Review of Hybridizations of Kalman Filters with Fuzzy and Neural Computing for Mobile Robot Navigation

M. Graña, I. Villaverde, J.M. Lopez Guede, B. Fernandez

Grupo Inteligencia COMputacional UPV/EHU

www.ehu.es/ccwintco

Contents

- Introduction
- Some definitions of Kalman Filters
- Hybridizations
- Conclusions

Introduction

- Kalman Filters (KF) are at the root of many computational solutions for autonomous systems navigation problems, besides other application domains.
- KF have been used for Simultaneous Localization and Mapping (SLAM):
  - To estimate the actual position, while creating a map of the environment
  - To estimate its uncertainty

Introduction

- The basic linear formulation has been extended in several ways to cope with non-linar dynamic environments.
- Introduce Computational Intelligence (CI) tools, inside its computational loop, such as
  - Fuzzy Systems or
  - Artificial Neural Networks.

Introduction

- We have found that the main KF hybridizations are:
    1. using KF as an estimation algorithm to train computational systems , instead of the simple gradient descent algorithms,
    2. using CI tools to model the KF elements more realistically,
    3. mixing EKF with other (fuzzy) representations.

Kalman Filters

- SLAM: Simultaneous self-localization and mapping
- System state at time *k* : $x_k \in \Re^n$
- Dynamic model $x_k = f(x_{k-1}, u_{k-1}, w_{k-1}),$
  - Linear case $x_k = A_k x_{k-1} + B_k u_k + w_k$

w_{k-1} is the motion noise

u_{k-1} the motion command

Kalman filters

- Environment measurements z_k
- Observation model $z_k = h(x_k, v_k)$
  - Linear case $z_k = H_k x_k + v_k$

observation noise v_k,

Kalman filters

- The KF is a recursive method to estimate the state of a system.
- Two steps
  - Prediction $(\hat{x}_k^- \text{ and } P_k^-)$
    - A priori state and error covariance matrix predicted by the dynamical model
  - Correction $(\hat{x}_k^+ \text{ and } P_k^+)$
    - Based on the actual observation

Kalman filters

- Discrete (linear) Kalman Filter (DKF)
  - Prediction $$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1};$$ $$P_k^- = AP_{k-1}A^T + Q,$$
  - Correction $$\hat{x}_k^+ = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-);$$ $$P_k^+ = (I - K_kH)P_k^-,$$
  - Kalman Gain K_k

Kalman filters

- The Extended Kalman Filter (EKF)
  - Deals with non-linear models via linearization
  - Assumes that the non-linear models are known, so we can use them for the prediction step

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_{k-1}, 0);$$

$$\hat{z}_k^- = h(\hat{x}_k^-, 0).$$

  - The correction step needs to compute the Jacobian matrices at each time instant

Kalman filters

- EKF correction step is based on the linearized model

$$x_k \approx \hat{x}_k^- + A(x_{k-1} - \hat{x}_{k-1}) + W w_{k-1};$$

$$z_k \approx \hat{z}_k^- + H(x_k - \hat{x}_k^-) + V v_k,$$

- So that the DKF correction equations apply.

Kalman filters

- EKF disadvantages
  - Sensitivity to initial conditions
  - Computation of Jacobian matrices
  - Unstable numerically

Kalman filters

- Unscented Kalman Filter
  - Replaces the linearization with the unscented transformation
    - Chose a set of points according to the error covariance matrix $\{\chi_i\}$
    - Apply the non-linear model to them $\gamma_i = f[\chi_i]$
    - Compute weighted averages

$$\bar{y} = \sum_{i=0}^{2n} W_i^{(m)} \gamma_i \qquad\qquad P_y = \sum_{i=0}^{2n} W_i^{(c)} (\gamma_i - \bar{y})(\gamma_i - \bar{y})^T .$$

