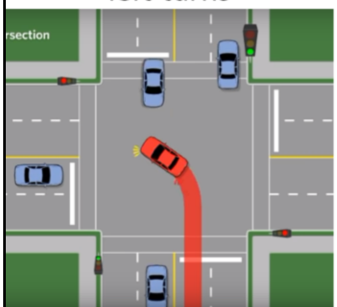
Range-based Cooperative Localization with Nonlinear Observability Analysis

Brandon Araki, Igor Gilitschenski, Tatum Ogata, **Alex Wallar**, Wilko Schwarting, Zareen Choudhury, Sertac Karaman, Daniela Rus

Intersection Localization

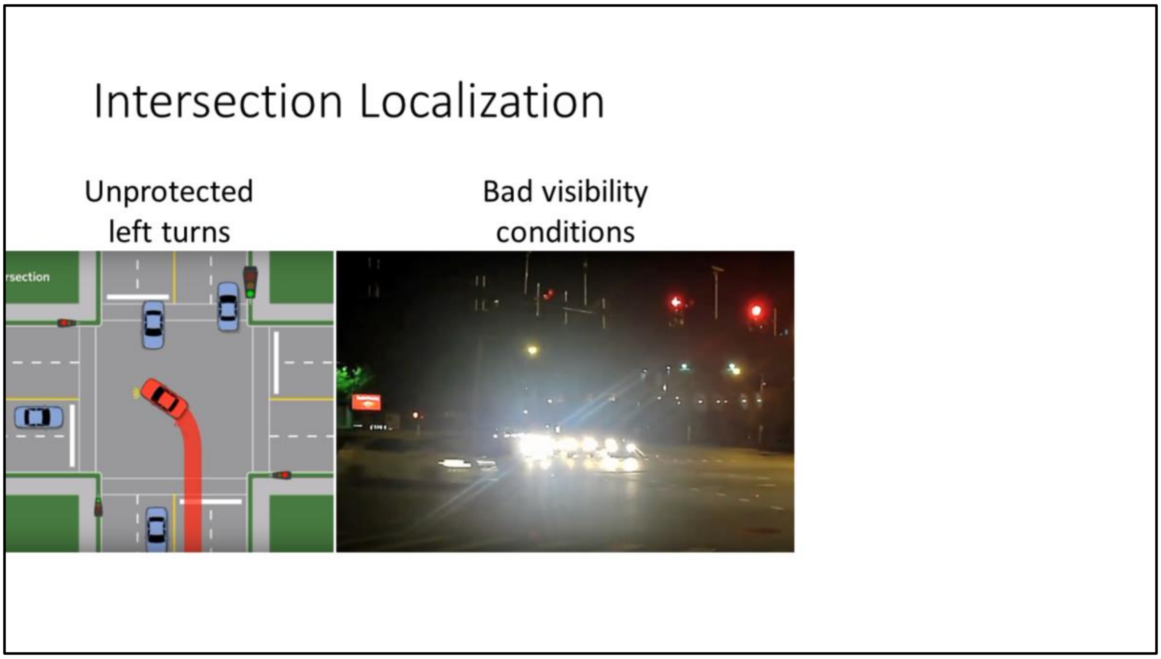
Intersection Localization

Unprotected left turns



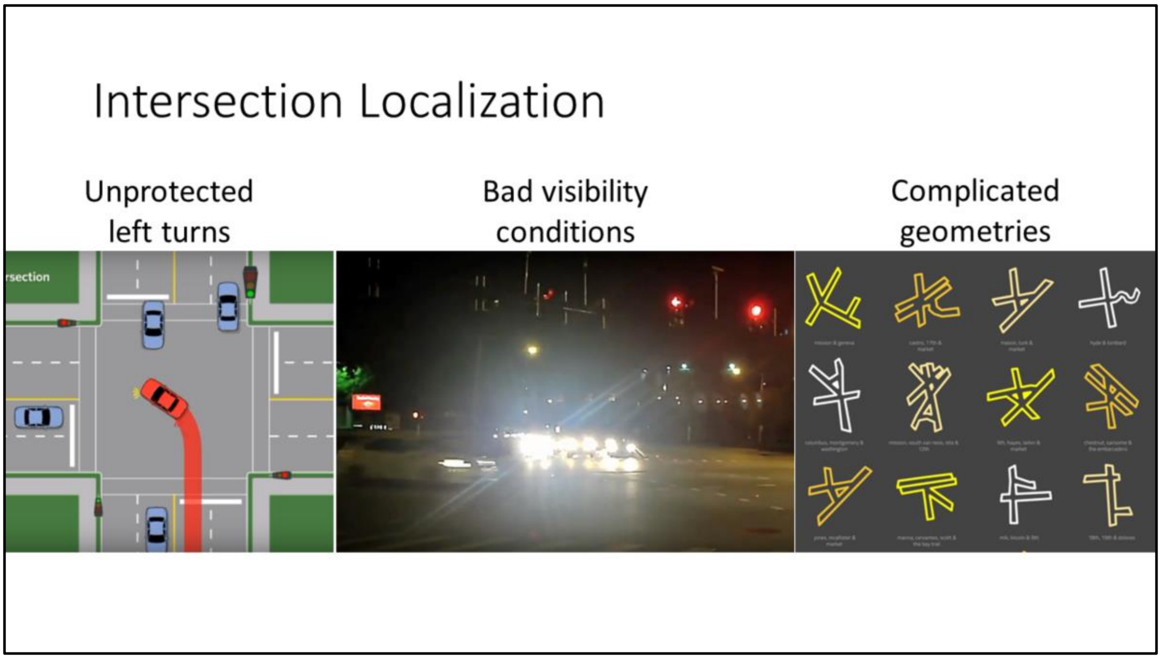
1. Intersections are one of the most challenging environments for doing localization. In intersections you have cars moving in opposite directions very close to each other, and in the case of unprotected turns, small cues such as a nudge or a tap on the brakes can carry a lot of meaning. In order to navigate intersections safely and intelligently, cars need to have fast and accurate localization.

Intersection at night: https://www.youtube.com/watch?v=67irRh4tPnE
SF intersections: https://sf.curbed.com/2018/2/13/17008224/intersection-art-artist-graphic-print-san-francisco-streets
Left-hand turn: https://www.youtube.com/watch?v=ZaX9Q6nvUK8



2. One challenge for accurate localization are bad visibility conditions, such as when it’s dark or there is rain or fog.

Intersection at night: https://www.youtube.com/watch?v=67irRh4tPnE
SF intersections: https://sf.curbed.com/2018/2/13/17008224/intersection-art-artist-graphic-print-san-francisco-streets
Left-hand turn: https://www.youtube.com/watch?v=ZaX9Q6nvUK8



3. Another challenge is that intersections often have complicated geometries that make non-line-of-sight localization difficult.

Intersection at night: https://www.youtube.com/watch?v=67irRh4tPnE
SF intersections: https://sf.curbed.com/2018/2/13/17008224/intersection-art-artist-graphic-print-san-francisco-streets
Left-hand turn: https://www.youtube.com/watch?v=ZaX9Q6nvUK8

Sensor Challenges

All of the sensors currently used for localization have limitations that make intersection localization difficult.

