River mapping using an Autonomous Surface Vehicle equipped with LiDAR

September 2019

Filip Kronström
Lund University
Lund, Sweden
elt15fkr@student.lu.se

Richard M. Murray
Control and Dynamical Systems
California Institute od Technology
Pasadena, CA
murray@cds.caltech.edu

Alistair T. Hayden
Geology and Planetary Sciences
California Institute od Technology
Pasadena, CA
ahayden@caltech.edu

***Abstract*—River floodings are one of the most costly natural hazards today. With more knowledge and data about rivers these expenses can be reduced. However, the data collection is a time consuming procedure and an automated way of doing this would save both time and other resources. This paper presents the work on an Autonomous Surface Vehicle (ASV) used for river mapping. With an Acoustic Doppler Current Profiler (ADCP) the depth of the river is measured together with the velocity of the water currents. This makes it possible to create depth profiles as well as calculating the river discharge. The main improvement presented is a new controller which uses LiDAR and ADCP measurements to automate transects as well as keeping a constant distance to the shore while going upstream. This allows new data to be collected and simplifies the work in the field. The implementation has been tested in Kern River, California, with current velocities of around 1.4m/s.**

Fig. 1. Autonomous Surface Vehicle and ADCP.

I. Introduction

Over the past ten years the costs following river flooding has been an annual average of 50 billion dollars. This makes it one of the most costly natural hazards [1]. The geometry of rivers partly determines where flooding will appear. It is thereby, with the right data, possible to predict where and when floodings might occur. As of today this data is difficult to come by. It is often either outdated, which is the case for the Mississippi river where the data is from the 1970s or, for smaller rivers, non-existing [2].

The necessary data could be collected in different ways. It could be done either using unmanned or manned vehicles. The latter would obviously require people operating the mission. This makes it a less flexible and more expensive option. Autonomous Underwater Vehicles (AUV) are already widely used for exploration of oceans and lakes [3]. For mapping of rivers both Unmanned Aerial Vehicles (UAV) equipped with laser scanners [4] and robotics systems dragging sensors across rivers [5] has been used. There are positives and negatives with both approaches. The UAV can map big areas, but information is lost as a result of the airborne equipment. The robotics system can produce very accurate measurements, but it is fixed in one location. Hence, either several systems are needed or the setup needs to be moved to cover larger parts of a river.

Lee and Wu developed a method allowing an Autonomous Surface Vehicle (ASV) equipped with an Acoustic Doppler Current Profiler (ADCP) to generate river profiles. This was done by traversing the river in straight lines and creating 3D lattices for depth and velocity of the water. This approach allows collection of data both over large sections of a river and over time which makes it possible to see how river profiles changes. As the velocity of the water is measured the river discharge can be calculated by integrating these velocities [6]. The ASV setup implemented needed some work to be able to run the boat. It required either a very exact map with GPS coordinates or that the user manually drove the boat to find reference coordinates.

The work presented in this paper builds on the prior work done by Lee and Wu. To increase the usability of the vehicle a Light Detection and Ranging (LiDAR) sensor is added to the ASV. Thanks to the LiDAR the path planning can be done online to conduct a specified test. This paper presents the implementation of the new controller along with results from field tests in Kern River, California. The paper is structured as follows: The controller is explained in Section II. In Section III the calculations for the LiDAR is presented. Section IV explains the controller program which uses new sensors for navigation. The results from deployment in Kern River is shown in Section V.

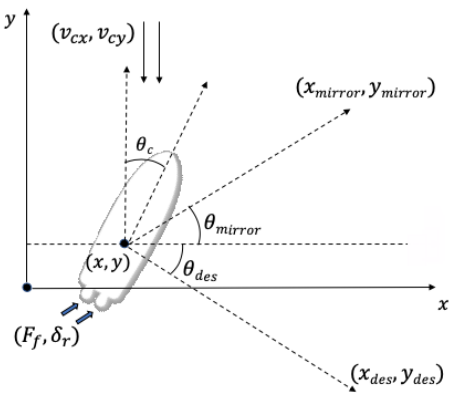

Fig. 2. Angle to current and mirroring of points downstream.

II. Controller

Two different controllers were used by [6] to control the ASV. One for transects perpendicular to the current and one line of sight (LOS) controller to travel along the river.

