par méthodes aléatoires, par échantillonnage au hasard
- Développées durant le projet Manhattan, années 1940.

http://hackaday.com/2015/09/11/fermiac-the-computer-that-advanced-the-manhattan-project/

- Pour problèmes multi-variables ou sans solutions analytiques
- Exemple : calcul d'intégrale définie



Filtrage Bayésien (rappel)

Algorithme FiltreBayes $(bel(x_{t-1}), u_t, z_t)$ distribution

for all x_t **do**

Prédiction $\Rightarrow$ $$bel_{pred}(x_t) = \int p(x_t \mid u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1} \quad \textbf{(1)}$$

Mise-à-jour $\Rightarrow$ $$bel(x_t) = \eta\, p(z_t \mid x_t) bel_{pred}(x_t) \quad \textbf{(2)}$$

endfor

return $bel(x_t)$

$\boldsymbol{bel()}$: *belief* ou croyance (raccourci de notation)

$\boldsymbol{u_t}$: commande au temps t

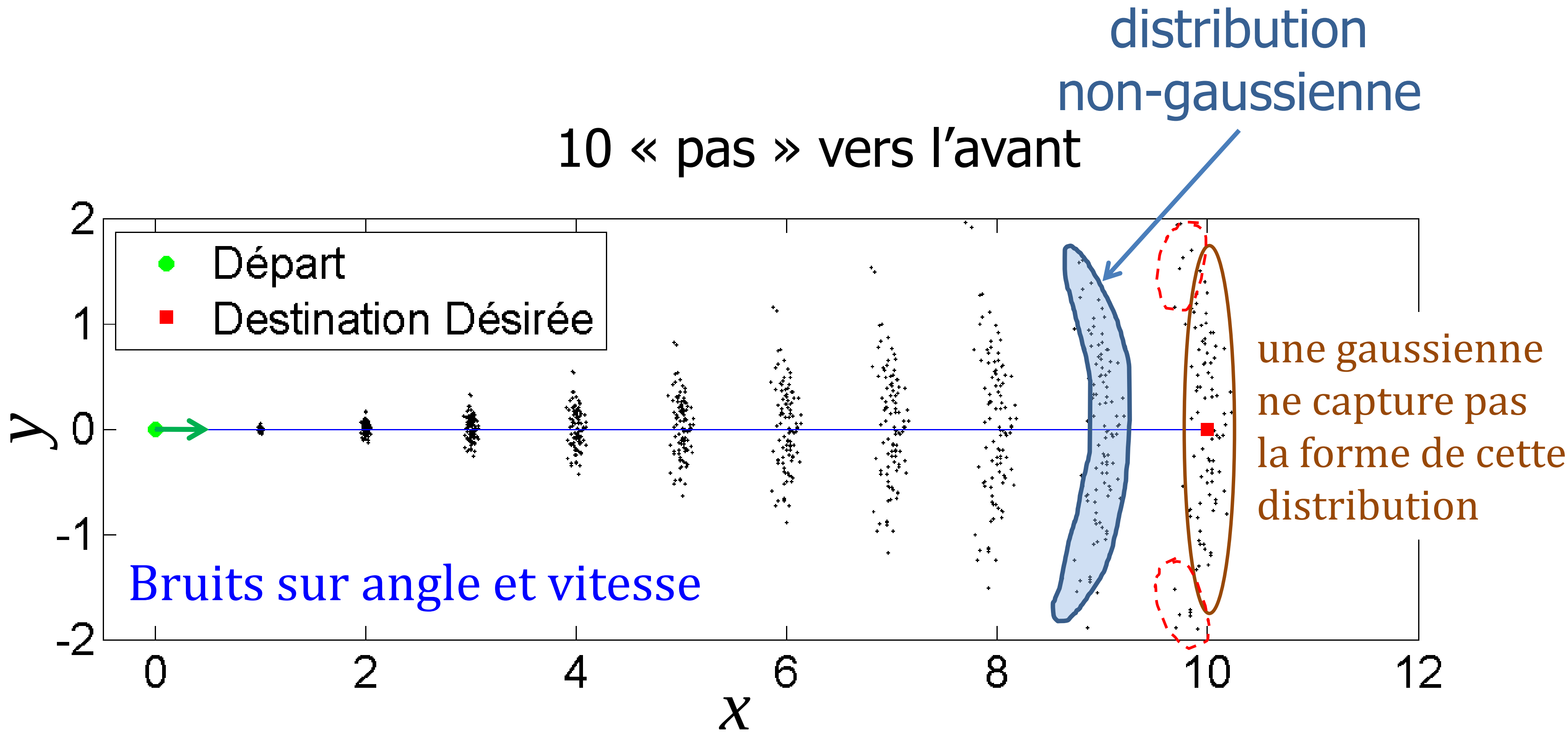
$\boldsymbol{z_t}$: mesure au temps t, après la commande $\boldsymbol{u_t}$

$\boldsymbol{x_t}$: pose du robot au temps t, après $\boldsymbol{u_t}$ et $\boldsymbol{z_t}$

η : facteur de normalisation

Difficile à implémenter <u>exactement</u> car pas toujours de solution analytique de l'intégrale (1)/produit (2)

Méthodes Monte-Carlo



100 simulations == 100 particules

Monte-Carlo → filtre à particules

- Chaque particule représente une hypothèse

$i^{\text{ème}}$ particule : $X_i = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$ (pour un robot 2D)

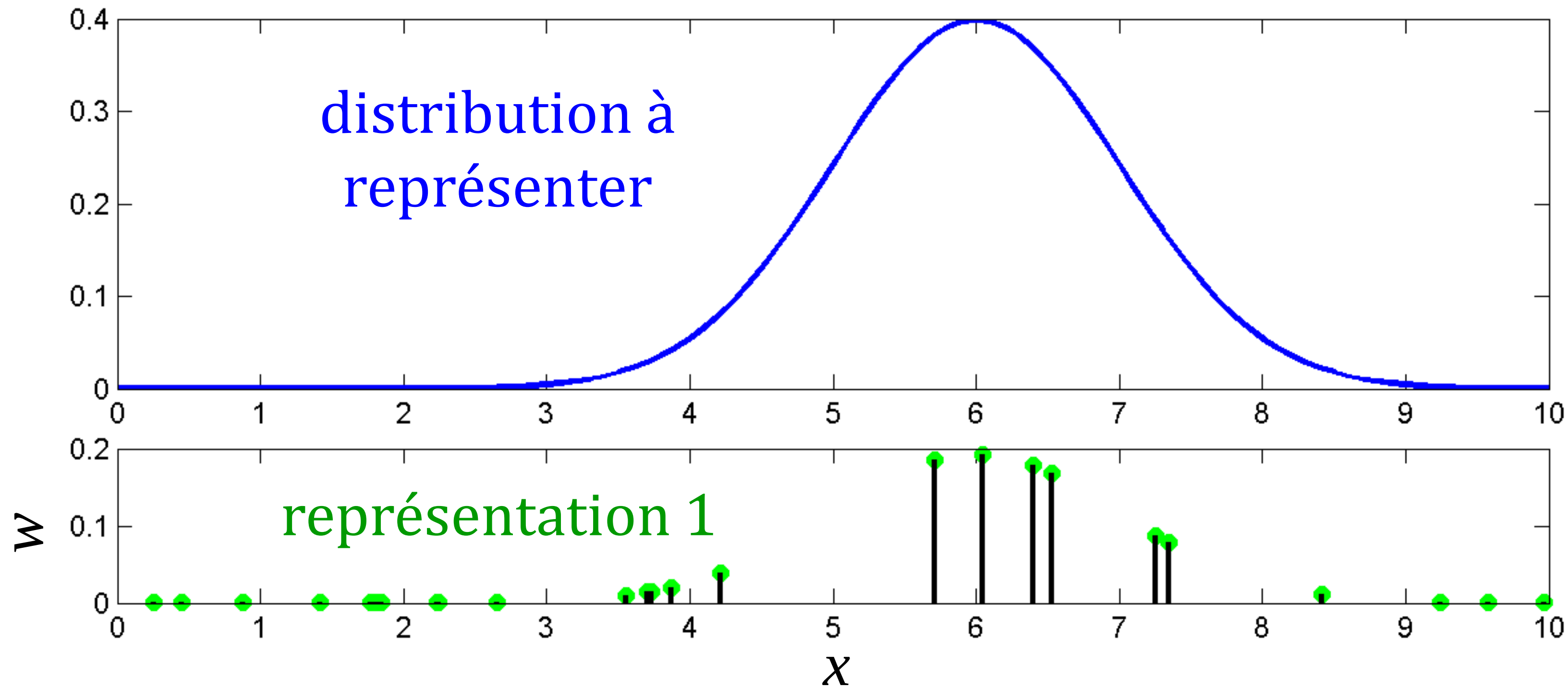
- Chaque particule a un poids associé

$$w_i$$

- Le poids w_i représente en quelque sorte la « confiance » sur cette hypothèse
- Filtre utilise (en général) un nombre C fixe de particules

