Underwater Vehicle Localisation using Extended Kalman Filter

Miroslav Radojević and Yvan Petillot

miroslav.radojevic@gmail.com

Universitat de Girona · Heriot Watt University · vibot

Contribution

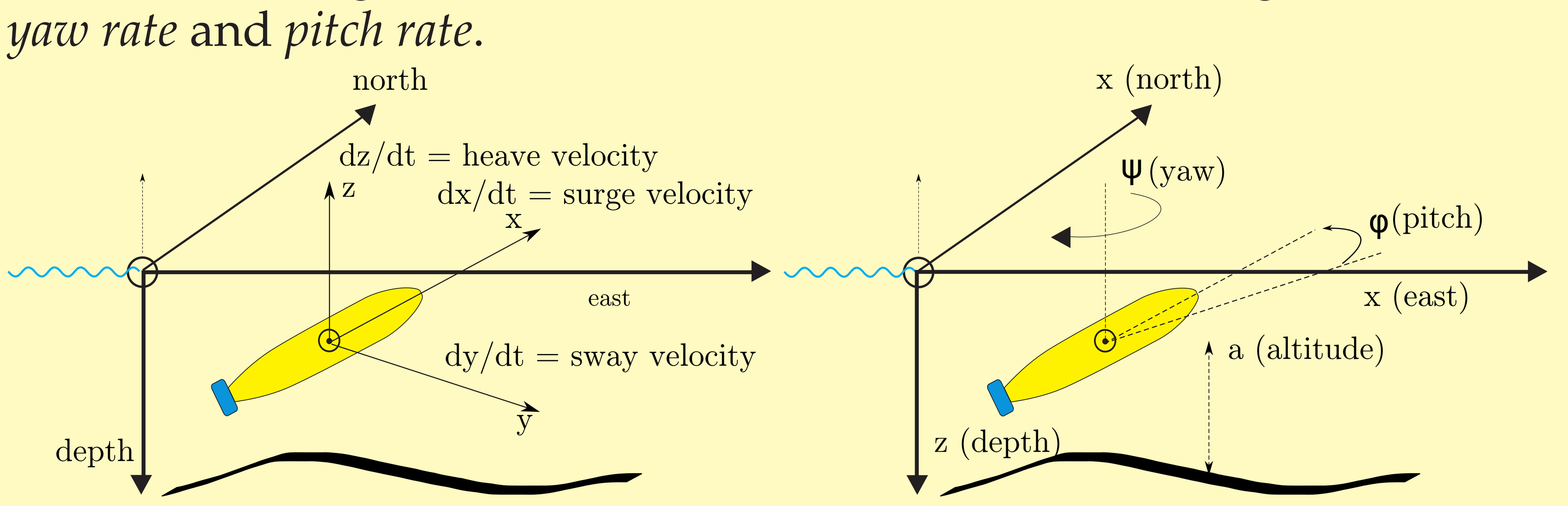
This work presents design and application of an algorithm that accomplishes the localisation of the Ocean Systems Lab's Nessie autonomous underwater vehicle (AUV) fusing together measurements from a number of sensors mounted on it. Well known Extended Kalman Filter (EKF) algorithm was implemented as a solution for robot self-localisation. Benefits of the usage of EKF for obtaining the position and orientation of the robot include more robust estimation which takes into account different types of measurements and the relations between them. Navigation is intended to work in an unstructured environment relying on the odometry and acoustic positioning.

AUV Navigation

Vehicle state is a vector that contains variables relevant for localising the vehicle. Vehicle navigation state describes its position and motion within the environment. Elements of the state vector $\boldsymbol{X}(k)$ are treated as Gaussian Random Variables (GRV).

$$\boldsymbol{X}(k) = \begin{bmatrix} x & y & z & a & u & v & w & \psi & \varphi & \dot{\psi} & \dot{\varphi} \end{bmatrix}^T \tag{1}$$

x, y, z and a take the value of *north*, *east*, *depth* and *altitude*. u, v and w stand for linear velocities: *surge*, *sway* and *heave*, respectfully. The rest of the state vector covers angular values: ψ and φ are used as *yaw* and *pitch*, hence describing the vehicle orientation. $\dot{\psi}$ and $\dot{\varphi}$ are angular velocities: *yaw rate* and *pitch rate*.