A. New Line of sight controller

To be able to follow a more general trajectory the LOS controller is changed. The new controller should be able to travel both up- and downstream as well as perpendicular to the current. It should also be able to hold the position. To accomplish this two PI-controllers are used.

In the new controller the direction of the current, obtained from the ADCP, is used. To keep as much control over the position of the vehicle as possible it is always pointed somewhat upstream. By varying the thrust force it is still possible to go both up- and downstream. If the point the vehicle is travelling to is downstream the boat will float with the current towards it. This is done by mirroring the reference point in a line through the boat perpendicular to the current, shown in Fig. 2, and using less thrust. Hence the boat will steer towards the point as if it was upstream, but float with the current downstream. If the current velocity measured by the ADCP is below a certain threshold, C_{th}, it is ignored and the controller works as a regular PI controller. As the current angle is derived from the current in the the x and y direction it is uncertain at low speeds. How the reference point is chosen is shown in Algorithm 1.

The rudder controller tries to angle the vehicle towards the desired point (x_{des}, y_{des}) and the control signal, u_r is calculated according to the following equations:

$$\begin{aligned} e_\theta(t) &= \theta_{des}(t) - \theta_{course}(t) \\ u_r(t) &= K_r(e_\theta(t) + \frac{1}{T_{i,r}} \int_0^t e_\theta(\tau) d\tau) \end{aligned} \tag{1}$$

To do this it looks at the direction in which the vehicle is travelling, obtained from GPS data. In a river with currents this is necessarily not the same as the orientation of the boat.

Algorithm 1 Approach to chose reference for controller

if Angle to current, $\theta_c > \pi/3$ & Current $> C_{th}$ **then**
  Desired angle is chosen to turn boat upstream and desired velocity is predefined.
else if Destination reached **then**
  Desired angle is chosen to turn boat upstream and desired velocity is zero.
else
  if Destination is downstream & Current $> C_{th}$ **then**
    Desired angle is calculated to the mirrored point upstream and desired velocity is negative.
  else
    Desired angle is the angle towards the destination and desired velocity is predefined.
  end if
end if
Control outputs are calculated according to (1) and (2) based on desired angle and velocity.

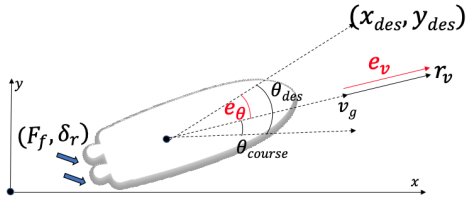

Fig. 3. The angles used for rudder control along with the vehicles coordinate frame and velocity error used for thrust control.

The controller error, e_θ, is generated as the difference between the desired angle, θ_{des}, and the direction in which the ASV is moving, θ_{course}. θ_{course} is obtained by differentiating the GPS velocities in the $x-$ and $y-$direction. The angles are visualized in Fig. 3. At low speeds the actual heading of the vehicle is used as the differentiated GPS signal turns uncertain.

The thrust control, u_t is calculated in a similar fashion as the rudder angle, using a PI-controller described by:

$$\begin{aligned} \begin{bmatrix} v_{r,x} \\ v_{r,y} \end{bmatrix} &= \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} v_{g,x} \\ v_{g,y} \end{bmatrix} \\ e_v(t) &= r_v(t) - v_{r,x}(t) \\ u_t(t) &= K_t(e_v(t) + \frac{1}{T_{i,t}} \int_0^t e_v(\tau) d\tau) \end{aligned} \tag{2}$$

The error, u_r, is now generated as the difference between the velocity along the vehicles x-axis, $v_{r,x}$, shown in Fig. 3 and a reference speed. As the ADCP samples at a fixed rate the reference speed determines the resolution in these measurements. The thrust controller is summarized in the equations below:

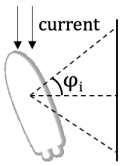

Fig. 4. Angles for distance measurement.