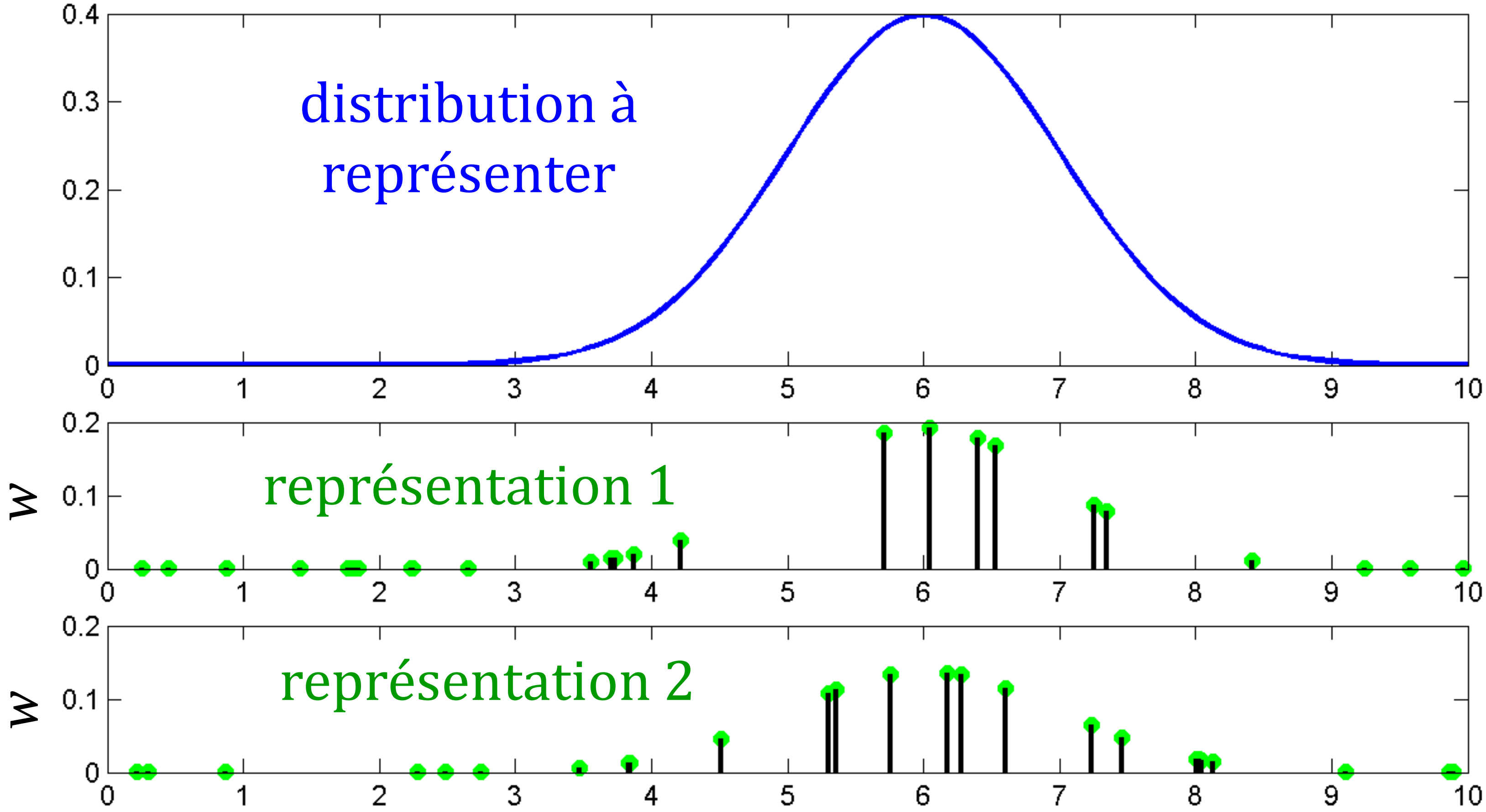
UNIVERSITÉ LAVAL

Exemple : distribution unimodale



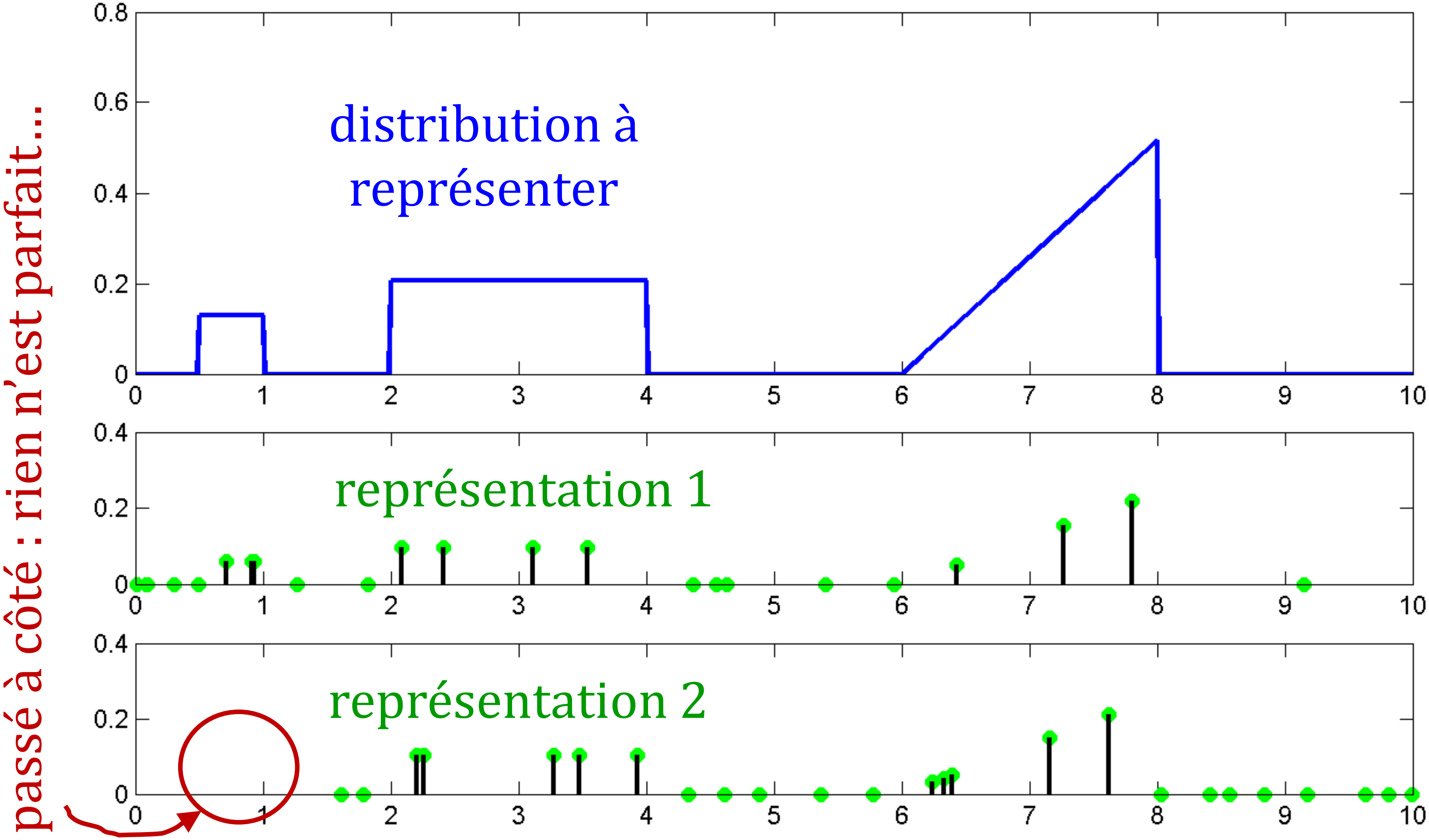
- On pige une valeur x au hasard
- Le poids w est proportionnel à la hauteur de la distribution

