Trajectory Estimation for Ultrashort Baseline Acoustic Positioning Systems

Vincent Ricordel, Sébastien Paris, Jan Opderbecke

HAL Id: hal-00451348
https://hal.science/hal-00451348

Trajectory Estimation for Ultrashort Baseline Acoustic Positioning Systems

Vincent Ricordel †, Sébastien Paris † and Jan Opderbecke ‡

† SIS/Signal - ISITV
Université de Toulon et du Var
Av. Georges Pompidou, BP56
83162 La Valette du Var Cedex (France)
Fax: (+33) 4 94 14 25 98
ricordel@univ-tln.fr, paris@isitv.univ-tln.fr

‡ Institut Français de Recherche pour l'Exploitation de la Mer (IFREMER)
Centre de Toulon, Zone Portuaire du Brégaillon, BP330
83500 La Seyne-sur-Mer Cedex (France)
Fax: (+33) 4 94 87 83 07
opderbecke@ifremer.fr

Abstract—The purpose of the paper is to describe an automatic estimation method of an underwater vehicle trajectory from ultrashort baseline system measurements. The standard Viterbi algorithm is then used to find the most likely trajectory of the dynamic system observed through the noisy measurements with a model of nonconstant speed as state variable. A decoupled trajectory is achieved by considering separately each coordinate of the point. Aberrant measurements are automatically discarded. For very long trajectory, a temporal window is shifted along the data set, and an automatic adjustment of the algorithm parameter is performed.

TABLE OF CONTENTS

I. INTRODUCTION

The French research institute for exploitation of the sea (IFREMER) has developed a new deep-sea positioning system (POSIDONIA) for its underwater vehicles (UV). This ultrashort baseline system (USBL) is composed by an acoustic vessel-born localization system and sensors measuring vessel position, heading and attitude. The latter are used to transform the acoustic UV position into earth related coordinates. The UV is localized down to 6000m. Because of the different sources of measurement error that contribute to the overall positioning error, the accuracy is about 30m and false alarms occur [1] (see figure 7). Our research is then focused on the automatic estimation of the UV trajectory from the prerecorded USBL measurements.

II. METHOD

The standard Viterbi algorithm [2] allows us to find the most likely trajectory of the dynamic system observed through the noisy measurements.

The problem of optimum estimation [3] is briefly formulated with reference to figure 1 as follows; the system state x_k at k th sampling instant is such as:

$$x_k = f(x_{k-1}, \eta_k).$$

x_k depends on the state at the previous sampling instant and on a state noise η_k. The relationship between the measurement and the system state is given by:

$$m_k = h(x_k, n_k),$$

where m_k is the measurement at kth sampling instant corrupted by the noise n_k. At the initialization, the measurement set $M_k = \{m_0, m_1, \ldots, m_k\}$ and $p(x_0)$ the *a priori* probability density function (pdf) of the initial state are given.

The problem is to find the sequence of states or trajectory $X_k = \{x_0, x_1, \ldots, x_k\}$ that maximizes the *a posteriori* conditional pdf $p(X_k|M_k)$. The solution or modal trajectory is such as:

$$\widehat{X}_k \triangleq \arg \max_{X_k} \Big(p(X_k|M_k) \Big).$$

Dynamic programming allows us to construct a wise algorithm. Starting from the fact that:

$$\widehat{X}_k = \arg \max_{x_k} \Big(\arg \max_{X_{k-1}} p(X_k|M_k) \Big),$$

and introducing for each possible value of x_k the quantity:

$$I(x_k) = \max_{X_{k-1}} \Big(p(X_k|M_k) \Big),$$

we end up with the classic recursion:

$$I(x_k) = \max_{x_{k-1}} \Big(p(m_k|x_k) p(x_k|x_{k-1}) I(x_{k-1}) \Big). \quad (1)$$

Let $\hat{x}_{k-1}(x_k)$ the value of x_{k-1} that maximizes (1). The resulting algorithm proceeds in three steps:
• an initialization by defining $I(x_0) = p(x_0)$;