B. Transect controller

For the transects the controller from [6] is used. It consists of two P controllers. The thrust force is calculated from the position error compared to the transect line as shown in the equations below:

$$\begin{aligned} e_y &= y_{line} - y \\ u_t &= K_t * e_y \end{aligned} \tag{3}$$

Similarly the rudder angle is controller based on the speed along the line according to:

$$\begin{aligned} e_{\dot{x}} &= r_{\dot{x}} - \dot{x} \\ \delta r &= K_\delta * e_{\dot{x}} \end{aligned} \tag{4}$$

III. LiDAR measurements

The LiDAR used on the boat is an Ouster OS1-16 which is a 16 beam 3D LiDAR. The LiDAR is used to keep track of the shorelines. The distance to the shoreline can be used to stay in the middle of the river, at a certain distance from the shore or to determine when to turn during transects.

The 16 beams from the LiDAR gives a 3D pointcloud. A 2D representation of the 3D data is saved as the polar coordinates, (r, θ), of the closest point within a certain height above and below the LiDAR's position. When the distance to the shore is calculated it is assumed that the shore is parallel to the current as shown in Fig. 4. Hence the shore at each side will be found at the angles perpendicular to the current. The distance is then calculated as the mean distance based on a specified number of points as shown below:

$$\begin{aligned} d &= \frac{\sum_{i=-N}^{N} r_i \cos \phi_i}{2N+1} \\ \phi_i &= \theta_i - \theta_c^{\perp} \end{aligned} \tag{5}$$

Where ϕ_i is the angle to the middle point of the LiDAR measurements.

IV. New controller program

The controller program uses all the available sensor data along with the two controllers described above for path planning and navigation. It has three basic modes in which it can operate. It can do transects, it can go upstream and it can navigate back to the home position. The modes can be run individually or they can be combined to create for instance a lawnmower pattern as shown in Fig. 5.

A. Transect mode

The transect mode uses the controller described in Section II-B to make the transects. Instead of manually placing the reference points for the transect this is done autonomously based on sensor data from the ADCP and the LiDAR. The reference for the transect controller is the two end points of the transect line and they are calculated according to:

$$\begin{aligned} x_{[l,r]} &= x_{robot} + (d_{[l,r]} + 10)\cos(\theta_c \pm \pi/2) \\ y_{[l,r]} &= y_{robot} + (d_{[l,r]} + 10)\sin(\theta_c \pm \pi/2) \end{aligned} \tag{6}$$

Once the vehicle has reached the position where the transect should be conducted it will hold position for a specified time. During this time an average current angle is calculated. From this the angles to the right and left reference point is obtained as the perpendicular angles. The distance to the shores at these angles are then obtained from the LiDAR data as described in (5). The points are placed a specified distance, in our tests 10m is used, in land from the shore. This is done to make sure the LiDAR measurements are used to determine when to change direction. The transect direction is then changed when the distance from (5) is shorter than some specified distance.

B. Upstream mode

The upstream mode keeps the boat in the middle of the river or at a constant distance from the shore while going upstream. To do this the LOS controller described in Section II-A is used. The reference for this controller is a reference point.

This point is generated by finding the middle point of the river from the LiDAR measurements perpendicular to the current. The reference point is then placed upstream a specified distance. This point is updated every sample.

C. Return home mode

In the return home state the boat will go downstream towards the home position. It will be kept in the middle of the river until the angle between the heading of the boat and the direction to the home position is smaller than $\pi/3$. The home position is either specified by the user or set as the start position.

V. Results

The results presented are from deployment in Kern River, California the 19th-20th of August 2019. The current velocity during the tests was around 1.4m/s. The data presented comes from two runs with the controller program described in Section IV.

A. Setup

For the tests conducted the following controller gains were used for the LOS controller: $K_t = 500$, $T_{i,t} = 1$, $K_r = 100$ and $T_{i,r} = 10$. The gains for the transect controller were $K_f = 400$ and $K_\delta = 480$. The sample time of the controller was 0.2s and the number of points used to average the LiDAR measurements was 21.