Exemple : distribution unimodale



Exemple : distribution multimodale

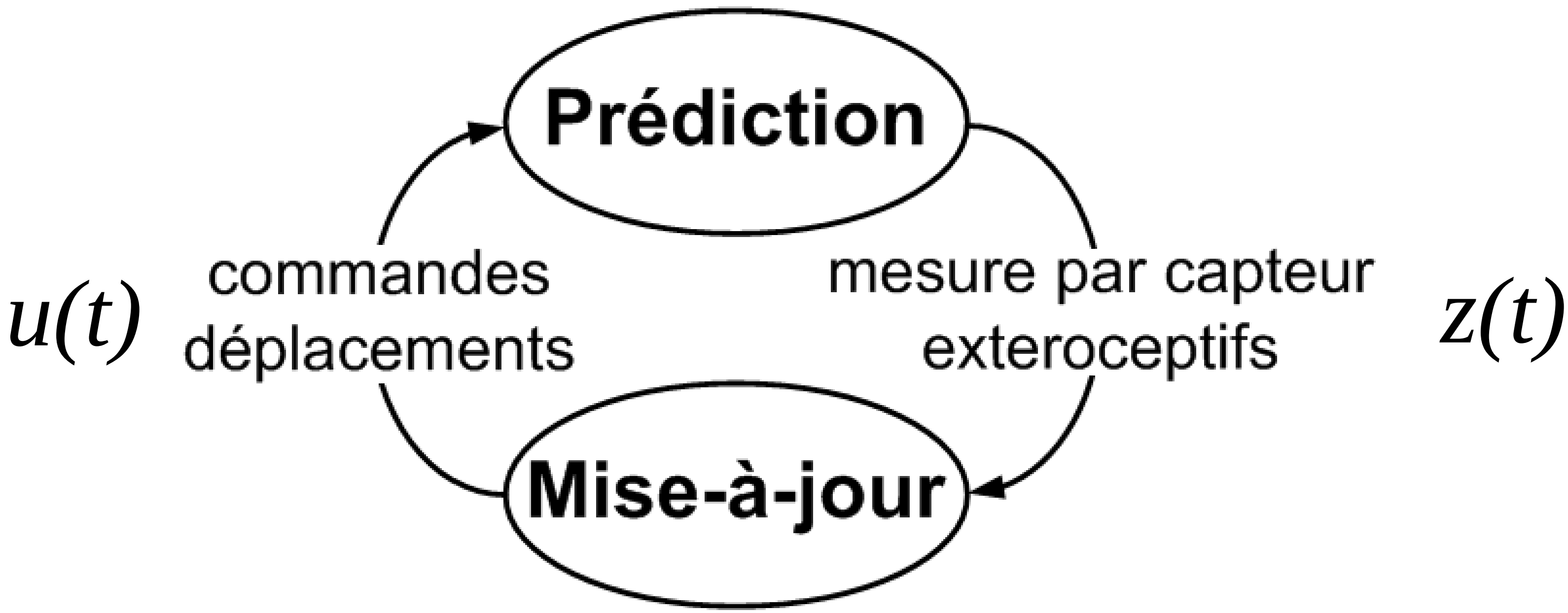
Non-gaussienne



Variance → étendu distribution

- On ne calcule plus la covariance P explicitement comme dans Kalman
- <u>Implicite</u> par la distribution des particules
- Permet de représenter des distributions arbitrairement complexes

Filtre à particule



Intègre l'**information** en 2 étapes,
tout comme les filtres de *Kalman*

Filtre à particules : aperçu

```
while (explore)
  for i=1:C
```
on simule avec le bruit!

$$X_i(k+1) = f_X(X_i(k), u(k), \sigma_V)$$ Prédiction

```
  end
```
$z(k+1)$ `= mesure();`
```
  for i=1:C
```
$$w_i(k+1) = p(z(k+1) \mid X_i(k+1))w_i(k)$$ Mise-à-jour

```
  end
  for i=1:C
```
$$w_i(k+1) = \frac{w_i(k+1)}{\sum_j \{w_j(k+1)\}}$$ Normalisation

```
  end
```
`Neff = 1/`$\sum_{i=1}^{N} w_i^2$
```
  if (Neff < Nseuil)
```
$X_i(k+1)$ `= resample(`$X_i(k+1)$ `,` $w_i(k+1)$`);` Ré-échantillonnage

$w_i(k+1) = 1/C$
```
  end
  k=k+1
end
```

Étape 1 : prédiction du F. P.

- On déplace la particule selon :
  - le modèle du système,
  - les commandes $u(k)$,
  - le **bruit *v***

$$X_i(k+1) = f_X(X_i(k), u(k), \sigma_V)$$

pas besoin d'avoir $f_X(\bullet)$ linéaire…

pour notre robot 2D

$$\boxed{\begin{aligned} \Delta d &= (V + N(0,\sigma_V))\Delta t \\ f_x &= x + \Delta d \cos\phi \\ f_y &= y + \Delta d \sin\phi \\ f_\phi &= \phi + \Delta t(\omega + N(0, v_\omega)) \end{aligned}}$$

- On répète pour i=1:C
- Bref on déplace les particules en simulant le robot avec le bruit

UNIVERSITÉ LAVAL

Étape 1 : prédiction, Kalman vs. F.P.

- Dans Kalman (étendu ou pas), on ne propage pas le bruit sur la pose

Kalman

$$\hat{x}(k+1|k) = \Phi\hat{x}(k) + \Gamma u(k)$$

Kalman étendu

$$\hat{X}(k+1|k) = f_X(\hat{X}(k), u(k))$$

- Dans filtre à particule, on propage avec du bruit

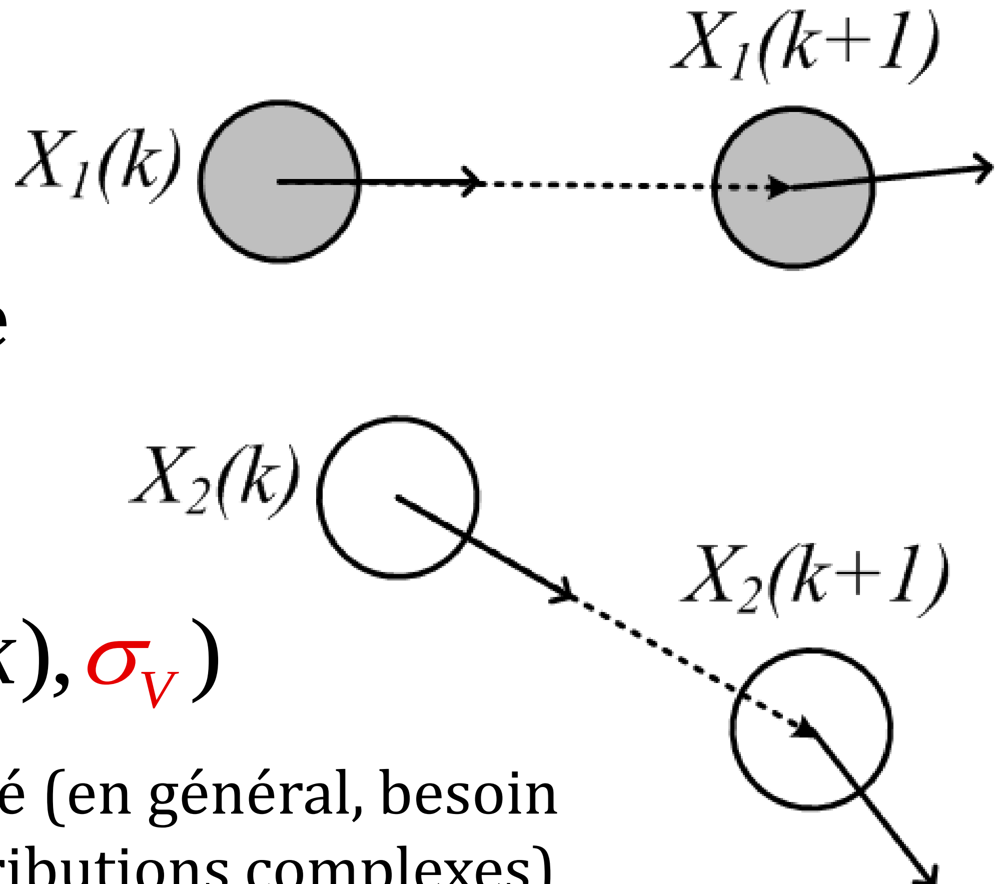
$$X_i(k+1) = f_X(X_i(k), u(k), \sigma_V)$$

