Sea glider navigation around a circle using distance measurements to a drifting acoustic source

Jan SLIWKA
Doctor - Engineer in underwater robotics
ENSTA Bretagne
Our activities in applied interval analysis to underwater robotics
SAUC’E robot competition

- SAUC-E (Student Autonomous Underwater Challenge - Europe)
Robot Swarms

- 4 CISCREA robots

[Diagram of robot swarms with SLAM and AUV connections]
ENSTA Bretagne’s Glider

A campaign of several months under the polar ice cape
Problems to solve: No GPS

- Localization of the vehicle?

Diagram:
- AITP
- satellite communication
- GPS
- ICE
- profiler for CTD data acquisition
- RAFOS module
- glider
- range measurement
- sound travel time \times \text{speed of sound}
Classical approach: (lon, lat) localization

Acoustic sources

Current

Glider $G(x, y, z, \theta, \nu)$

$S_3(x_3, y_3)$

$S_1$ $S_2$

$d_3 = \Delta t_3 \times c$

RAFOS range data

$S_4$
Problems to solve: Sources are drifting
Other problems

- The pings are rare (to economize the energy) → little redundancy of data
- The readings of the compass are poor in the North Pole
- ...
Solution

- Split the localization (longitude, latitude) and navigation

  Master buoy for circular navigation

  pings every 20 minutes

  Annex buoys for (lon, lat) localization in post processing

  pings every 2H

  50km

  invariance through rotation of the problem
Plan of the presentation

- Navigation around a single acoustic beacon using set membership methods
  - General idea about the control
  - Setting up the equations
  - Setting up the problem as a set inversion problem
  - Estimate error and its derivative using robust regression
  - Simulation results of the PID control loop
- Results of (lon,lat) localization also using set membership methods
- Conclusion
Single acoustic beacon navigation
The controller of the glider

Master buoy for circular navigation

pings every 20 minutes

Video
State equations

- Circle following case

\[
\begin{align*}
\dot{d} &= v_c \sin \beta + v_g \sin \delta \\
\dot{\delta} &= \arctan \left( \frac{v_c \cos \beta + v_g \cos \delta}{d} \right) + \omega \\
\dot{\psi} &= \omega.
\end{align*}
\]

Big radius (>5km) with respect to glider’s speed (1km/h)
State equations: approximation

- Approximation by a line

\[
\begin{align*}
\dot{d} &= v_c \sin \beta + v_g \sin \delta \\
\dot{\delta} &= \omega \\
\dot{\psi} &= \omega.
\end{align*}
\]

Big radius (>5km) with respect to glider’s speed (1km/h)
Discrete regulation

- **Step 1**: Make a range measurement
- **Step 2**: Compute heading correction
- **Step 3**: Correct the heading (using a compass or gyroscope) – this step takes several minutes -
- **Step 4**: Go straight until the next measurement (~20 minutes)
- **Step 5**: Go to Step 1
The error

\[ e_k = d_k - r \]

The PID controller for the rotation speed

\[
\omega_k = K_1 \sum_{i=0}^{k} e_i + K_2 e_k + K_3 \dot{e}_k
\]

Filtering a noisy signal with outliers

**Derivative** of a noisy signal with outliers
Method 1: state estimation
State estimation

- We estimate all the variables which can be estimated

\[ d_n, \delta_n, v_c \sin \beta \]
Discrete form of the state equations

The state equations for the case of line following are

\[
\begin{align*}
\dot{d} &= v_c \sin \beta + v_g \sin \delta \\
\dot{\delta} &= \omega
\end{align*}
\]

The discrete form of those state equations can be written

\[
\begin{align*}
d_{k+1} &= (v_c \sin \beta + v_g \sin \delta_{k+1}) \Delta t + d_k \\
\delta_{k+1} &= \delta_k + u_k + w_k \\
\tilde{d}_k &= d_k + \tilde{w}_k
\end{align*}
\]
Set up the problem as a set inversion

The **discrete** form of those state equations can be written

\[
\begin{align*}
d_{k+1} &= (v_c \sin \beta + v_g \sin \delta_{k+1}) \Delta t + d_k \\
\delta_{k+1} &= \delta_k + u_k + w_k \\
\tilde{d}_k &= d_k + \tilde{w}_k
\end{align*}
\]
Set up the problem as a set inversion

The discrete form of those state equations can be written:

\[
\begin{align*}
    d_{k+1} &= (v_c \sin \beta + v_g \sin \delta_{k+1}) \Delta t + d_k \\
    \delta_{k+1} &= \delta_k + u_k + w_k \\
    \tilde{d}_k &= d_k + \tilde{w}_k
\end{align*}
\]