5 d.o.f. system model is describing how the state $\boldsymbol{X}(k)$ evolves in time. It is a *constant speed* model [1] that uses previous state $\boldsymbol{X}(k-1)$ corrupted with zero-mean GRV acceleration noise to make a prediction of the next state vector value.

System model

A *constant velocity* nonlinear model was used as a basis for the EKF state transition model.

$$\boldsymbol{X}(k) = f(\boldsymbol{X}(k-1), \boldsymbol{N}(k-1)) \tag{2}$$

$$\begin{bmatrix} x \\ y \\ z \\ a \\ u \\ v \\ w \\ \psi \\ \varphi \\ \dot{\psi} \\ \dot{\varphi} \end{bmatrix}_{(k)} = \begin{bmatrix} x + (uT + \dot{u}\frac{T^2}{2})\cos(\psi)\cos(\varphi) - (vT + \dot{v}\frac{T^2}{2})\sin(\psi)\cos(\varphi) \\ y + (uT + \dot{u}\frac{T^2}{2})\sin(\psi)\cos(\varphi) + (vT + \dot{v}\frac{T^2}{2})\cos(\psi)\cos(\varphi) \\ z + (wT + \dot{w}\frac{T^2}{2})\cos(\varphi) \\ a - (wT + \dot{w}\frac{T^2}{2})\cos(\varphi) \\ u + \dot{u}T \\ v + \dot{v}T \\ w + \dot{w}T \\ \psi + \dot{\psi}T + \ddot{\psi}\frac{T^2}{2} \\ \varphi + \dot{\varphi}T + \ddot{\varphi}\frac{T^2}{2} \\ \dot{\psi} + \ddot{\psi}T \\ \dot{\varphi} + \ddot{\varphi}T \end{bmatrix}_{(k-1)} \tag{3}$$

