CHALMERS
UNIVERSITY OF TECHNOLOGY

Ego-Motion Estimation Using Radars

Master's thesis in Systems, Control and Mechatronics

LUCAS HAGLUND
JOHAN RUNEBY

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's Thesis in Systems, Control and Mechatronics

Ego-Motion Estimation Using Radars

LUCAS HAGLUND
JOHAN RUNEBY

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mechanics and Maritime Sciences
Vehicle Engineering and Autonomous Systems
Vehicle Dynamics
Chalmers University of Technology
Gothenburg, Sweden 2024

Ego-Motion Estimation Using Radars
LUCAS HAGLUND
JOHAN RUNEBY

Supervisor: Murat Kumru, Volvo GTT
Examiner: Mats Jonasson, Department of Mechanics and Maritime Sciences

Master's Thesis 2024
Department of Mechanics and Maritime Sciences
Vehicle Dynamics
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone +46 31 772 1000

Cover: Truck driving during an IPG TruckMaker simulation.

Typeset in LaTeX
Gothenburg, Sweden 2024

Ego-Motion Estimation Using Radars
LUCAS HAGLUND
JOHAN RUNEBY
Department of Mechanics and Maritime Sciences
Vehicle Engineering and Autonomous Systems
Vehicle Dynamics
Chalmers University of Technology

Abstract

Vehicle ego-motion can be described by three ego-pose states, three translational ego-velocities and three angular ego-velocities. All nine states can be estimated using range and Doppler velocity measurements from six radars mounted on the ego-vehicle. This thesis proposes a Kalman Filter with a linear measurement model for estimating ego-velocity from Doppler measurements and an Extended Kalman Filter with a nonlinear measurement model for estimating ego-pose from range measurements. Ego-motion groundtruth and radar measurements were simulated using IPG TruckMaker for a set of simple road variations and driving maneuvers. The ego-velocity estimate error is sensitive to how the radars are mounted, for which a design procedure is proposed.

Keywords: Automotive radar, Ego-motion, Ego-pose, Ego-velocity, Measurement model, State estimation, IPG TruckMaker.

Acknowledgements

We would like to express our gratitude to our supervisor, Murat Kumru, for all the support, advice, and encouragement throughout our research and while writing this thesis. Your knowledge and guidance have been very helpful.

We'd also like to thank our examiner from Chalmers University of Technology, Mats Jonasson, for the valuable feedback and suggestions throughout this spring. Thanks to our MPSYS classmates Guna Sekar Karri and Akshay S Aravind from the IMU project for sharing this master's thesis journey.

Also, thanks to David Andersson and Alexander Hägglund at IPG for being very helpful and engaged whenever we had questions regarding TruckMaker.

Lastly but not least, we would like to acknowledge and thank the whole VME team at Volvo GTT for a great welcoming!

Lucas Haglund, Johan Runeby, Gothenburg, June 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

ADAS	Advanced Driver Assistance Systems
CoR	Center of Rotation
CV	Constant Velocity
DFT	Discrete Fourier Transform
(E)KF	(Extended) Kalman Filter
EM	Electro-Magnetic
FFT	Fast Fourier Transform
FMCW	Frequency-Modulated Continuous Wave
FoV	Field of View
HIL	Hardware in the Loop
MIMO	Multiple Input Multiple Output
RDM	Range-Doppler Map
RMSE	Root Mean Square Error
RSI	Raw Signal Interface
RSC	Road Surface Conditions
Rx	Receiver
SIL	Software in the Loop
SNR	Signal-to-Noise Ratio
SVD	Singular Value Decomposition
TM	TruckMaker
Tx	Transmitter
UAQ	User Accessible Quantities

Contents

List of Figures

List of Tables

1
Introduction

1.1 Background

Precise knowledge of the ego-vehicle motion plays a key role in facilitating effective vehicle motion control. Current solutions, which mainly make use of wheel speed sensors and inertial measurements, are known to be limited due to unknown vehicle parameters and operation modes. Ground speed estimations using wheel speed sensors are subject to errors due to slip and deviation of wheel radius. Inertial measurement units are subject to drift. Accordingly, there is a growing interest in developing new technologies to alleviate existing shortcomings. Vehicle radars have been used since the 70s [1], but for collision avoidance applications. The first vehicle speedometer using radar was implemented in 1973 [2], but has not been in common use since.

