Multirobot Tethering for Localization and Control

Baillio, Brad; Vladimerou, Vladimeros

2009

Multirobot Tethering for Localization and Control

Brad Baillio
General Dynamics Robotic Systems
bbaillio@gdrs.com

Vladimeros Vladimerou
Reglerteknik, LTH, Lund University
vladimeros.vladimerou@control.lth.se

***Abstract*—We investigate the use of particle filter (PF) estimation techniques on a hovercraft vehicle in an office environment. Monte Carlo Localization (MCL) with particle filtering is a popular method for localizing robots with laser range finders. In maps featuring long, uniform corridors though, a PF can produce low confidence estimates. When used as feedback to control an unstable vehicle this can prove fatal. This is because, unlike grounded wheeled vehicles, an airborne hovercraft requires accurate localization not only for path planning, but for stabilization as well. We solve the low confidence problem using a secondary networked robot as a mobile map feature.**

***Keywords*-localization; particle filter; hovercraft; networked autonomous vehicles; laser range finder;**

I. Primary Hovercraft Vehicle

Figure 1 shows our HoTDeC hovercraft vehicle (design details in [3]) which has a circular foam-mold body and a slightly inflatable rubber skirt. Inside are embedded processors and amps, batteries and a battery charging system. Four thrusters are positioned as seen in Figure 3. By their arrangement they can provide lateral forces as well as torque. A fifth thruster is used for lift. Using a nonlinear map from force and torque to thruster RPMs we are able to design on a simple nominal viscous friction model shown in Fig. 4. We designed an H_∞ controller to stabilize the hovercraft on a given world coordinate and minimize the effect of input noise. The same controller is extended to traverse predefined trajectories. It is not possible to use the output of a controller to estimate the state because of gravity forces originating from unknown floor incline. 6-DOF models using inertial sensors did not prove effective. Feedback is therefore received from a PF which uses a top-mounted laser scanner and a predefined map, so the nominal model is augmented to account for PF processing delays. Without accurate location and orientation estimates the vehicle can become unstable due to the nonlinear nature of the controller.

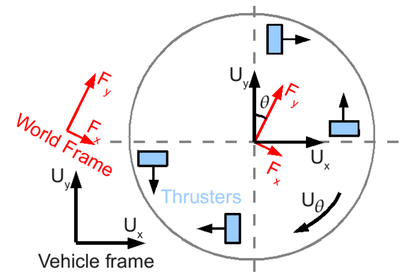

Fig. 3. Thrusters and torque/forces w.r.t vehicle frame (U_x, U_y, U_θ) and world frame (F_x, F_y, U_θ)

II. Particle filtering for Localization

The core of the PF technique is to approximate a target distribution by a finite number of parameters by choosing a set of random state samples (particles) drawn from the target distribution [4]. When used in the localization problem, PFs fit into an algorithm called Monte Carlo localization (MCL). Each particle is sent through the state equations to generate a new predicted pose sample. This predicted sample is then sent through an observation equation (based on the sensors and map) to generate a likelihood, or weight, for the predicted pose. Our map is generated a priori and stored in the vehicle memory. When we have our target distribution, to complete the estimation, the correction is incorporated by resampling the particle set proportional to the weight of the particles. A

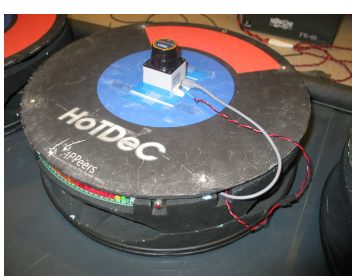

Fig. 1. Hovercraft with laser scanner on top.

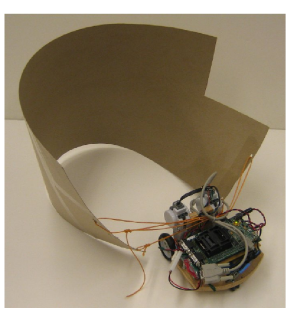

Fig. 2. Herdbot with parachute.

$$\dot{Y} = AY + Bu + Gw \qquad z = CY + Hv$$

$$A = diag(\begin{bmatrix} 0 & 1 \\ 0 & -\frac{\beta_x}{m} \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & -\frac{\beta_y}{m} \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & -\frac{\beta_\theta}{J} \end{bmatrix})$$

$$Y = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ \theta \\ \dot{\theta} \end{bmatrix} \qquad B = G = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{J} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Fig. 4. Hovercraft Dynamics

likelihood-weighted particle average is used for feedback to the controller. An overview of the algorithm is shown in Fig 5.

Note that in predicting the pose of the hovercraft for resampling, we cannot rely heavily on the known thruster commands, as some of the output is used to compensate for relatively large ground incline disturbances. We force the hovercraft to move slowly and use random perturbations around the current position instead.