B. Transect

In the first run a lawnmower pattern is created by making three transects and going 15m upstream in between. The minimum distance to the shore is set to 10m and the reference when going upstream is chosen as one fourth of the width of the river from the shore. The data in Fig. 5, 6, 7, 8 and 9 are from this run. Fig. 5 shows the run together with the reference lines on a map over the river and in Fig. 6 the controller error is displayed. From these two figures it is clear that there is a static error in the position. This is expected as the controller has no integral part. At $t = 25s$ in Fig. 6 the transect direction is changed. The sudden change in the error is probably due to the transect line not being perfectly perpendicular to the current. Fig. 7 shows the LiDAR data from this run where it is clear that the vehicle changes direction when it is 10m from the shore. The depth profile shown in Fig. 8 is estimated using a 1D Kalman filter according to the method presented by [6]. The depth close to the shore is around 0.2m and the depth close to the other side of the river is around 1.4m. In Fig. 9 the current measurements from the ADCP is shown. The measurement is noisy, but the mean value of 1.4m/s is fairly close to the measurements of 1.3m/s at the same location made by [6].

During the transects there were very obvious visual oscillations in the boat's position. However these oscillations do not show in the data. Hence they were probably due to faulty GPS readings.

C. Upstream

Fig. 10, 11 and 12 shows shows data from a run where the boat goes upstream in the middle of the river for 300s before returning back to the starting position. Fig. 10 shows the boat's position on a map of the river and 11 shows the LiDAR readings while staying in the middle. Around $t = 240s$ the shoreline on the right hand side changes drastically as can be seen in Fig. 10. There is a small bay, but mainly it changes from having trees hanging out over the river to flat ground. Fig. 12 the depth estimations are shown and these corresponds well with the depth shown in Fig. 8.

VI. Conclusions

As shown above the new controller program does what it is supposed to do and should simplify the use of the vehicle. Still there are things to improve with the design.

At the moment the LiDAR only measures the distances to the shores. Hence there are a lot of data that is not used. To look forward as well would be a good improvement as this could allow some obstacle avoidance.

Another thing that should be improved is probably the position estimation. At the moment only the GPS is used for this. As mentioned in Section V-B oscillations occurred in the transect, probably due to inaccurate GPS readings. Even if the GPS position would not oscillate it still only has an accuracy of 40cm. This might not be accurate enough to measure how the river bed changes over time. By using more sensors to calculate the position the estimate would be more reliable. Sensors used for this could be the ADCP, the IMU and the LiDAR.

Fig. 5. Transect using the program described in Section IV. The red lines are the boat's position and the blue lines are the reference lines for the transects.

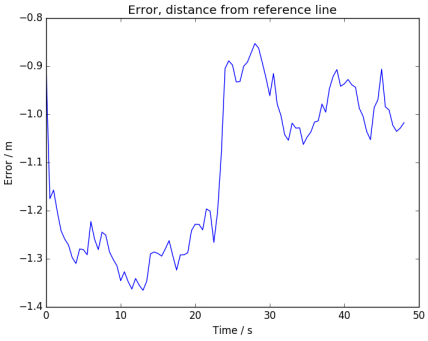

Fig. 6. Distance to the transect line during middle transect in Fig. 5.

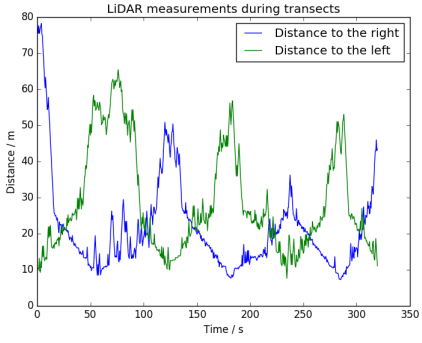

Fig. 7. Distance to the shore measured by the LiDAR during the transects shown in Fig. 5. Distances calculated according to (5 and the transect direction changes when the distance to the shore is shorter than 10m.

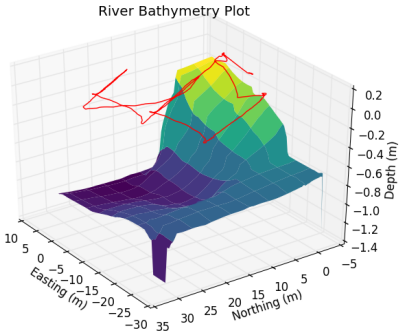

Fig. 8. Transects shown above together with depth estimations from the ADCP.