Radars transmit electromagnetic signals and observe the reflections from the environment, to measure the radial distance between the sensor and the reflection points. The radial distances coupled with their respective observation angles yield relative positions. Radars are also able to measure radial velocity by utilizing the Doppler effect. If the reflection point has a velocity relative to the sensor, the received radar signal wavelength will differ from the transmitted. Vehicles drive in varying environments and radar sensors can function under a wide range of conditions. Whether it be low light, rain or snow, a radar is reliable. Hence, the use of radar for velocity estimation as a complement to already existing vehicle odometry is of interest.

The development of advanced driver assistance systems (ADAS) is important for the safety of the driver as well as the vehicle's surroundings. Changes in the environment must be accounted for and an accurate state of the ego-vehicle must be known. Therefore, radar sensors can be used for state estimation when other sensors might fail.

1.2 Related work

Ego-velocity estimation using Doppler radar measurements has been studied in several articles. In 1993 Kleinhempel implemented a Doppler speedometer. The two sensors were mounted underneath of the vehicle in a Janus configuration, i.e., the sensors look in opposite direction of each other. The measurements from the Doppler

speedometer had approximately 1 % difference to the reference speed. A Kalman filter was applied to further improve measurement accuracy [3].

An attempt at estimating the ground speed by mounting a Doppler radar underneath the car was made by Hantsch et al. They were able to measure ground speed and slip angle [4].

Kellner et al. presents how multiple measurements from a single $\pm 65°$ field of view Doppler radar, can determine the ego-velocity and ego-yaw rate. The algorithm uses RANSAC to eliminate Doppler outliers. Further, it applies the single-track model along with the Ackermann condition. This condition yields that only the ego-vehicles longitudinal velocity and yaw-rate can be estimated. Hence it is not the full 2D velocity with 3 degrees of freedom (i.e., longitudinal and lateral velocity and yaw rate) of the vehicle that is estimated [5].

Kellner et al. solve this problem in a later publication. The proposed full 2D ego-velocity algorithm uses at least two Doppler radars. However, it can be generalized to as many radars as deemed necessary. The proposed method is not dependent on the simplification with the single-track model [6].

Kollberg investigates fusing Doppler radar measurements with measurements from wheel speed encoders and inertial measurement units, to estimate longitudinal velocity and yaw rate. A second approach is proposed, where the Doppler measurements are used to calibrate the other sensors, which in turn are used to estimate the states. In presence of moving radar measurement obstacles, the calibration approach worked better than the fusion approach. Wheel slip is however not investigated. [7].

Sigonius used an Unscented Kalman Filter (UKF) to estimate lateral velocity, lateral acceleration, yaw rate, yaw acceleration and xy center of rotation [8].

Cen et al. present an indirect method based on keypoint extraction and data association. The keypoint extraction's task is to process the raw radar data and determine valuable keypoints such as buildings, cars and humans. In the next step, data association, Cen et al. make use of graph matching and does not rely on any tunable parameters and where outliers are removed automatically. Another advantage of the algorithm is that no a priori knowledge about the ego-motion is needed [9].

1.3 Purpose

The purpose of this master's thesis is to develop and evaluate a method of estimating vehicle ego-motion using radar measurements exclusively.

1.4 Research questions

1. Is it possible to estimate the three ego-pose states from radar measurements?

2. Is it possible to estimate the six ego-velocity states from radar measurements?
3. To estimate the full nine states of ego-motion, how many radars are needed and what are the requirements on how should they be mounted?
4. How accurately can the ego-motion be estimated?
5. What sources of estimate error are there and how sensitive is the estimate to them?

1.5 Objectives

The objectives of this master's thesis are:

1. To construct a measurement model of the ground speed radar to simulate range and radial velocity measurements.
2. To develop an algorithm that processes radial velocity measurements to estimate the ego-velocity.
3. To develop algorithms that process range measurements to estimate the ego-pose.
4. To evaluate our state estimator in simulation.
5. To investigate the sensitivity of the estimates with regards to sensor noise and radar configuration (number of sensors, positions and orientations).
6. To assess the performance of the developed methods using real sensor measurements.

1.6 Scope

Longitudinal position x, lateral position y and yaw angle φ_z will not be estimated, as it would require an absolute/stationary reference from e.g. landmark detection or a GNSS source.