Sensor Challenges

Lidar

- Relies on line-of-sight
- Slow

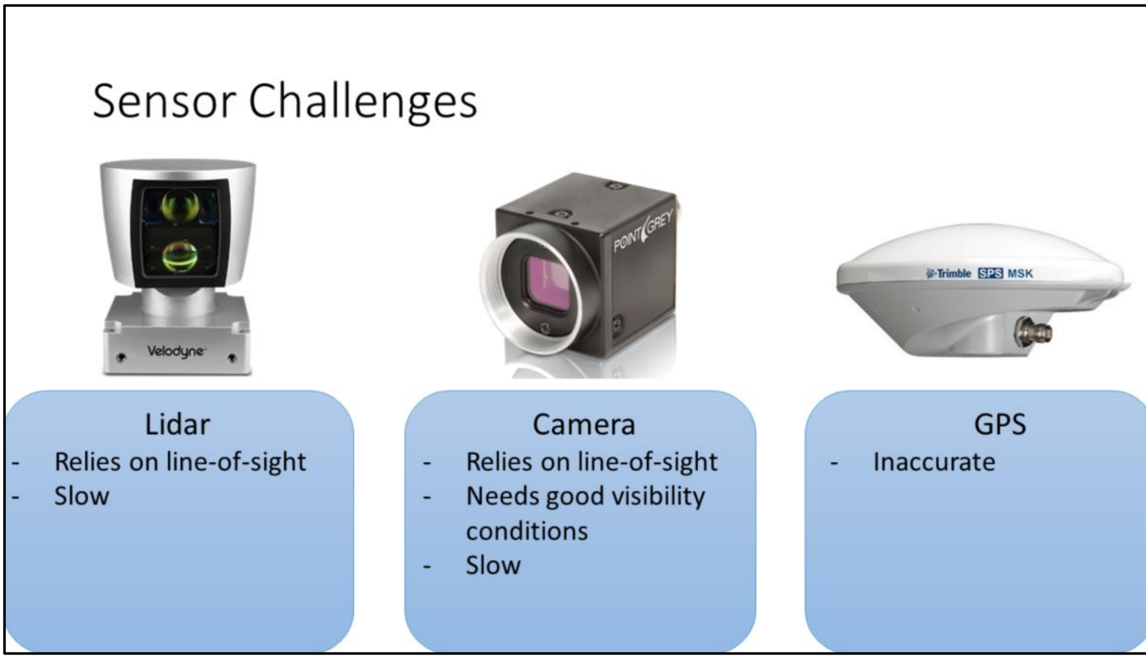
1. Lidar relies on line-of-sight, and object segmentation is relatively slow.

Sensor Challenges

Lidar
- Relies on line-of-sight
- Slow

Camera
- Relies on line-of-sight
- Needs good visibility conditions
- Slow

2. Cameras also rely on line-of-sight as well as good visibility conditions, and object segmentation is once again relatively slow.



3. Lastly, GPS has poor accuracy.

Ultra-wideband radio

- Does not require line-of-sight
- Robust to poor visibility and bad weather
- Fast

In this work we used ultra-wideband radios, or UWBs, for localization. UWBs transmit electromagnetic waves over a wide range of frequencies in order to make time-of-flight range measurements. The wide range of frequencies used helps make them robust to non-line-of-sight measurements, and it also means that they are robust to poor visibility and bad weather. Moreover, since range measurements are obtained directly from the UWB, no post-processing of the data is necessary before integrating it into localization.

Ultra-wideband radio

- Does not require line-of-sight
- Robust to poor visibility and bad weather
- Fast

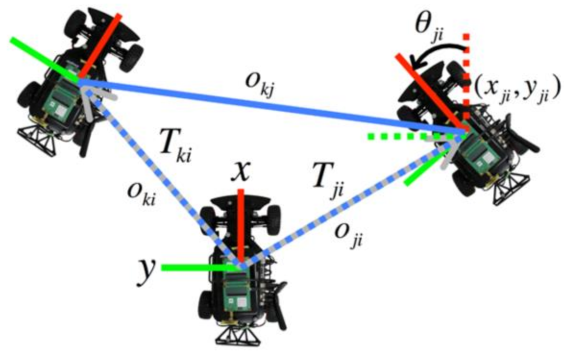
What's the catch?

- UWBs provide only a range measurement – no angle

1. There is one catch – UWBs provide only a range measurement and no angle. Therefore the relative angles between cars must be inferred over a series of measurements.

Problem Outline

- Given a set of vehicles and initial guesses of their relative transforms, we receive **odometry** and **relative range** measurements in a **central server**.
- How to estimate the relative transforms between the vehicles?



So our problem is that we want to estimate the relative transforms between vehicles. We assume that each vehicle's odometry and relative range measurements are sent to a central server, which then uses the data to estimate the relative transforms.

Model

State and action space

Relative transforms: $T_{ji} = [x_{ji}, y_{ji}, \theta_{ji}]$

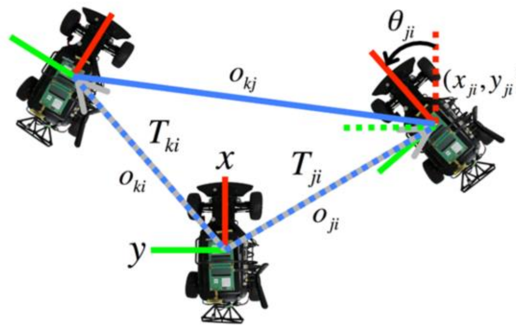
Inputs: $u_i = \{v_i, \omega_i\}$

Motion model

$$\frac{dT_{ji}}{dt} = \begin{bmatrix} v_j \cos\theta_{ji} \\ v_j \sin\theta_{ji} \\ \omega_j \end{bmatrix} + \begin{bmatrix} y_{ji}\omega_i - v_i \\ -x_{ji}\omega_i \\ -\omega_i \end{bmatrix}$$