• a forward procedure with, at each sampling instant i, the calculation of $I(x_i)$ from m_i, equation (1) and $\hat{x}_{i-1}(x_i)$ obtained at the previous iteration. The iterative procedure is repeated until the kth instant is reached;
• the modal trajectory is found by tracing backward a stage at a time using the function $\hat{x}_i(\hat{x}_{i+1})$.
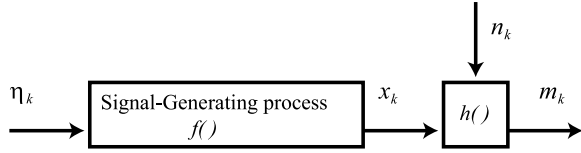

Fig. 1
Optimum estimation problem formulation.

III. STATISTICAL MODELING

A. Monodimensional model

Each position measurement is a three dimensional point, at the kth sampling instant:

$$m_k = [m_{k,x}, m_{k,y}, m_{k,z}]^T .$$

Considering only one dimension, a model with nonconstant speed as state variable is used:

$$x_k \triangleq \left[\begin{array}{c} r_{k,x} \\ v_{k,x} \end{array} \right],$$

where $r_{k,x}$ indicates the position and $v_{k,x}$ the nonconstant speed. It obeys the following state equation:

$$x_k = \left[\begin{array}{cc} 1 & \Delta t_k \\ 0 & 1 \end{array} \right] x_{k-1} + \left[\begin{array}{c} 0 \\ \xi \end{array} \right] \eta_{k,x},$$

where Δt_k is the non constant time interval between the two successive measurements m_{k-1} and m_k, ξ is a constant, and $\eta_{k,x}$ the Gaussian state noise such as:

$$E\Big(\eta_{k,x}\Big) = 0,$$
$$Var\Big(\eta_{k,x}\Big) = \sigma_1^2.$$

The choice of speed depends on Δt_k:
1. if $\Delta t_k \leq 5$s, a choice among nine values is done:

$$\eta_x = [-4, \ldots, -1, 0, 1, \ldots, 4]^T;$$

2. if $\Delta t_k > 5$s, fifteen values are available:

$$\eta_x = [-7, \ldots, -1, 0, 1, \ldots, 7]^T.$$

At the initialization Δt_0 is supposed to be equal to 5s and we choose:

$$x_0 = \left[\begin{array}{cc} 1 & 5 \\ 0 & 1 \end{array} \right] \left[\begin{array}{c} m_{0,x} - 5 \\ 1 \end{array} \right] + \left[\begin{array}{c} 0 \\ \xi \end{array} \right] \eta_{0,x}.$$

B. Transition probabilities

The transition pdf from the x_{k-1} state to the x_k state is defined as follows:

$$p\Big(x_k|x_{k-1}\Big) = K\delta\Big(r_{k,x} - r_{k-1,x} - \Delta t_k v_{k-1,x}\Big) \times$$
$$\exp\left(-\frac{1}{2}\left(\frac{v_{k,x} - v_{k-1,x}}{\xi\sigma_1}\right)^2\right), \quad (2)$$

where K is a normalization constant and $\delta()$ the Kronecker symbol. We denote:

$$p\Big(v_k|v_{k-1}\Big) = K\exp\left(-\frac{1}{2}\left(\frac{v_{k,x} - v_{k-1,x}}{\xi\sigma_1}\right)^2\right). \quad (3)$$

The most probable transitions give an advantage to small accelerations or decelerations, because the expected trajectory must be smooth. At the initialization:

$$p\Big(x_0\Big) = \frac{1}{9}.$$

C. Measurement likelihood

This pdf can be written as the uniform-Gaussian mixture [4] defined as:

$$p\Big(m_{k,x}|x_k\Big) = p\Big(m_{k,x}|r_{k,x}\Big) =$$
$$\frac{1 - P_c}{u_x} + P_c\frac{1}{\sqrt{2\pi}\sigma_x}\exp\left(-\frac{(m_{k,x} - r_{k,x})^2}{2\sigma_x^2}\right), \quad (4)$$

where σ_x is the standard deviation of the measurement noise, u_x the measurement domain and P_c denotes the probability for a measurement to be correct (typically $P_c = 0.7$). When the gap between the measurement $m_{k,x}$ and the position $r_{k,x}$ is small, the likelihood is high (see figure 2). If $|m_{k,x} - r_{k,x}|$ is greater than half of the domain u_x, the likelihood equals zero so the aberrant measurement $m_{k,x}$ is kept away. It implies two consequences:
• a jump appears in the trajectory at the kth instant (see figure 3);
• the sum $(\Delta t_k + \Delta t_{k+1})$ will be used to determinate the next state.
The algorithm manages with consecutive jumps. In such a case the time interval to determinate the current state is the sum of the successive time intervals from the first aberrant measurement.