Radar sensor input is assumed to be already pre-processed to be in the form of range and Doppler velocity measurements, from several short-range, Frequency-Modulated Continuous Wave (FMCW) ground speed radars, coupled with elevation and azimuth angles from the radar to the measured point.

Real radar measurements and estimates based on them will not be included in this thesis due to company restrictions.

2
Theory

This chapter introduces the theory necessary to understand the problem definition and methods used later in the thesis. The theory consists of coordinate frames, state notation, estimation theory and radar theory.

2.1 Coordinate frames

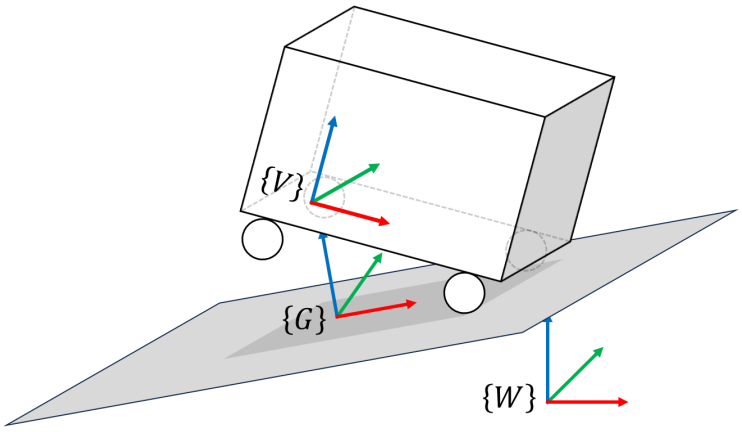

Figure 2.1: World $\{W\}$, Vehicle $\{V\}$ and Ground $\{G\}$ frames.

Nominally all wheels of the vehicle remain in contact with the road and the ego-pitch angle is instead due to difference in the suspensions. Figure 2.1 is exaggerated to show the coordinate frames.

A coordinate frame consists of an origin and an orientation, and is generally fixed to a body. Coordinate frames are used to define the relationships of bodies and quantities. Three coordinate frames are required to describe the states in this thesis:

- **World frame**:
  The world coordinate frame $\{W\}$ is the inertial coordinate frame. It is fixed to the earth and its z axis is parallel to the gravity vector. Its remaining degrees of freedom are arbitrary and inconsequential to the subject of this thesis.
- **Vehicle frame**:
  The vehicle coordinate frame $\{V\}$ is fixed to the ego-vehicle body, oriented so that its x axis, the longitudinal direction, points toward the front of the

vehicle, its y axis, the lateral direction, toward the left side of the vehicle and its z axis, the heave direction, toward the top of the vehicle. If the vehicle body is rigid, the origin of the vehicle frame can be anywhere on body. Otherwise, the vehicle body that the origin is fixed to is considered the ego-vehicle.

- **Ground frame**:
  The ground coordinate frame $\{G\}$ is an intermediate frame between the road and the vehicle. Its z axis is normal to the road plane and its x and y axes are parallel to the projections of the x and y axes of the vehicle frame onto the road plane.

The pose of the ground frame relative to the world frame is divided into elevation for z translation, bank for angle around x, and slope for angle around y.

2.2 Ego-motion states

The states are latent variables, i.e. unknown quantities that cannot be directly observed but need to be estimated. The motion state of the ego-vehicle consists of pose and velocity.

2.2.1 Ego-pose

The ego-pose is defined as the vehicle frame relative to the ground frame, and consists of position and orientation. The state notation and description is displayed in Table 2.1. Though the pose state has six degrees of freedom, due to the definition of the coordinate frames the x and y positions and yaw angle φ_z are always zero.

Name	State	Unit	Frame	Description
Position	z	m	$\{G\}$	Heave
Orientation	$\boldsymbol{\varphi} = \begin{bmatrix} \varphi_x \\ \varphi_y \end{bmatrix}$	deg	$\{G\}$	Roll angle Pitch angle
Pose	$\mathbf{x}_{\text{pose}} = \begin{bmatrix} z \\ \boldsymbol{\varphi} \end{bmatrix}$		$\{G\}$	

Table 2.1: State notation for ego-pose.