For each time step \( n \) we search for:

\[
d_n, \delta_n, v_c \sin \beta
\]
Set up the problem as a set inversion

The \textit{discrete} form of those state equations can be written:

\begin{equation}
\begin{align*}
    d_{k+1} &= (v_c \sin \beta + v_g \sin \delta_{k+1}) \Delta t + d_k \\
    \delta_{k+1} &= \delta_k + u_k + w_k \\
    \tilde{d}_k &= d_k + \tilde{w}_k
\end{align*}
\end{equation}

For each time step \( n \) we search for:

\[ d_n, \delta_n, v_c \sin \beta \]

We obtain equations in the form of:

\begin{equation}
    d_n - kv_c \Delta t \sin \beta - \sum_{i=0}^{k-1} v_g \Delta t \sin \left( \delta_n - \sum_{j=1}^{i} (u_{n-j} + w_{n-j}) \right) + \tilde{w}_{n-k} = \tilde{d}_{n-k}
\end{equation}
Set up the problem as a set inversion

The **discrete** form of those state equations can be written

\[ d_{k+1} = (v_c \sin \beta + v_g \sin \delta_{k+1}) \Delta t + d_k \]
\[ \delta_{k+1} = \delta_k + u_k + w_k \]
\[ \tilde{d}_k = d_k + \tilde{w}_k \]

For each time step \( n \) we search for

\[ d_n, \delta_n, v_c \sin \beta \]

We obtain equations in the form of

\[ d_n - kv_c \Delta t \sin \beta - \sum_{i=0}^{k-1} v_g \Delta t \sin \left( \delta_n - \sum_{j=1}^{i} (u_{n-j} + w_{n-j}) \right) + \tilde{w}_{n-k} = \tilde{d}_{n-k} \]

\[ f(d_n, \delta_n, v_c \sin \beta) = \begin{bmatrix} \tilde{d}_n \\ \vdots \\ \tilde{d}_{n-N} \end{bmatrix} \]

3D Multi-occurrence

...
Set up the problem as a set inversion

The **discrete** form of those state equations can be written

\[
\begin{align*}
  d_{k+1} &= (v_c \sin \beta + v_g \sin \delta_{k+1}) \Delta t + d_k \\
  \delta_{k+1} &= \delta_k + u_k + w_k \\
  \tilde{d}_k &= d_k + \tilde{w}_k
\end{align*}
\]

For each time step \( n \) we search for

\[
d_n, \tilde{\delta}_n, v_c \sin \beta
\]

We obtain equations in the form of

\[
d_n - kv_c \Delta t \sin \beta - \sum_{i=0}^{k-1} v_g \Delta t \sin \left( \delta_n - \sum_{j=1}^i (u_{n-j} + w_{n-j}) \right) + \tilde{w}_{n-k} = \tilde{d}_{n-k}
\]

\[
\frac{d_n - d_{n-1}}{\Delta t} = v_c \sin \beta + v_g \sin \delta_n
\]

\[
f(d_n, \delta_n, v_c \sin \beta) = \begin{pmatrix}
  \tilde{d}_n \\
  \vdots \\
  \tilde{d}_{n-N}
\end{pmatrix}
\]

3D Multi-occurrence

...
Waiting for IBEX 2

- Constraint in the form

\[ d_n - kv_c \Delta t \sin \beta - \sum_{i=0}^{k-1} V_g \Delta t \sin \left( \delta_n - \sum_{j=1}^{i} (U_{n-j} + W_{n-j}) \right) + \tilde{w}_{n-k} = \tilde{d}_{n-k} \]
Simple

Method 2: linear regression
Numerical differentiation: linear regression

Glider moving in a strait line

Least squares?
Robust linear regression

Find the lines which pass through a maximum number of intervals $[y_i]$.
Simulation results

- Go straight line
- Every N measurements (N=7 -> 2 hours) perform PID control
- → use frequent measurements!
Simulation result

- 20% outliers
- 20% missing data
- 1 measurement every 20 minutes
- speed 0.3m/s
- north current 0.15m/s
- circle radius 10km
Simulation result

- 20% outliers
- 20% missing data
- 1 measurement every 20 minutes
- speed 0.3m/s
- north current 0.15m/s
- circle radius 10km
One year navigation

20km
The spiral
Offline longitude / latitude localization
Localization using acoustic beacons

Localization by trilateration

Dataset from AWI (Allemagne) taken in the Fram strait

Set membership methods for underwater robotics
Localization using acoustic beacons
Conclusion

- A system for oceanic survey
  - A real-time method for navigation
  - A method for data geo-localization performing offline computation
- Using set membership methods allow to handle outliers and non-linear equations
- Perspectives: decrease the rate of measurements as much as possible
Questions?

Presented by
Jan SLIWKA
jan.sliwka@ensta-bretagne.fr