- Et il n'y a pas de matrice de covariance P

$$\xcancel{P(k+1|k) = \Phi(k)P(k)\Phi(k)^T + C_v(k)}$$

Filtre à particules : exemple prédiction

- Deux particules, X_1 et X_2
- Vitesse constante V
- Léger bruit sur angle
- Léger bruit sur vitesse



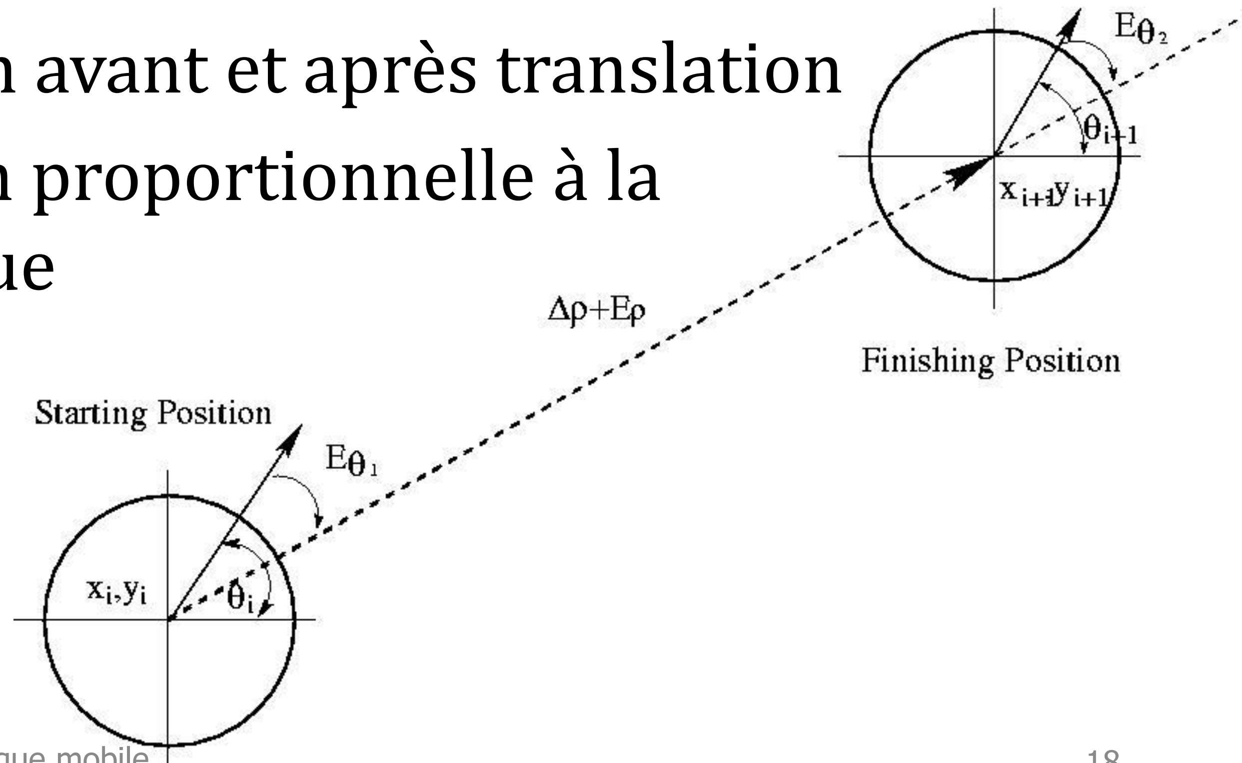
$$X_i(k+1) = f_X(X_i(k), u(k), \sigma_V)$$

peut être arbitrairement compliqué (en général, besoin de plus de particules pour les distributions complexes)

Autre exemple bruit pour un pas

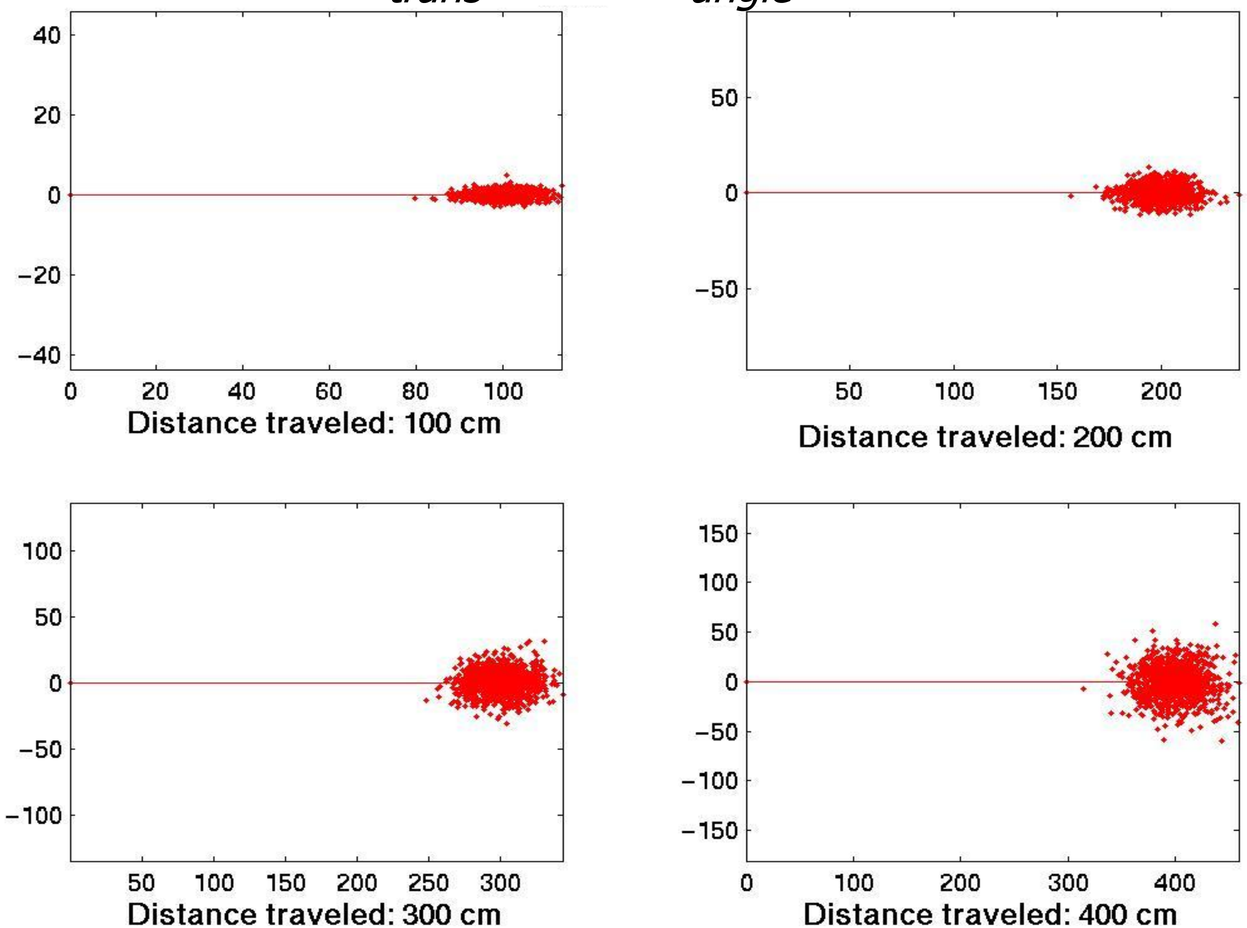
- Rotation : bruit gaussien
- Translation : bruit gaussien
- Un pas:
  - erreur de rotation avant et après translation
  - erreur translation proportionnelle à la distance parcourue