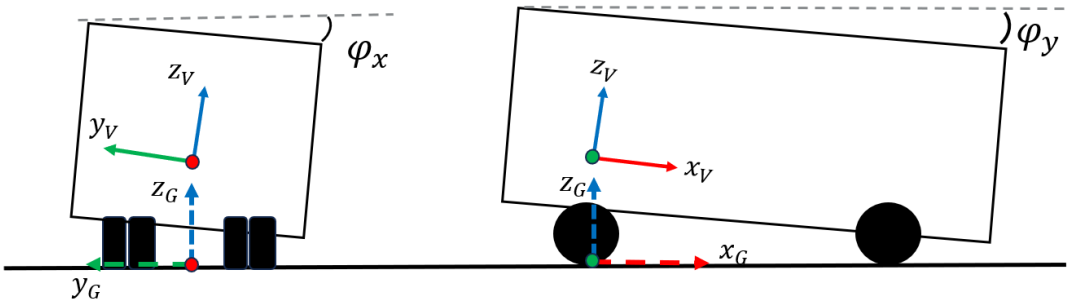

Figure 2.2: Roll angle φ_x (left) and Pitch angle φ_y (right) illustration on a vehicle.

The ego-orientation can be described by the ZYX-order rotation matrix $\mathbf{R}$ from $\{V\}$ to $\{G\}$:

$$\begin{aligned}
\mathbf{R} &= \mathbf{R}_{ZYX}(\boldsymbol{\varphi}) && (2.1)\\
&= \mathbf{R}_Z(0)\mathbf{R}_Y(\varphi_y)\mathbf{R}_X(\varphi_x) && (2.2)\\
&= I \begin{bmatrix} \cos(\varphi_y) & 0 & \sin(\varphi_y) \\ 0 & 1 & 0 \\ -\sin(\varphi_y) & 0 & \cos(\varphi_y) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\varphi_x) & -\sin(\varphi_x) \\ 0 & \sin(\varphi_x) & \cos(\varphi_x) \end{bmatrix} && (2.3)\\
&= \begin{bmatrix} \cos(\varphi_y) & \sin(\varphi_x)\sin(\varphi_y) & \cos(\varphi_x)\sin(\varphi_y) \\ 0 & \cos(\varphi_x) & -\sin(\varphi_x) \\ -\sin(\varphi_y) & \sin(\varphi_x)\cos(\varphi_y) & \cos(\varphi_x)\cos(\varphi_y) \end{bmatrix} && (2.4)
\end{aligned}$$

2.2.2 Ego-velocity

The ego-velocity is defined as the velocity of vehicle frame relative to the world frame but in the vehicle frame coordinates. It consists of the six translational and angular velocities. The state notation and description is seen in Table 2.2 and Figure 2.3.

Name	State	Unit	Frame	Description
Translational velocity	$\mathbf{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$	m/s	$\{V\}$	longitudinal lateral heave
Angular velocity	$\boldsymbol{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$	deg/s	$\{V\}$	roll rate pitch rate yaw rate
Velocity	$\mathbf{x}_{\text{velocity}} = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix}$		$\{V\}$	

Table 2.2: State notation for ego-velocity.

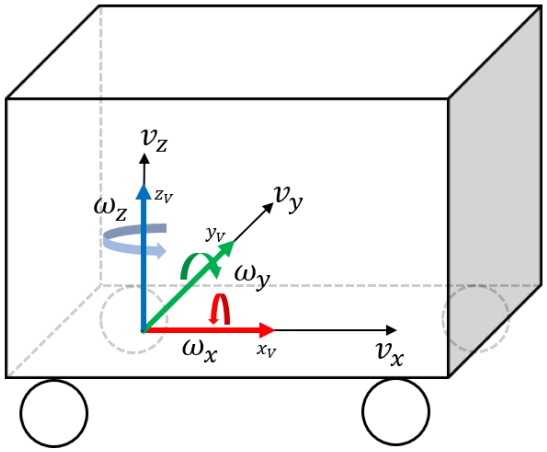

Figure 2.3: Velocity states in vehicle coordinate frame.

The subset of ego-velocity states that lie in the road plane, $\mathbf{x}_{2D}$, is of specific interest. It consists of:

$$\mathbf{x}_{2D} = \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} \tag{2.5}$$

2.3 Bayesian inference

Bayesian inference is an approach where Bayesian statistics are used to model unknown parameters as random variables. If an unknown parameter θ is related to a measured quantity y, Bayesian statistics can be used to update the beliefs about θ. This is called the posterior density, i.e. the probability of θ after measuring y, and can be computed using Bayes rule,

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \tag{2.6}$$

where $p(y|\theta)$ is the likelihood function or measurement model, $p(\theta)$ is the prior distribution and $p(y)$ is a normalization constant factor [10]

2.4 State estimation

State estimation is a method used to determine the state of a dynamical system based on available measurements. Since it is often impossible to measure all unknown states of a system, one can derive models regarding how the unknown quantities are related and evolve over time.