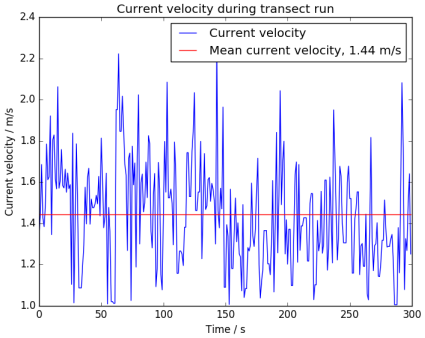

Fig. 9. Current velocity during transect run shown in Fig. 5 with mean current of 1.44 m/s.

Another welcome improvement would be to add integral gain to the transect controller to remove the static error.

ACKNOWLEDGEMENTS

This project was done thanks to funding from the THOR grant, the SURF program and CAST at Caltech. The control system of the boat was developed together with John Lee. James D. Walker helped with packaging of the LiDAR and during testing. Christopher M. Clark and Michael P. Lamb gave valuable inputs and advise.

Fig. 10. Run upstream in the middle of the river, using the program described in Section IV.

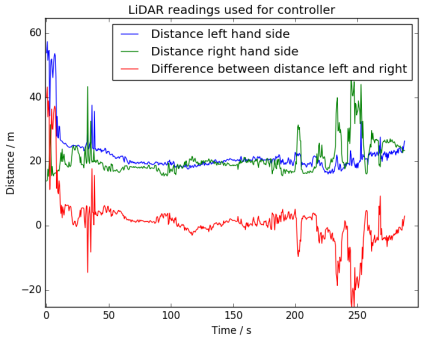

Fig. 11. LiDAR readings while going upstream in run shown in Fig. 10 calculated using (5).

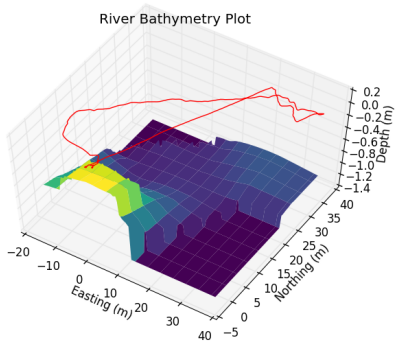

Fig. 12. Run shown above together with the depth estimations from the ADCP.

REFERENCES

[1] A. Benfield, 2016, "2016 Annual Global Climate and Catastrophe Report", 56 pp. [Online]. Available: http://thoughtleadership.aonbenfield.com/Documents/20170117-ab-if-annual-climate-catastrophe-report.pdf

[2] M.P. Lamb, J.A. Nittrouer, D. Mohrig, J. Shaw, 2012, "Backwater and river-plume controls on scour upstream of river mouths: Implications for fluvio-deltaic morphodynamics". Journal of Geophysical Research Earth Surface, v. 117, no. F1. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2011JF002079

[3] R.B. Wynn, V.A. Huvenne, T.P.L. Bas, B.J. Murton, D.P. Connelly, B.J. Bett, H.A. Ruhl, K.J. Morris, J. Peakall, D.R. Parsons, E.J. Sumner, S.E. Darby, R.M. Dorrell, J.E. Hunt, 2014, "Autonomous underwater vehicles (AUVs): Their past, present and future contributions to the advancement of marine geoscience," Marine Geology, vol. 352, pp. 451 – 468, 50th Anniversary Special Issue. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0025322714000747

[4] S. Scherer, J. Rehder, S. Achar, H. Cover, A. Chambers, S. Nuske, S. Singh, 2012, "River mapping from a flying robot: state estimation, river detection, and obstacle mapping," Autonomous Robots, vol. 33, no. 1, pp. 189–214. [Online]. Available: https://doi.org/10.1007/s10514-012-9293-0

[5] A. Singh, M.A. Batalin, V. Chen, M. Stealey, B. Jordan, J.C. Fisher, T.C. Harmon, M.H. Hansen, W.J. Kaiser, 2007, "Autonomous robotic sensing experiments at san joaquin river," in Proceedings 2007 IEEE International Conference on Robotics and Automation, pp. 4987–4993.

[6] C. Lee, J. Wu, R.M. Murray, C.M. Clark, W.W. Fischer, A.T. Hayden, M.P. Lamb, 2018, "River Mapping with Autonomous Surface Vehicle in Turbid Currents". Caltech SURF reports 2018.