D. Viterbi recursive equation

Note that because of equations (2), (3) and (4), equation (1) is equivalent to recursively define $I(x_k)$ by:

$$I(x_k) =$$
$$\max_{x_{k-1}}\left(p\Big(m_{k,x}|r_{k,x}\Big)p\Big(v_{k,x}|v_{k-1,x}\Big)I(x_{k-1})\right). \quad (5)$$

In practice we use:

$$p\Big(v_k = \xi i|v_{k-1} = \xi j\Big) = K\exp\left(-\frac{1}{2}\left(\frac{i-j}{\xi\sigma_1}\right)^2\right),$$

with $\xi = 0.125$ and $\sigma_1 = 0.125^{-1}$. When handling with a very large set of data, the calculation of equation (5) suffers
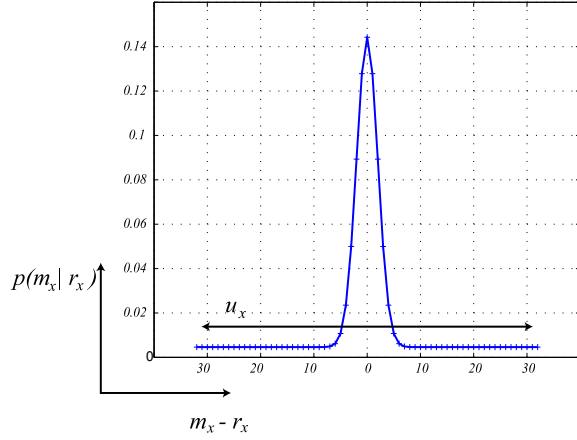
Fig. 2
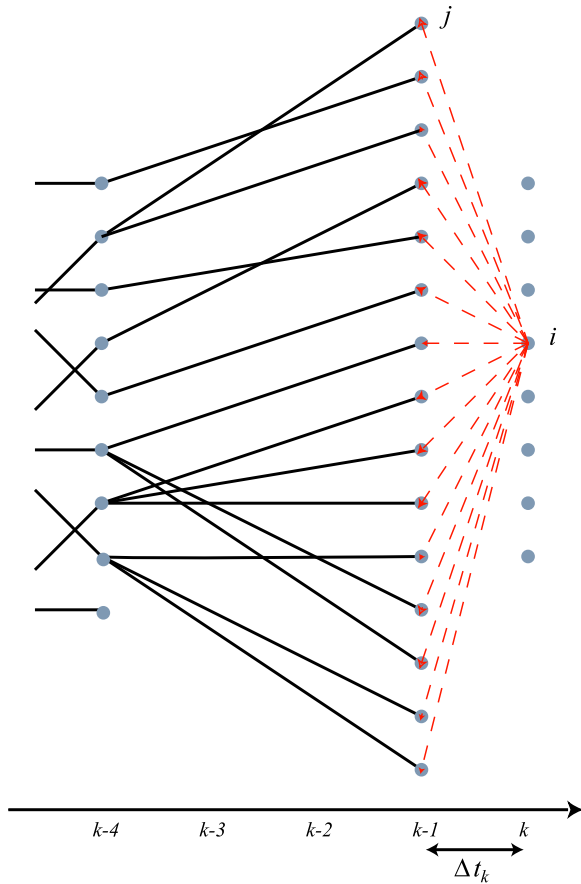Measurement probability with the typical values $u_x = 65$, $\sigma_x = 1$ and $P_c = 0.7$