MCL_Algorithm($\chi_{t-1}, u_t, z_t, map$):
$\overline{\chi}_t = \chi_t = 0$
for $m = 1$ to M do
  sample $x_t^{[m]} \sim p(x_t | u_t, x_{t-1}^{[m]})$
  $w_t^{[m]} = p(z_t | x_t^{[m]}, map)$
  $\overline{\chi}_t = \overline{\chi}_t + < x_t^{[m]}, w_t^{[m]} >$
endfor
for $m = 1$ to M do
  draw i with probability $\propto w_t^{[i]}$
  add $x_t^{[i]}$ to χ_t
end for
return χ_t

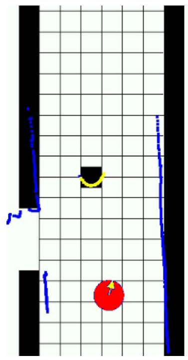

Fig. 5. The MCL Algorithm, hovercraft, herdbot and laser scan.

III. Tethering

Due to the static nonlinearity in controller implementation the confidence of the PF measurement on orientation needs to be quite high. The same is necessary for velocity and position estimation along our trajectory. Very good tracking has been obtained in maps with distinguishable features like in the beginning of the long hallway in Fig. 5.

As already stated, in certain uniform environments (such the upper part of the long hallway in Fig 5) the particle filter will generate multiple pose hypotheses that are equally likely. When this happens, the system loses velocity control, oscillates and cannot recover. If the hovercraft vehicle could recognize a distinct feature in these otherwise "homogeneous" environments, the state estimation could produce a single hypothesis. We provide a mobile, controllable feature: the robot referred to as the *herdbot*. The herdbot is shown in Figure 2. It uses two wheel motors for differential drive and a ball-bearing caster wheel. In order to better sense the small robot using the laser scanner on the hovercraft, we attach a cardboard parachute that the herdbot drags behind. Although easily controllable, a mobile feature conflicts with the static environment assumed by the particle filter. To overcome this problem, the hovercraft and herdbot leapfrog their movements. This leapfrog action is the crux of the multirobot tethering system. Coordination is achieved through a Bluetooth wireless channel.

IV. Multirobot Tethering Algorithm Overview

The hovercraft operates in two modes: Overseer and Worker, depicted in Figure 6.

In the Overseer mode, the hovercraft is at rest at a known location. From this stable state, it can track the changes in the environment, as well as issue commands to the herdbot. It thus commands the herdbot forward a specified distance. We use the Split-and-Merge line detection algorithm from [2] to detect the herdbot. Once the it reaches its destination, the hovercraft commands it to stop moving, records the final destination location, and modifies the map to reflect the change. In this way, the localization assumption of static maps is not violated because we change the map when localization does not occur. At this point, the hovercraft is ready to switch into Worker mode.

In Worker mode, the hovercraft performs all the calculations to localize and move. This is the mobile autonomous phase for the hovercraft. It lifts off, moves toward the herdbot, settles into a stable location, and lands. It is during this mode that the localization algorithms are exercised. In our MCL algorithm, we bias our sensor model around the herdbot. If we bias our sensor model around it, we are more likely to get higher weights for the particles closest to the true pose. We are doing a mixture of scan-based localization and feature-based localization. Our leapfrog approach encourages this hybridization. The resultant pose is fed into the H-infinity controller, which does its job to keep the craft stable during movement. Once the hovercraft is at rest at a known location, it switches back to the Overseer mode, and the whole process starts again. Details in[1].

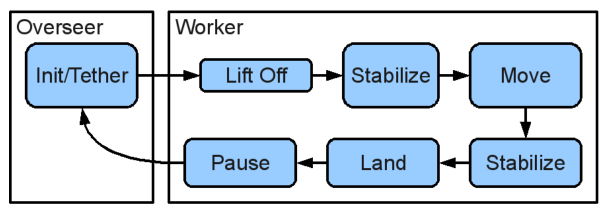

Fig. 6. The hovercraft modes of operation

V. Acknowledgements

The authors would like to thank Daniel McKenna for his work on the herdbot. Part of the hovercraft development and related research was funded by NSF under grant 0729500. Vladimeros Vladimerou is partially sponsored by the EU/IST project CHAT.

References

[1] B. Baillio. Multirobot tethering for localization and control. Master's thesis, University of Illinois at Urbana-Champaign, 2009.
[2] V. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart. A comparison of line extraction algorithms using 2d laser rangefinder for indoor mobile robotics. *Autonomous Robots*, 23:97–111, August 2007.
[3] A. Stubbs, V. Vladimerou, A. Fulford, D. King, J. Strick, and G. Dullerud. Multivehicle systems control over networks. *IEEE Control Systems Magazine*, 8(3):56 – 69, June 2006.
[4] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, Cambridge, MA, 2005.