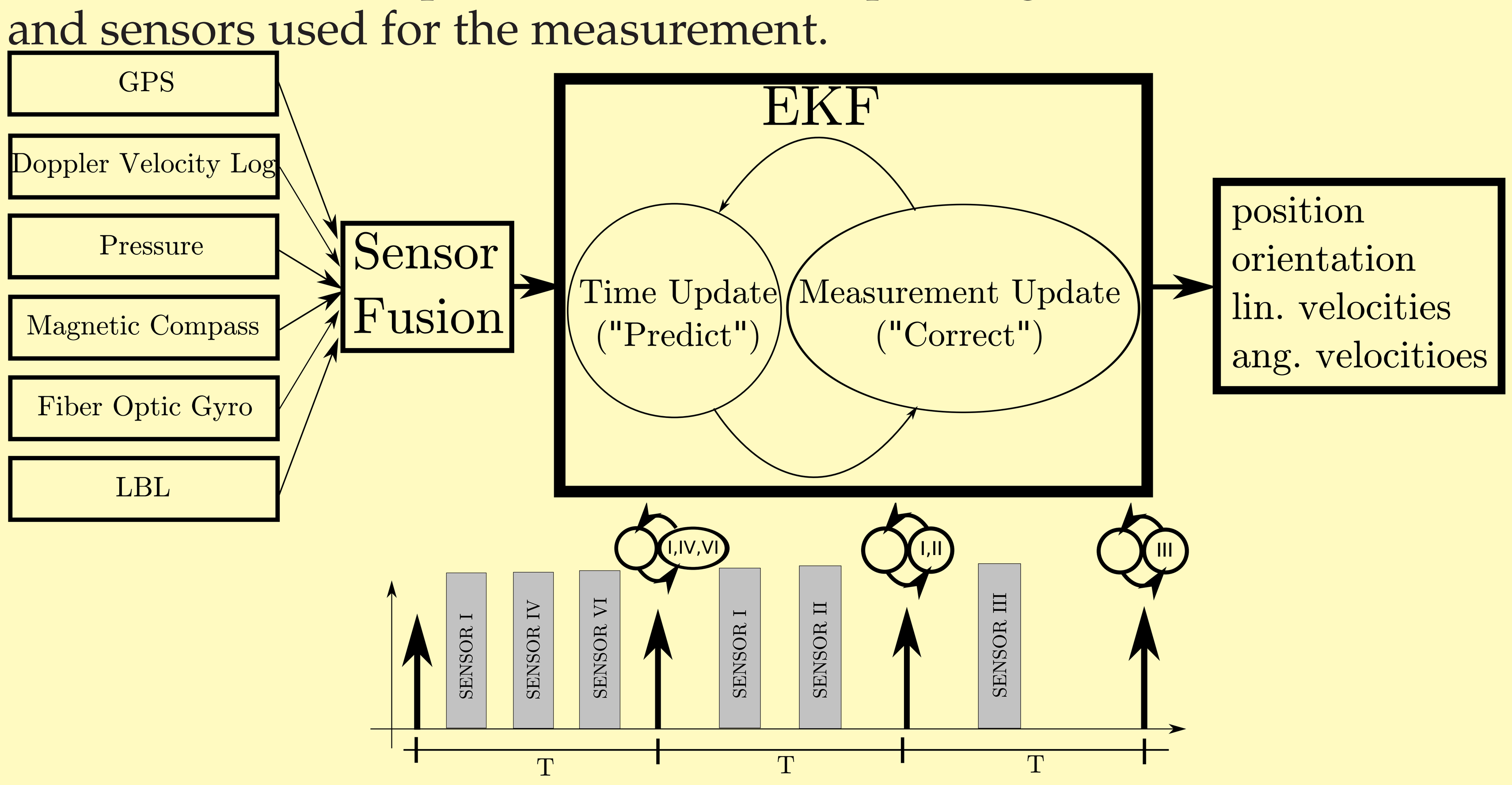
$\boldsymbol{N}(k) = \begin{bmatrix} \dot{u} & \dot{v} & \dot{w} & \ddot{\psi} & \ddot{\varphi} \end{bmatrix}^T$ represents the process noise consisting of linear and angular accelerations. State vector elements are measured directly, hence measurement model $h()$ can be expressed with matrix containing "ones" at particular positions since the measurement relation becomes equality.

$$\boldsymbol{Z}(k) = h(\boldsymbol{X}(k), \boldsymbol{M}(k)) = \boldsymbol{H}\boldsymbol{X}(k \mid k-1) + \boldsymbol{M}(k) \tag{4}$$

$\boldsymbol{M}(k)$ is a zero-mean GRV vector representing the measurement noise. Measurement (observation) and process noise are characterised with diagonal covariation matrices whose elements (σ_N) are given as filtering parameters influencing estimation strategy.

Sensor Fusion

EKF implements sensor fusion. One of the features of navigation process is that sensor measurements are not available all the time. Simply - messages from sensors arrive at different moments and sensors could be unavailable due to different causes. The idea is to take all the gathered information at the moment of filtering and integrate it together in the measurement model (Eq. 5), which varies depending on the measured values and sensors used for the measurement.