2.4.1 State space model

A discrete time state space model can be expressed in the general form

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{q}_{k-1}) \tag{2.7}$$

$$\mathbf{y}_k = h_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{r}_k) \tag{2.8}$$

where $\mathbf{x}_k$ is the state vector of unknown parameters, $\mathbf{u}_k$ is the system input and $\mathbf{q}_{k-1}$ is process noise, $\mathbf{r}_k$ measurement noise and initial prior distribution of the state at time zero,

$$\mathbf{x}_0 \sim p(\mathbf{x}_0) \tag{2.9}$$

Equation 2.7 is referred to as the process or motion model which describes the evolution of the state vector $\mathbf{x}_k$ over time. It is equivalent to describe the process model as a distribution $p(\mathbf{x}_k|\mathbf{x}_{k-1})$. Equation 2.8 is referred to as measurement model, and it describes how to relate the current state vector $\mathbf{x}_k$ to the measurements $\mathbf{y}_k$. It is equivalent to describe the measurement model as a distribution $p(\mathbf{y}_k|\mathbf{x}_k)$, also called the likelihood function describing how likely certain values of θ are given the new information in measurement y. The models, $f(\cdot)$ and $h(\cdot)$, can be either linear or non-linear, which in turn will decide what filter needs to be applied [10].

2.4.2 Kalman filter

Consider that Equation 2.7, 2.8 are linear and Gaussian. The state space model is on the general form

$$\mathbf{x}_k = \mathbf{A}_k\mathbf{x}_{k-1} + \mathbf{B}_k\mathbf{u}_{k-1} + \mathbf{q}_{k-1} \tag{2.10}$$

$$\mathbf{y}_k = \mathbf{C}_k\mathbf{x}_k + \mathbf{D}_k\mathbf{u}_k + \mathbf{r}_k \tag{2.11}$$

$$\mathbf{q}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k) \tag{2.12}$$

$$\mathbf{r}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \tag{2.13}$$

where $\mathbf{Q}_k$ and $\mathbf{R}_k$ is the corresponding covariance matrix. Initial prior distribution,

$$\mathbf{x}_0 \sim \mathcal{N}(\mathbf{x}_0, \mathbf{P}_{0|0}) \tag{2.14}$$

The input matrix $\mathbf{B}_k$ models the actuators of the system, i.e. how the input signal $\mathbf{u}_k$ affects the state. The feedforward matrix $\mathbf{D}_k$ models how the input signal affects the measurement $\mathbf{y}_k$. The process model could be improved by including an input such as e.g. steering wheel angle. This would however make the estimate dependent on additional sensors. For this thesis it was prioritized to have an independent estimate, and thus $\mathbf{C}_k = \mathbf{D}_k = \mathbf{0}$.

The Kalman filter (KF) is a state estimator based on Bayesian statistics described in 2.3. Assuming a linear and Gaussian model described by Equation 2.10-2.14, it recursively computes the densities

$$p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}) \tag{2.15}$$

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}) \tag{2.16}$$

where Equation 2.15, 2.16 are the prediction and update step of the Kalman filter algorithm. The Gaussian densities of the prediction and update steps are fully described by their mean and its covariance.

- **Prediction step**:
  The prediction step computes a predicted mean value $\hat{\mathbf{x}}_{k|k-1}$ and a predicted covariance $\mathbf{P}_{k|k-1}$ based on the state estimate from $k-1$ i.e. the previous time instance. It predicts the evolution of the state based on the motion/process model $\mathbf{A}_k$. This step does not incorporate any information regarding the current time step k.

  The predicted mean and covariance

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}_k\hat{\mathbf{x}}_{k-1|k-1} \tag{2.17}$$

$$\mathbf{P}_{k|k-1} = \mathbf{A}_k\mathbf{P}_{k-1|k-1}\mathbf{A}_k^\top + \mathbf{Q}_k \tag{2.18}$$

- **Update step**:
  The update step of the Kalman filter utilizes the measurement $\mathbf{y}_k$, i.e. the new information obtained at the current time step k. The new information is captured in the innovation which is the residual between the observed and predicted measurement. When updating the mean, the Kalman gain scales the innovation according to the uncertainties in the measurements. A higher Kalman gain implies that the measurement is more reliable [11].