Kalman filters

- UKF prediction

$$\hat{x}_k^- = \bar{y} \text{ and } P_k^- = P_y$$

- UKF correction

$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - \hat{z}_k^- \right);$$

$$P_k = P_k^- - K_k P_{z_k} K_k^T,$$

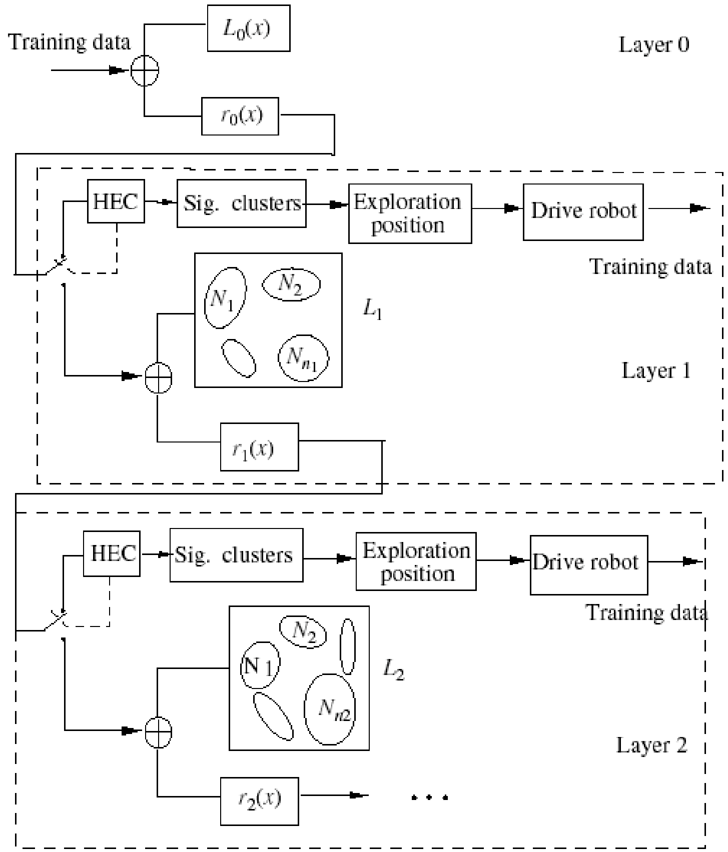
- UKF gain $$K_k = P_{x_z y_z} P_{z_k}^{-1}.$$

Hybridizations

- EKF as a training algorithm
  - Replaces gradient descent
  - Includes an error covariance estimation
    - Gives a measure of the quality of learning
    - Can be used to stop the learning process
  - Applied to Radial Basis Function (RBF) networks
    - State corresponds to the RBF weights

Hybridizations

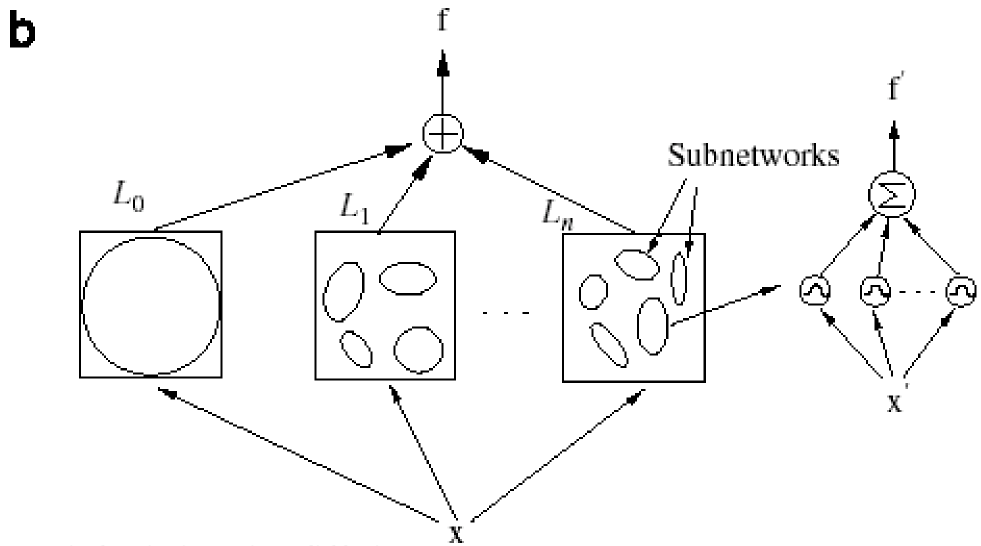
- The error covariance is used to determine regions of the input space which deserve more detailed training



14. Qinggang Meng and Mark Lee. Error-driven active learning in growing radial basis function networks for early robot learning. *Neurocomputing*, 71(7-9):1449–1461, March 2008.