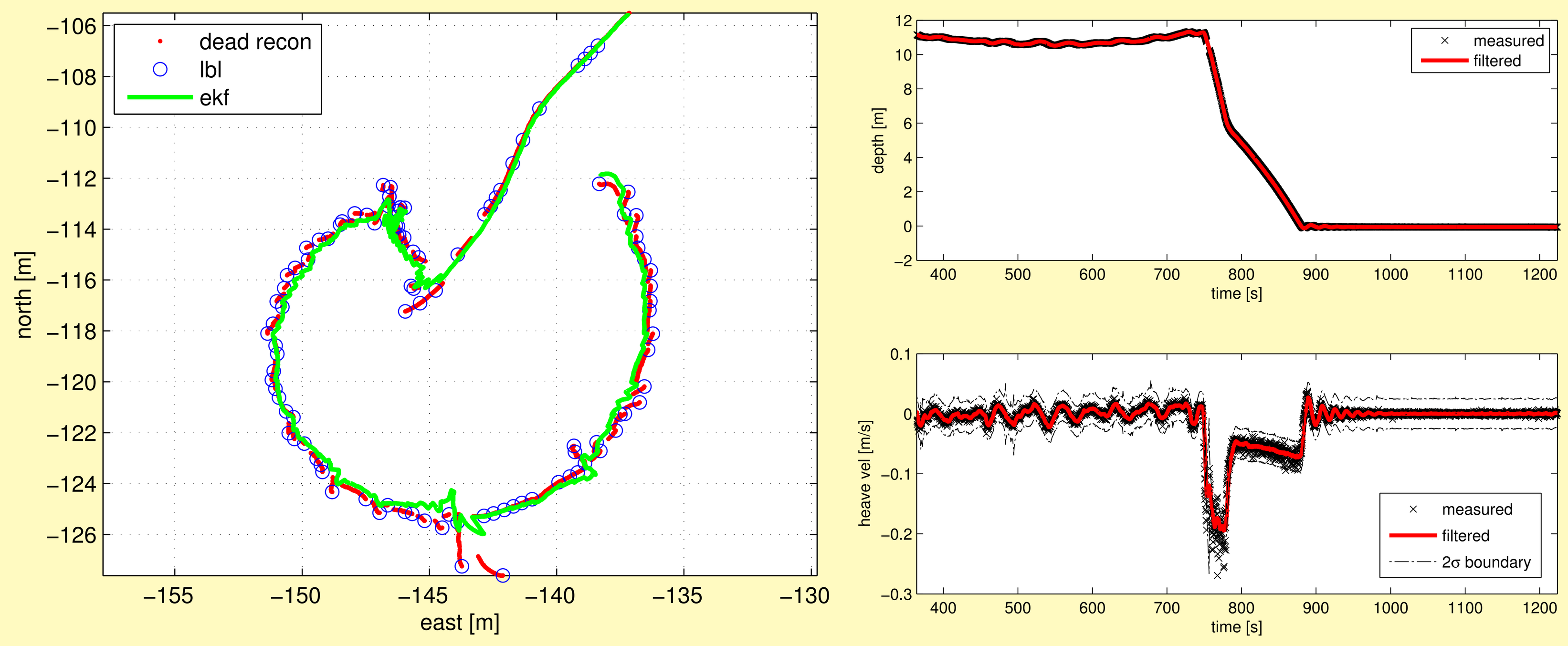


$$\boldsymbol{Z}(k) = \begin{bmatrix} \boldsymbol{Z}_{sen.I} \\ \boldsymbol{Z}_{sen.II} \end{bmatrix}, \boldsymbol{H}(k) = \begin{bmatrix} \boldsymbol{H}_{sen.I} \\ \boldsymbol{H}_{sen.II} \end{bmatrix}, \boldsymbol{R}(k) = \begin{bmatrix} \sigma^2_{sen.I} & 0 \\ 0 & \sigma^2_{sen.II} \end{bmatrix} \tag{5}$$