Fig. 3
Backtracking procedure of the Viterbi algorithm: the points symbolise the system states and the lines the random paths. The measurements at the $(k-3)$ and $(k-2)$ sampling states are aberrant and a jump is done.

from underflow numerical problem. A solution consists in scaling $I(x_k)$ [5]:

$$I^S(x_k) = \frac{\widetilde{I}(x_k)}{\Sigma_{x_k}\widetilde{I}(x_k)},$$

where:

$$\widetilde{I}(x_k) =$$
$$\max_{x_{k-1}}\left(p\Big(m_{k,x}|r_{k,x}\Big)p\Big(v_{k,x}|v_{k-1,x}\Big)I^S(x_{k-1})\right).$$

E. Tridimensional model

The equation of the 3D state model is:

$$x_k \triangleq [r_{k,x}, r_{k,y}, r_{k,z}, v_{k,x}, v_{k,y}, v_{k,z}]^T.$$

The state equation follows:

$$x_k =$$
$$\begin{bmatrix} 1 & 0 & 0 & \Delta t_k & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t_k & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t_k \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \xi \\ \xi \\ \xi \end{bmatrix} \eta_k,$$

and the Gaussian state noise:

$$\eta_k = G\left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{bmatrix}\right).$$

At the initialization we have:

$$x_0 = \begin{bmatrix} 1 & 0 & 0 & 5 & 0 & 0 \\ 0 & 1 & 0 & 0 & 5 & 0 \\ 0 & 0 & 1 & 0 & 0 & 5 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} m_{0,x} - 5 \\ m_{0,y} - 5 \\ m_{0,z} - 1.25 \\ 1 \\ 1 \\ 0.25 \end{bmatrix}$$

$$+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ \xi \\ \xi \\ \xi \end{bmatrix}\eta_{0,x}.$$

A decoupled 3D trajectory is achieved: namely at each sampling time the three coordinates of the point r_k are estimated separately using for each of them the monodimensional model described previously. Nevertheless, if a jump occurs when estimating a coordinate, this jump is forced on the two other point coordinates. The algorithm parameter set to adjust is finally:

$$S = \{\sigma_x, \sigma_y, \sigma_z, u_x, u_y, u_z\}.$$

IV. TEMPORAL WINDOWS AND OVERLAPPING

Because of the very large amount of data and in order to reduce the computing cost, it is preferable to estimate a piece-wise trajectory instead of process straight one only long track [6]. Figure (4) shows how a window of T measurements is shifted along the data set (typically $T = 50$).
As figure (5) illustrates it, after the backtracking stage of the

Viterbi algorithm, the possible tracks differ only in the last sampling instants of a temporal window. We take advantage of this propriety to overlap the trajectories, considering a window:

- the point, from which the 3D trajectory is split, is detected;
- all of the possible tracks from this point are stored;
- the next trajectory estimation is initialized using this group of tracks.
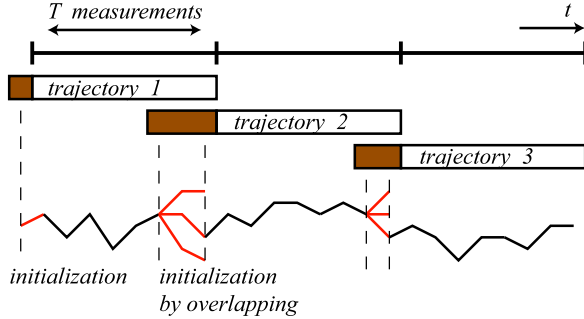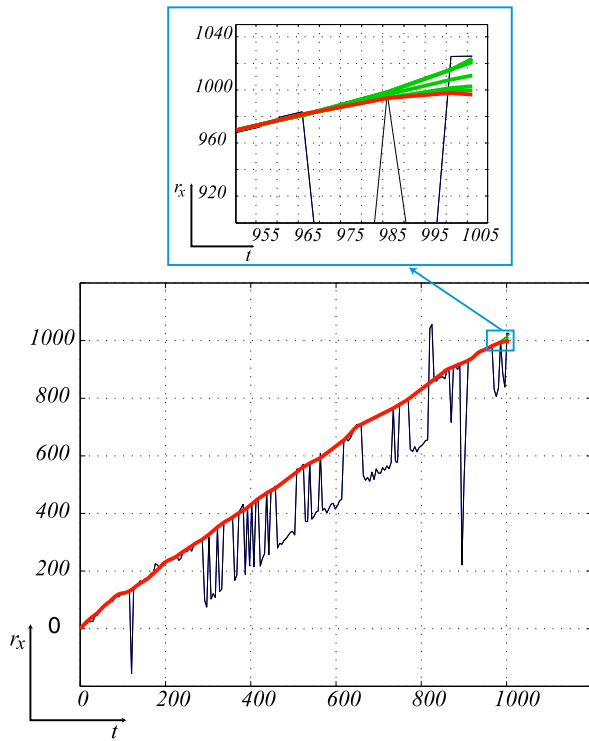