Hybridizations

- The response of the system is an aggregation of the hierarchy of networks



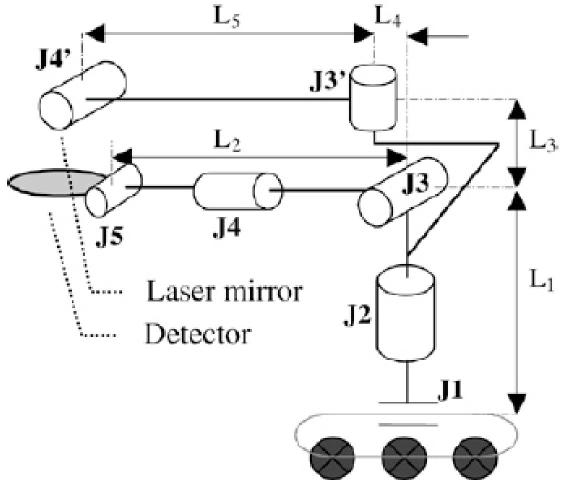
14. Qinggang Meng and Mark Lee. Error-driven active learning in growing radial basis function networks for early robot learning. *Neurocomputing*, 71(7-9):1449–1461, March 2008.

Hybridizations

- Non-linear enhancements of KF
  - Estimation of the covariance in the case of colored noise
  - Estimation of the Kalman gain in non-linear unknown systems
  - Modify the state covariance predictor
  - Perform non-linear estimations of the local behavior of the system

Hybridizations

- Estimation of the Kalman gain for a landmine detection robot SLAM



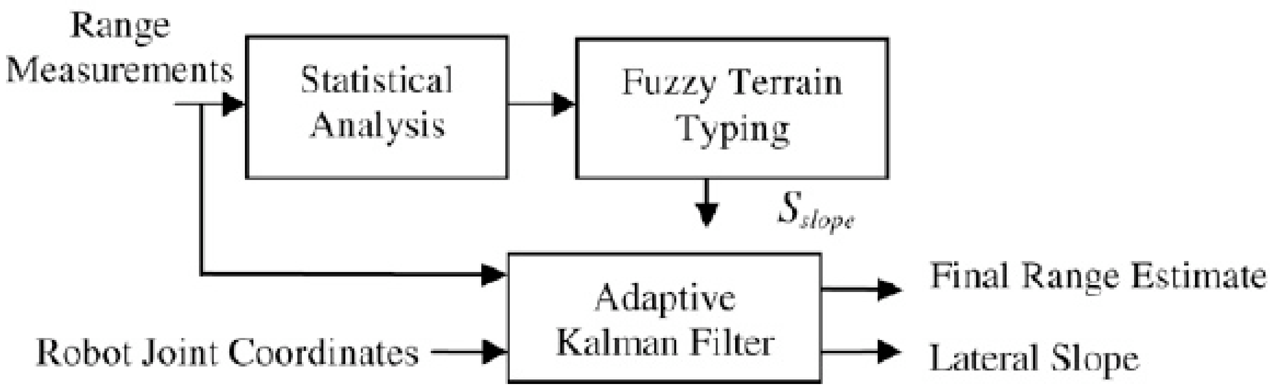
16. Homayoun Najjaran and Andrew Goldenberg. Real-time motion planning of an autonomous mobile manipulator using a fuzzy adaptive kalman filter. *Robotics and Autonomous Systems*, 55(2):96–106, February 2007.

Hybridizations

- The DKF is used to maintain an accurate estimate of the terrain model
  - Integration of range finder readings
- Critical to align the sensor to the terrain slope

Hybridizations

- To avoid non-linear modelling, the Kalman gain is set according to the terrain classification performed by a Takagi-Sugeno system, trained on measurement samples from an in-house terrain database.



16. Homayoun Najjaran and Andrew Goldenberg. Real-time motion planning of an autonomous mobile manipulator using a fuzzy adaptive kalman filter. *Robotics and Autonomous Systems*, 55(2):96–106, February 2007.

Conclusions

- Hybrid KF and Computational Intelligence systems:
  - Training of RBF
  - Providing approximations to components of linear KF
    - Noise covariance
    - Kalman Gain

Further work?

- Integration of KF into Evolution Strategies for improved self-tuning
- Approximation of non-linear observation functions
- Applications to multi-robot systems…