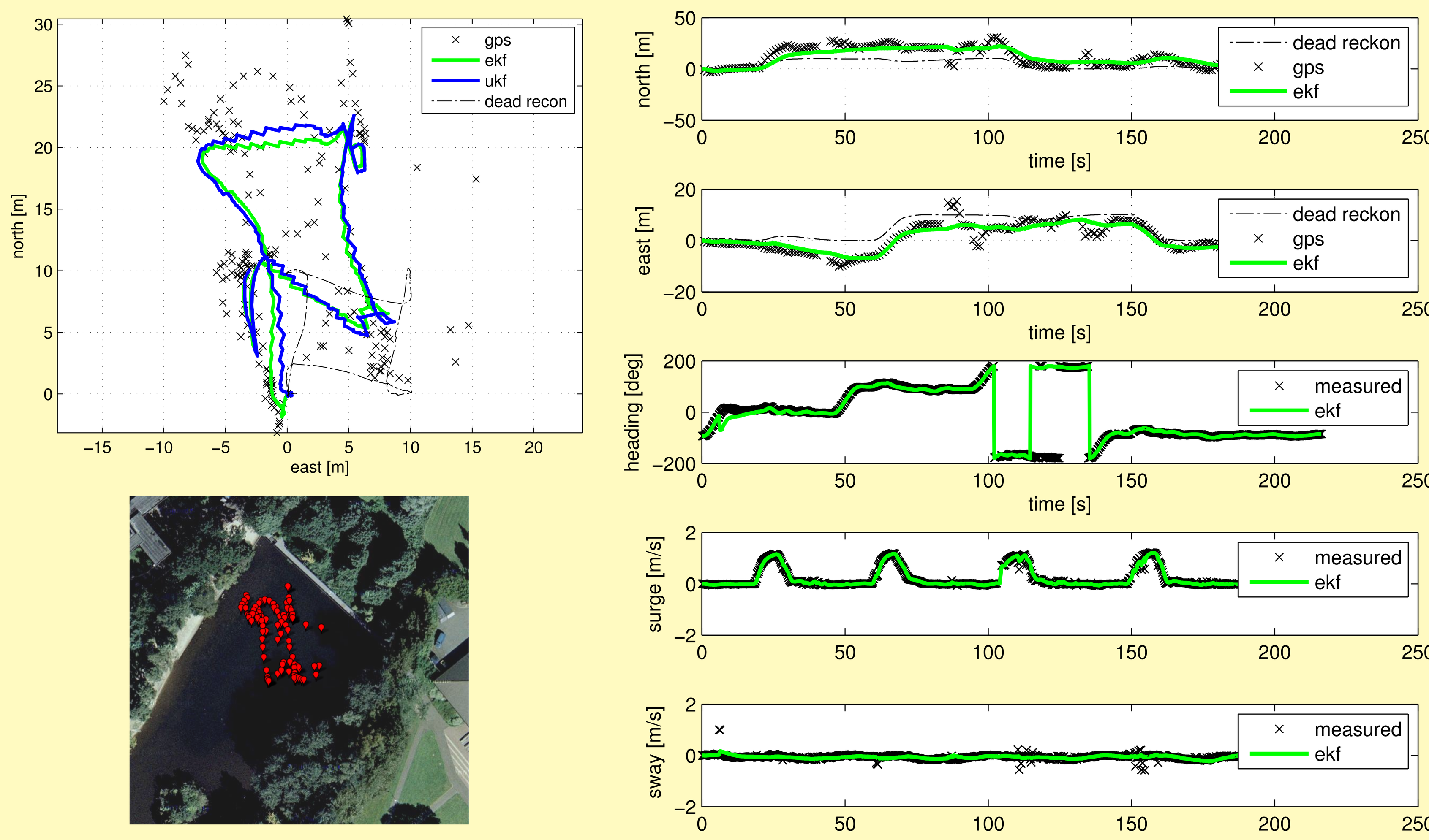
Results

Navigation method was tested on number of real missions with Nessie vehicle:

- Spiral trajectory and surfacing action: localisation results in a smooth path, less prone to drifting than dead reckoning. EKF filters out the absolute position (LBL) outliers. Furthermore, sensor fusion is able to compensate for the missing measurements. Tuning of the filter parameters ($\sigma_{sen.}$) enables giving more or less trust in particular sensor measurements. Question of suitable choice of heading sensor can be treated by adjusting the trust given to *yaw* and *yaw rate* measurement obtained from different devices.



- Rectangular trajectory aided with GPS position measurements: localisation using Unscented Kalman Filter (UKF) was simulated using the real data. Trajectory obtained using UKF tends to be slightly more precise compared with the one obtained using EKF.



EKF improves the localisation performance, even when confined to blend imprecise and sketchy position data from GPS.

References

[1] D. Ribas, P. Ridao, and J. Neira. Underwater Slam for Structured Environments using an Imaging Sonar *Springer Verlag, 2010*

[2] S. Thrun, W. Burgard, and D. Fox. Probabilistic robotics *MIT Press, 2005*