(permet de mieux décorréler *x*, *y* et θ car autant de sources de bruits que de degrés de liberté)



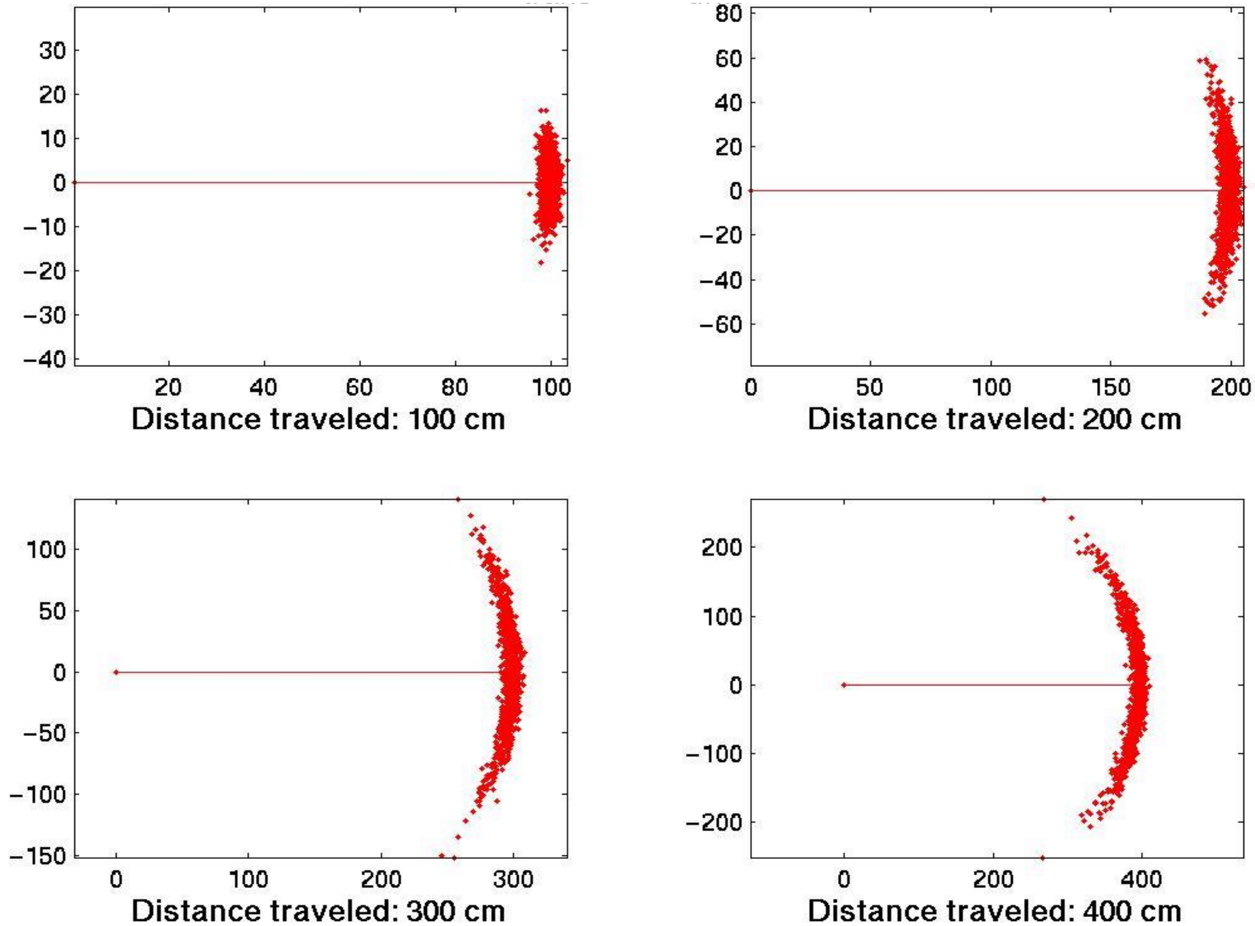
Modèle bruit déplacement

$\sigma_{trans} = 5 \quad \sigma_{angle} = 1$



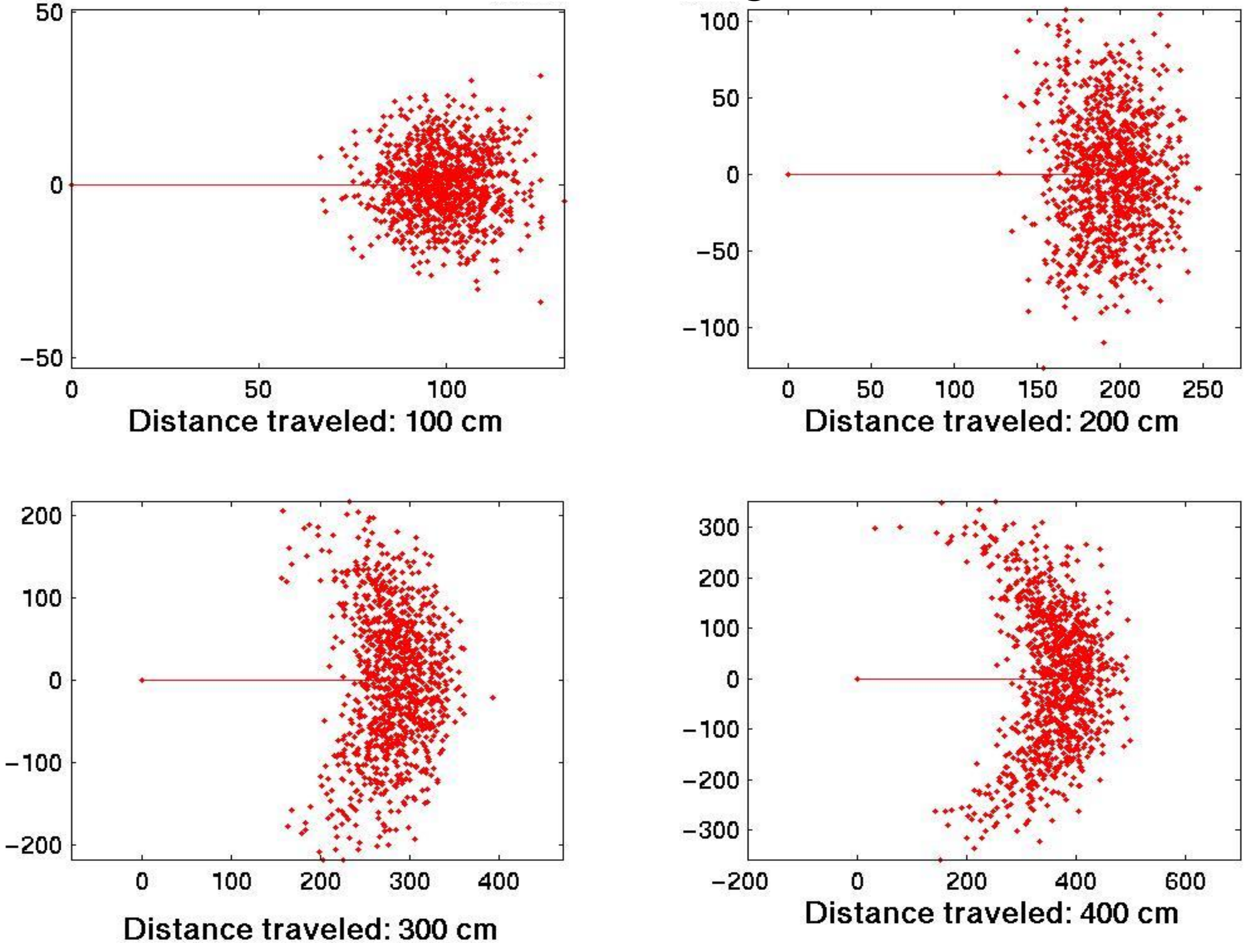
Modèle bruit déplacement

$\sigma_{trans} = 1 \quad \sigma_{angle} = 5$



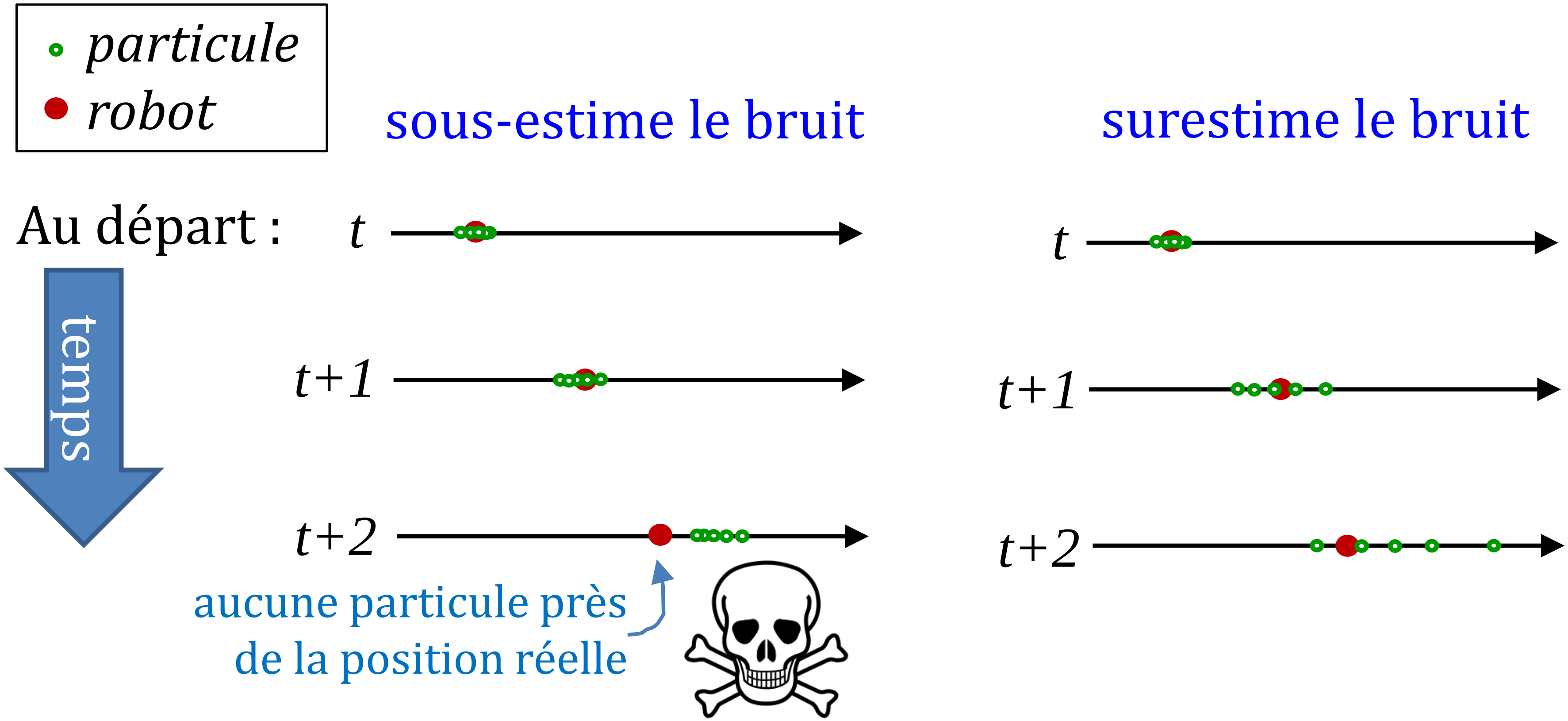
Modèle bruit déplacement

$$\sigma_{trans} = 10 \quad \sigma_{angle} = 10$$