Fig. 4
Temporal windows and overlapping.

Fig. 5
Trajectory estimation in one dimension with 200 measurements ($S = \{\sigma_x = 2, u_x = 60\}$). A zoom in the end of the trajectory is shown. The estimated track is thick.

V. AUTOMATIC ADJUSTMENT OF THE PARAMETERS

The result of figures (5) and (8), where the measurements are very noisy, shows the robustness of our method but the estimated tracks are short. When processing a very large set of data, it is necessary to adjust locally the parameter set S of each window in order to avoid any break in the track. To overcome this drawback an heuristic process is designed that aims to adjust automatically the parameter set S_l of each lth window. Figure 6 illustrates the algorithm principles:

1. at the beginning a set S^* of typical parameters is chosen and:
$$S_l = S^*, \forall l.$$

2. when estimating sequentially the trajectory, a break occurs in the lth window (namely all the measurements of the window are aberrant), considering the previous $(l-1)$th window:

(a) the number of jumps for each dimension is counted;

(b) S_{l-1} is modified by reducing σ_x or/and σ_y or/and σ_z corresponding to the coordinate(s) with the jump maximum number. The goal is to penalize the measurements spread.

(c) the trajectory of the $(l-1)$th window is estimated again with the new set of parameters S_{l-1}.

3. the trajectory estimation of the lth window is achieved with S_l. If a break occurs again, go to 2 and continue to modify S_{l-1}.

This algorithm for the automatic parameter adjustment is not expensive because in practice a break occurs rarely and, in this case, each iteration implies the new trajectory estimation for only two temporal windows.

The experimental result shown on figure 7 illustrates the efficiency and robustness of our algorithm: the measurement set contains 1600 samples, and the whole trajectory is automatically estimated for different choices of the temporal window size T (for instance $T = 30, 50$ or 100 measurements). We used:

$$S^* = \{\sigma_x = \sigma_y = 2, \sigma_z = 2.5,$$
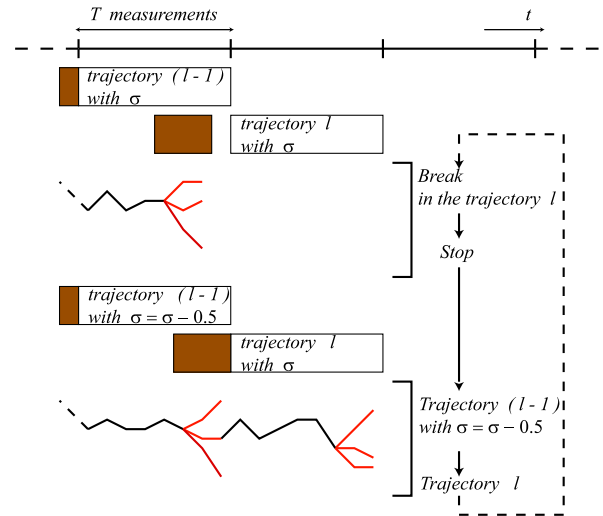$$u_x = u_y = 60, u_z = 30\}.$$

Fig. 6
Process for an automatic adjustment of the parameters.

VI. CONCLUSION

The work presented in this paper concerns the problem of estimating of an underwater vehicle trajectory from ultrashort baseline system measurements.
Using the Viterbi algorithm, we have constructed an original

method with a nonconstant speed state model, a jump possibility in the track, a scaled metric, temporal windows with overlapping, and an automatic adjustment of the parameters. The algorithm is robust w.r.t false alarms and the amount of calculus is real time compatible.