Observation model

Range measurements: $o_{ji} = \sqrt{x_{ji}^2 + y_{ji}^2}$



1. The state space of our model is the relative transforms between each pair of cars, where x, y, and theta represent the relative Cartesian and angular distances between the cars.
2. The inputs of the model are linear and angular velocity for each vehicle.
3. This is the motion model, which captures the relative dynamics between each pair of vehicles.
4. The observation model is the relative distance between each pair of vehicles.

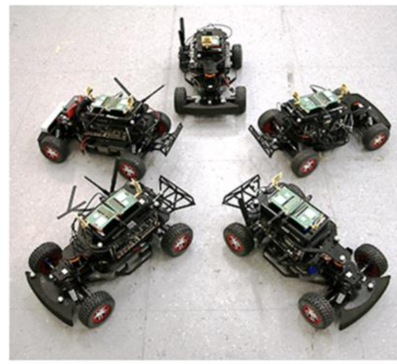
Note that in our model, odometry measurements are interpreted as inputs to the model, and the UWB measurements are treated as observations.

Nonlinear Observability Analysis

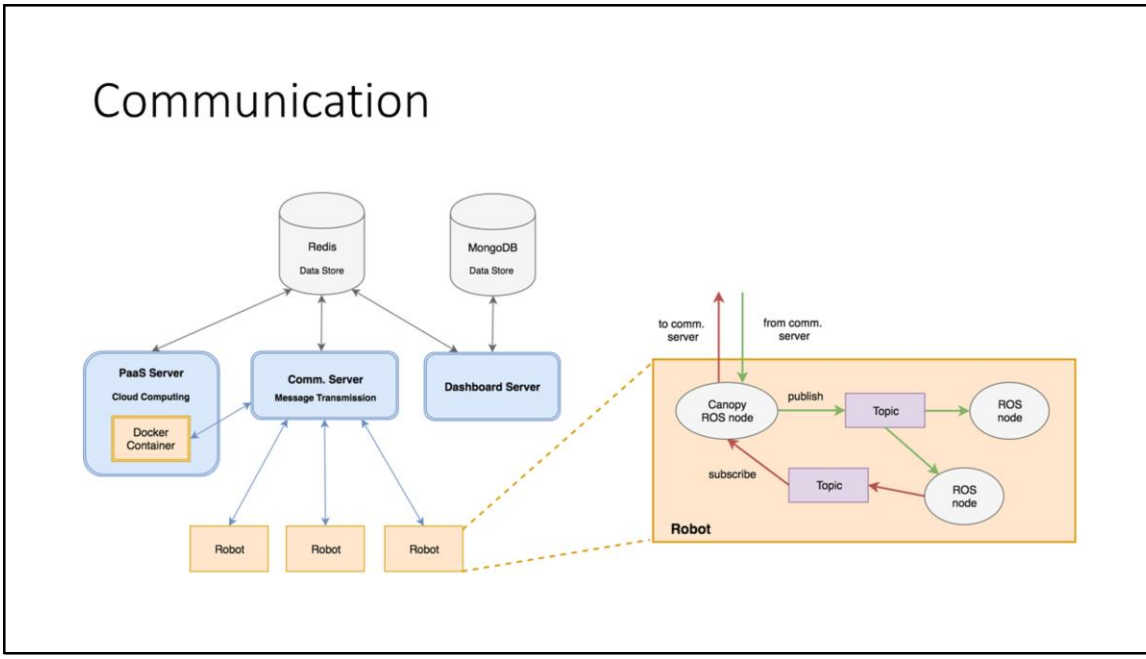
- First of all, is this even possible?
- We show in the paper in the 2-car case, as long as both cars have non-zero velocity, the transform between them is observable.