Bien modéliser le bruit de déplacement

- Au pire, on <u>surestime</u> le bruit
  - sous-estimer est néfaste… (divergence du filtre à particule)

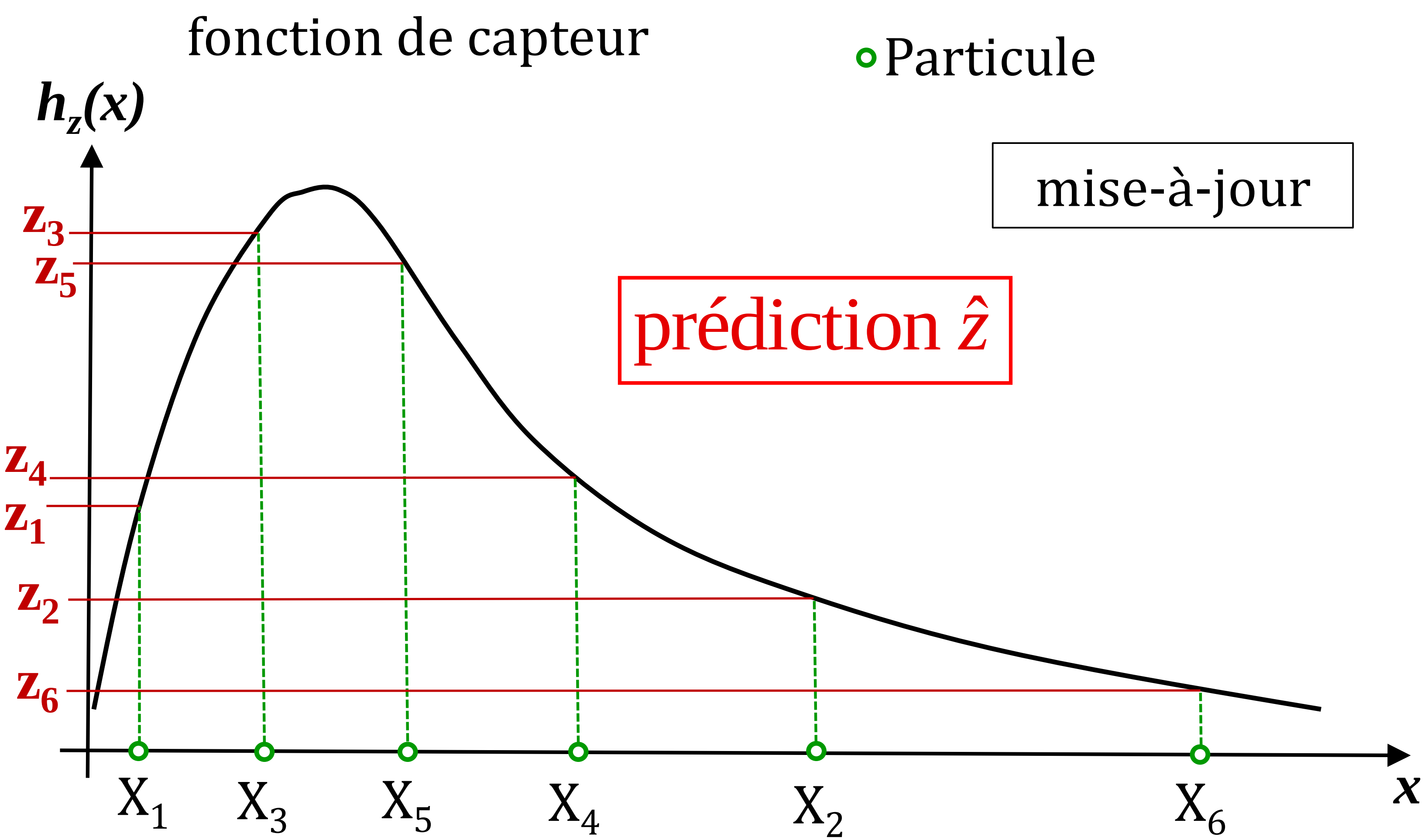


Étape 2 : Mise-à-jour

- Incorporer l'information de capteurs <u>extéroceptifs</u>
- Vient réduire l'incertitude sur la position
- Utiliser la fonction du capteur $h_z(X)$ + bruit estimé pour ajuster le poids w_i de chaque particule