REFERENCES

[1] J. Opderbecke, "At-sea calibration of a usbl underwater vehicule positioning system," in *Proc. of the IEEE Oceans'97*. Halifax, 1997.

[2] G.D. Forney, "The viterbi algorithm," *Proc. of the IEEE*, vol. 61, no. 3, pp. 268–278, March 1973.

[3] R.E. Larson and J. Peschon, "A dynamic programming approach to trajectory estimations," *IEEE Transactions on Automatic Control*, pp. 537–540, July 1966.

[4] C. Jauffret and Y. Bar-Shalom, "Target motion analysis in the presence of false alarms," *Journal of Acoustical Society of America*, vol. 94, no. 3, September 1993.

[5] S. Paris, *Extraction automatique de pistes fréquentielles en sonar passif par chaînes de Markov cachées*, Ph.D. thesis, Université de Toulon et du Var, November 2000.

[6] A. Kriouile, J.F. Mari, and J.P. Haton, "Some improvements in speech recognition algorithms based on hmm," in *Proc. of International Conference on Acoustics, Speech, and Signal Processing ICASSP*. Albuquerque, USA, April 1990, pp. 545–548.

Vincent Ricordel was born in 1967. He received the Ph.D. degree in Signal Processing and Telecommunication from the University of Rennes, France, in 1996. He now holds an assistant professor position at the University of Toulon, France. His research interests include video coding, trajectory estimation and image sequence analysis.