Before I proceed, we might ask the question, is this estimation problem even possible? Given two cars, the only information we receive is odometry information from both cars and a single range measurement. In our paper we do a nonlinear observability analysis to check if it possible to use this information to estimate the relative transform between the two cars – the technical details are in the paper, but we show that in the 2-car case, as long as both cars have non-zero velocity, the transform between them is observable.

Experiments

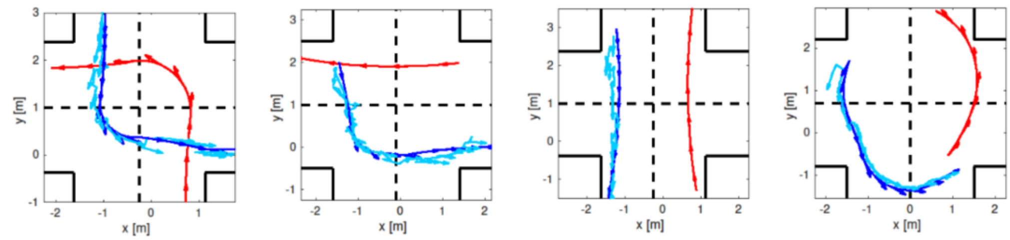


We implemented an EKF, a UKF, and a particle filter based on the model specified earlier. We tested the localization algorithms on a real platform with up to 5 miniature racecars. These racecars are based on the open-source MIT Racecar platform. They have a Traxxas chassis and an Nvidia Jetson TX2 on-board computer. Odometry measurements come from a Vedder ESC and a steering servo. Each car also has two Decawave TREK1000 UWB modules which make range measurements.



The cars communicated with a central server. We used an open-source ROS multi-master system called Canopy. The multi-master system allows selective message passing between ROS masters, allows vehicles to join and leave the server at will, and allows communication through multiple networks.

Experiments



RMSE	Filter	Left Turns	Left Turn/ Straight	Straight	Semicircle
Dist [m]	PF	0.26	**0.41**	0.32	0.33
	UKF	0.26	0.80	**0.29**	**0.29**
	EKF	**0.25**	0.85	0.33	0.33
Theta [rad]	PF	0.11	**0.17**	0.08	0.15
	UKF	0.10	0.27	0.03	**0.03**
	EKF	**0.09**	0.28	**0.02**	0.08

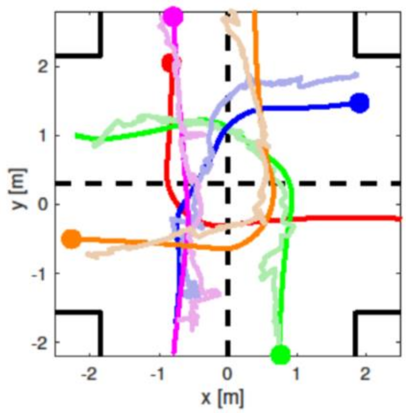
We ran four different two-car experiments.
In the graphs, the red and dark blue arrows represent ground truth, while the light blue arrows represent the red car's estimate of the blue car. We are showing the results using the particle filter.

In the first experiment, both cars turn left. In the second, the red car goes straight while the blue car turns left. In the third experiment, both cars drive in a straight line. In the last experiment, both cars drive a semicircle to the left.

As can be seen in the experiments, localization is effective even in the case where we receive just odometry information and a single range measurement.

All of the filters had about the same performance, with localization error being around 30 cm for all of the experiments except for the 2nd experiment, where it was about 80 cm for the UKF and EKF and 41 cm for the particle filter.

Experiments



RMSE	Filter	T_{21}	T_{31}	T_{41}	T_{51}
Dist [m]	PF	0.31	0.43	0.26	0.44
	UKF	**0.26**	**0.20**	**0.29**	**0.32**
	EKF	0.36	0.80	0.64	0.59
Theta [rad]	PF	0.22	0.20	0.24	0.38
	UKF	0.20	**0.18**	**0.17**	**0.16**
	EKF	**0.14**	**0.18**	0.29	0.34

We also ran a 5-car experiment, where four cars make left turns and one car drives straight through the intersection. In this case the errors for the transforms were between 14 and 32 cm. The main challenge with this experiment was to enable the server to receive data from all of the cars with the proper time stamps and with low-latency, which we were able to achieve with our multi-master system.

Thank you!