Exemple de filtre, fonction non-bijective

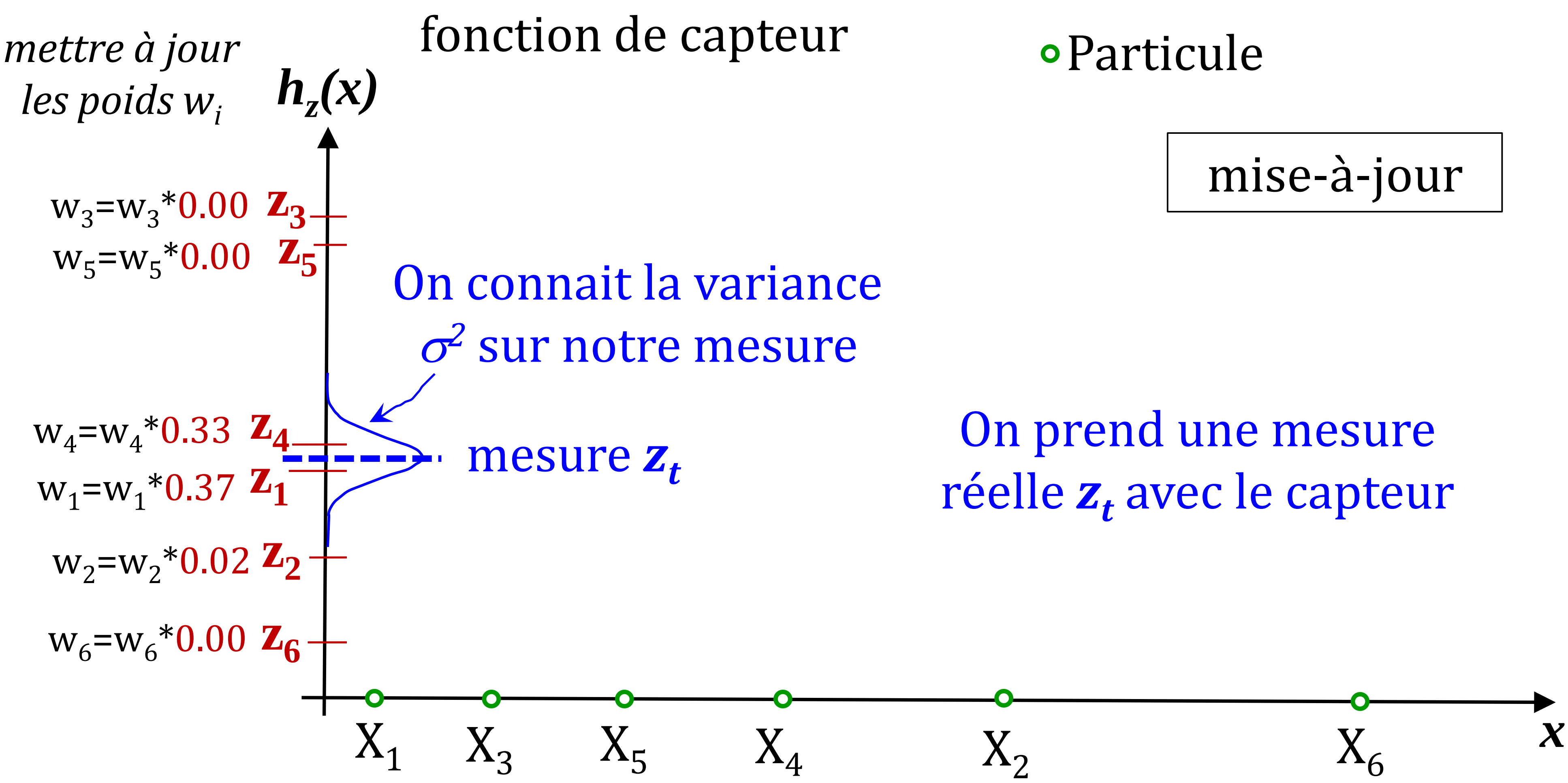
Prédire la mesure attendue pour chaque particule X_i



particules X_i un peu partout : je suis perdu!