Sébastien Paris was born in 1973. He received the Ph.D. in Optic, Signal and Image from the University of Toulon, France, in 2000. He now holds an temporary assistant professor position at the University of Toulon, France. His research interests include statistical signal processing, estimation and detection using HMM.
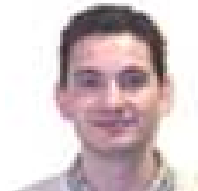
Jan Opderbecke was born in 1963. He received the Diploma in electrical engineering from the Technical University of Braunschweig, Germany, in 1990, and the PhD at Ecole Normale Supérieure de Cachan, France, in 1994. He now holds a research engineer position at the french institute for research on the exploitation of the sea (Ifremer) in Toulon, France. His research interests include signal and image processing applications related to underwater vehicle navigation and seabed mapping. Among his work is the fully automatic integrated navigation system of Ifremer's manned submersible Nautile and the visual mosaicking system of the remote operated vehicle Victor 6000.
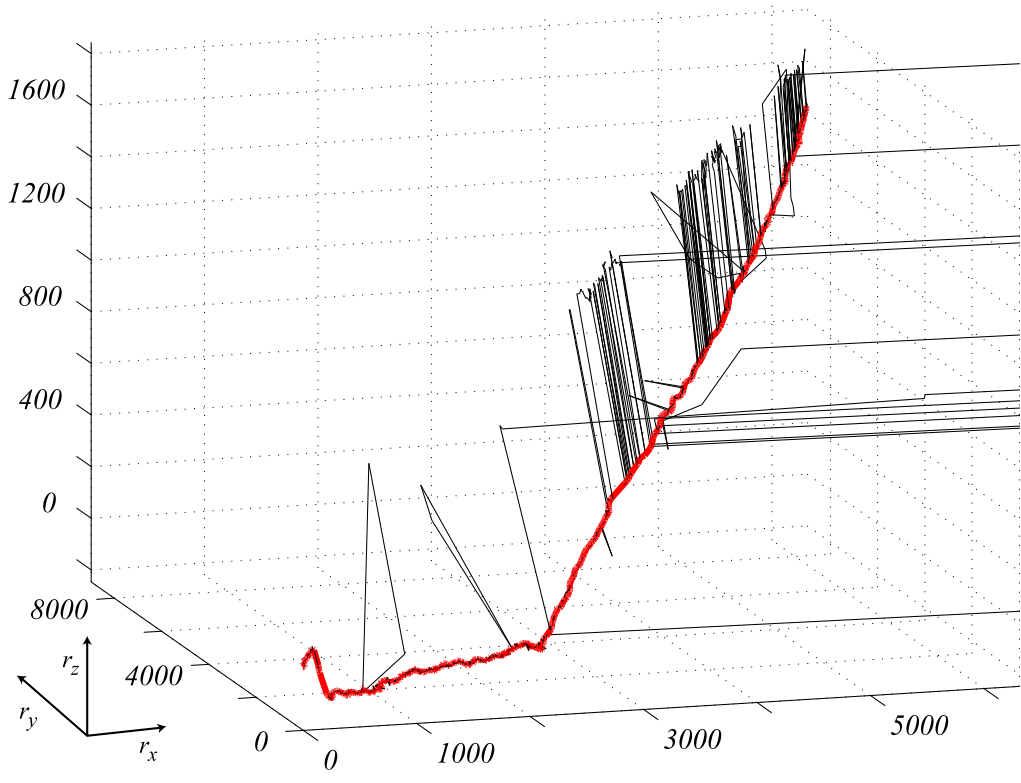
Fig. 7
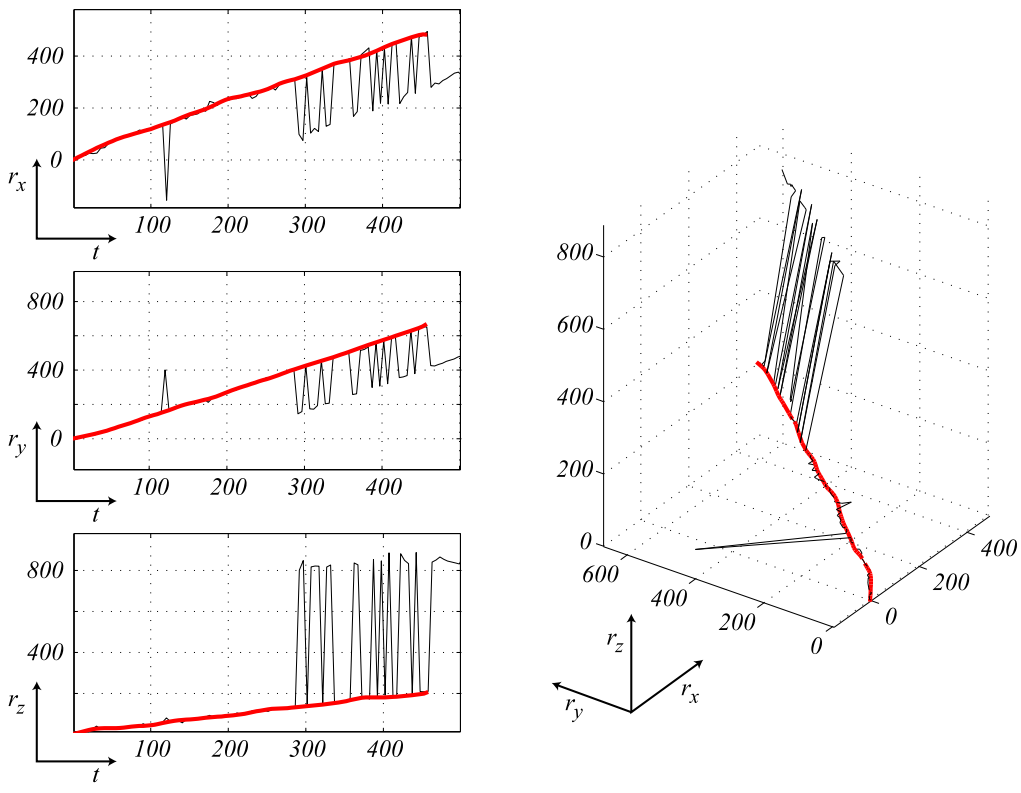The data from the underwater vehicle. The estimated track is thick.

Fig. 8
Trajectory estimation in 3 dimensions with 100 measurements ($S = \{\sigma_x = \sigma_y = 1.5, \sigma_z = 2, u_x = 60, u_y = 65, u_z = 30\}$).
Two temporal windows of $T = 50$ measurements are used. The estimated track is thick.