UNIVERSITÉ LAVAL

Exemple de filtre, fonction non-bijective

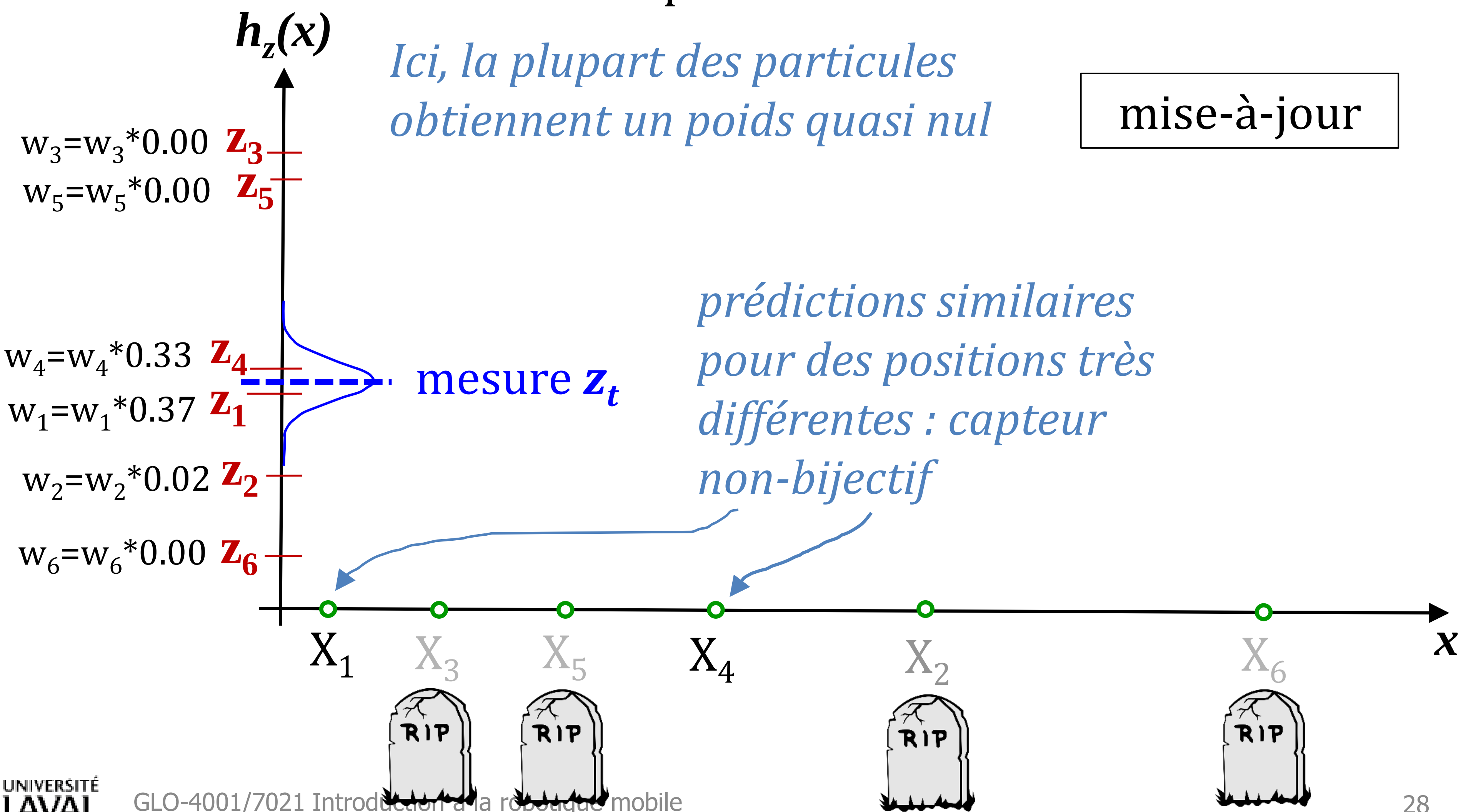


Exemple de filtre, fonction non-bijective

fonction de capteur

Ici, la plupart des particules obtiennent un poids quasi nul

mise-à-jour



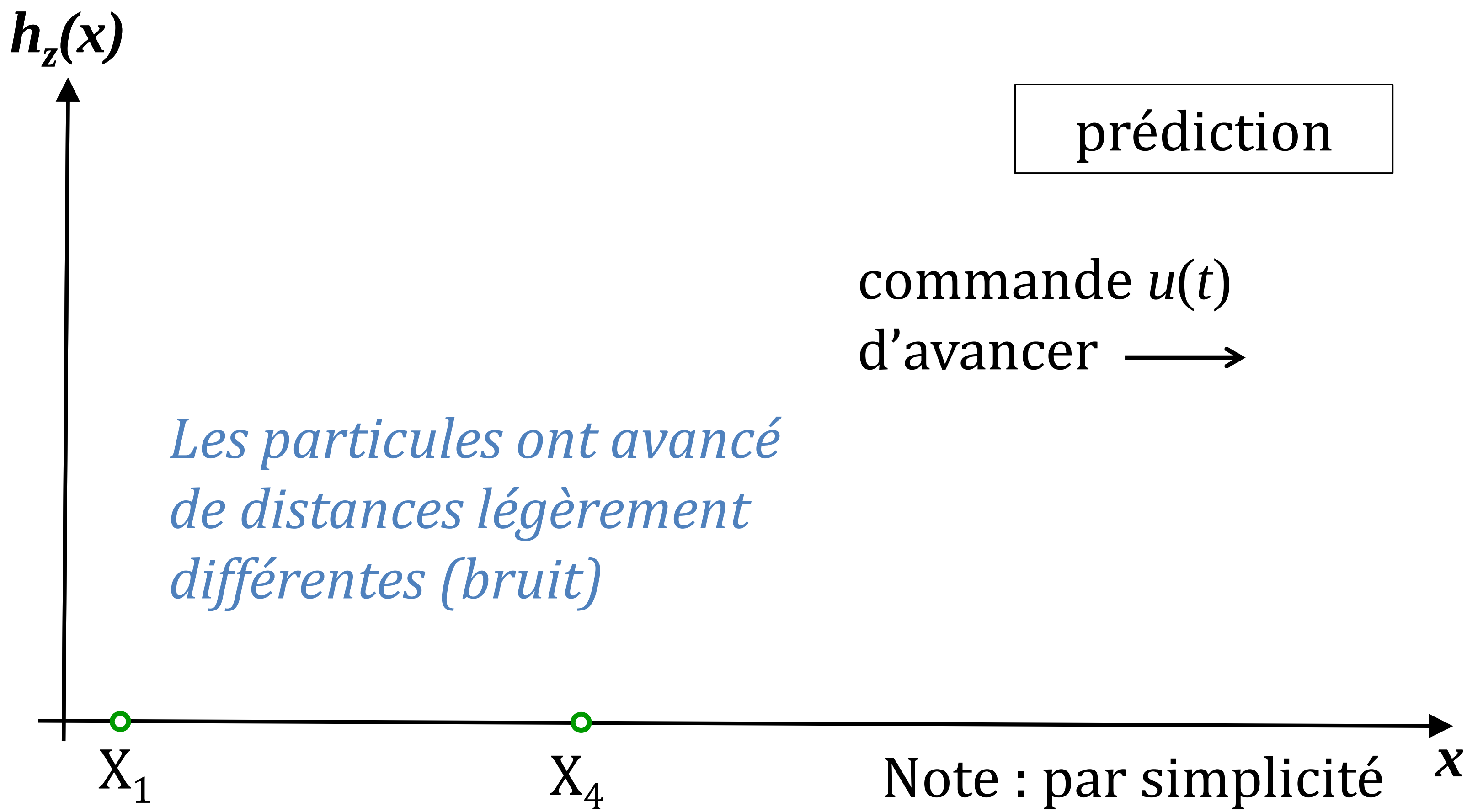
prédictions similaires pour des positions très différentes : capteur non-bijectif

Exemple de filtre, fonction non-bijective

prédiction

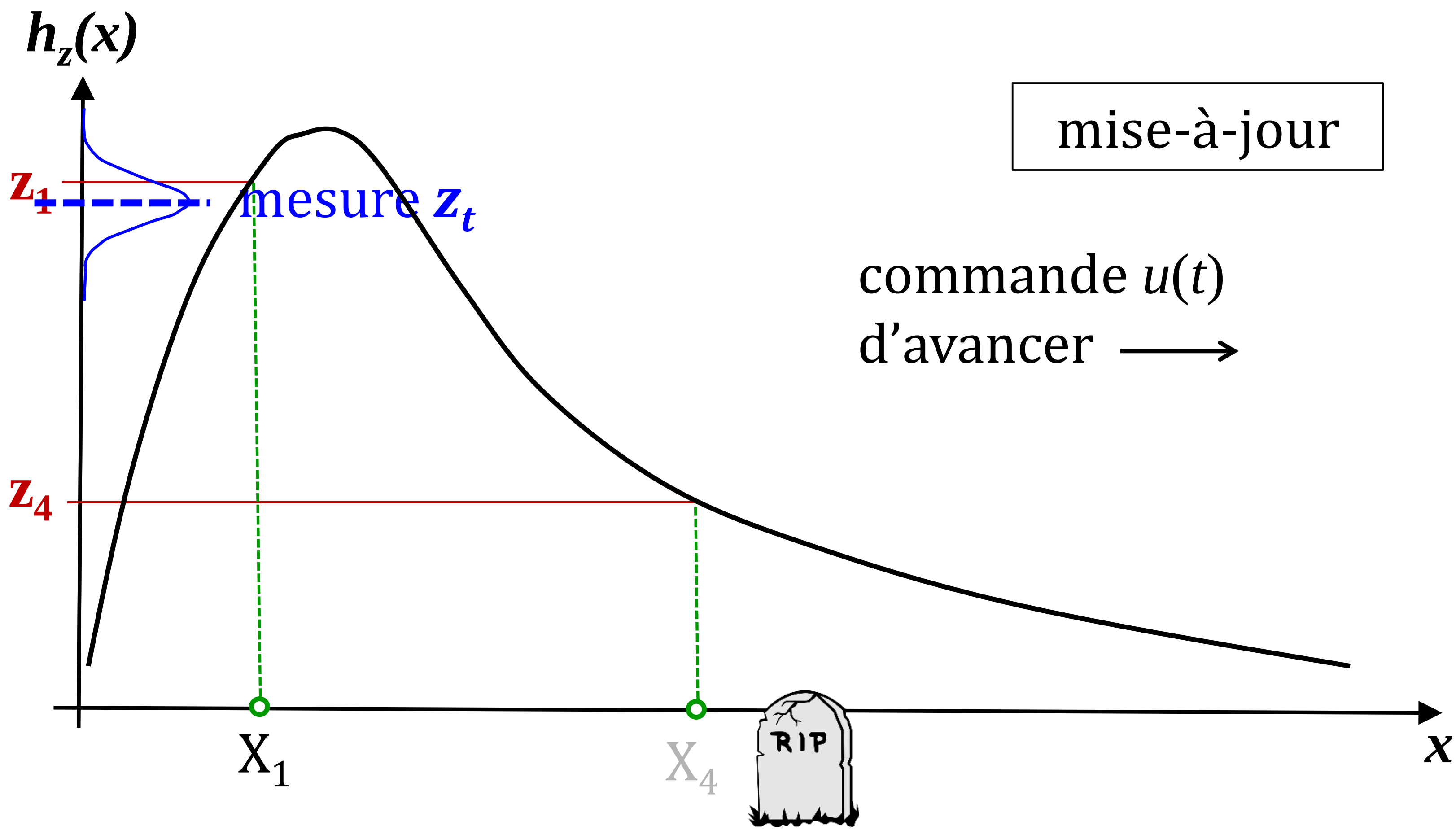
commande $u(t)$
d'avancer →

Les particules ont avancé de distances légèrement différentes (bruit)



Note : par simplicité je n'affiche plus les autres particules ayant des w_i faibles

Exemple de filtre, fonction non-bijective



Le filtre à particule a donc réussi à isoler la bonne hypothèse, après deux pas!