  The updated mean and covariance

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\tilde{\mathbf{e}}_k \tag{2.19}$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{C}_k)\mathbf{P}_{k|k-1} \tag{2.20}$$

  Innovation

$$\tilde{\mathbf{e}}_k = \mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_{k|k-1} \tag{2.21}$$

  Innovation covariance

$$\mathbf{S}_k = \mathbf{C}_k\mathbf{P}_{k|k-1}\mathbf{C}_k^\top + \mathbf{R}_k \tag{2.22}$$

  Kalman gain

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{C}_k^\top\mathbf{S}_k^{-1} \tag{2.23}$$

2.4.3 Extended Kalman Filter

The Extended Kalman Filter (EKF) is an extension of the Kalman filter to handle non-linear models. Consider that Equation 2.7, 2.8 are non-linear. The state space model is thus on the form

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}) + \mathbf{q}_{k-1} \tag{2.24}$$

$$\mathbf{y}_k = h_k(\mathbf{x}_k) + \mathbf{r}_k \tag{2.25}$$

$$\mathbf{q}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1}) \tag{2.26}$$

$$\mathbf{r}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \tag{2.27}$$

The idea behind the EKF is to linearize the non-linear process model $f_k(\mathbf{x}_{k-1})$ and non-linear measurement model $h_k(\mathbf{x}_k)$ to obtain a locally linearized system on which the normal Kalman filter prediction and update step can be applied. It is assumed to be a Gaussian approximation where the approximation stems from a Taylor series expansion [10].

- **Prediction step**:
  The predicted mean is computed by propagating the posterior mean $\hat{\mathbf{x}}_{k-1|k-1}$ through the non-linear process model $f(\cdot)$.

  The predicted mean and covariance

$$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}) \tag{2.28}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{\hat{x}_{k-1|k-1}} \mathbf{P}_{k-1|k-1} \mathbf{F}^{\top}_{\hat{x}_{k-1|k-1}} + \mathbf{Q}_{k-1} \tag{2.29}$$

  where $\mathbf{F}_{\hat{x}_{k-1|k-1}}$ is the Jacobian matrix

$$\mathbf{F}_{\hat{x}_{k-1|k-1}} = \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}_{k-1|k-1}} \tag{2.30}$$

- **Update step**:
  The update step of the EKF is similar to the KF, but it propagates the predicted mean $\hat{\mathbf{x}}_{k|k-1}$ through the non-linear measurement model $h(\cdot)$ to compute the innovation.

  The updated mean and covariance

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \tilde{\mathbf{e}}_k \tag{2.31}$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_{\hat{x}_{k|k-1}}) \mathbf{P}_{k|k-1} \tag{2.32}$$

  Innovation

$$\tilde{\mathbf{e}}_k = \mathbf{y}_k - h(\hat{\mathbf{x}}_{k|k-1}) \tag{2.33}$$

  Innovation covariance

$$\mathbf{S}_k = \mathbf{H}_{\hat{x}_{k|k-1}} \mathbf{P}_{k|k-1} \mathbf{H}^{\top}_{\hat{x}_{k|k-1}} + \mathbf{R}_k \tag{2.34}$$

  Kalman gain

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}^{\top}_{\hat{x}_{k|k-1}} \mathbf{S}_k^{-1} \tag{2.35}$$

  and where $\mathbf{H}_{\hat{x}_{k|k-1}}$ is the Jacobian matrix

$$\mathbf{H}_{\hat{x}_{k|k-1}} = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}_{k|k-1}} \tag{2.36}$$

2.5 Radar

A radar transmits an electromagnetic signal and interprets the received reflection to measure radial distance (range) to and relative velocity (Doppler) of the reflected surface. By using several transmitter (Tx) and receiver (Rx) antennas, direction of arrival (angle) of the target can be determined as well.

There are several radar realizations. For longer distances, pulse radars are common, where range is measured by the time difference between transmitting and receiving a pulse. Another radar type is Frequency Modulated Continuous Wave (FMCW). The transmitted signal frequency is repeatedly swept over a band in so-called chirps. From the shifts in phase and frequency of the received reflection, range and Doppler can be processed.

The FMCW instantaneous frequency $f(t) \in [f_0, f_0 + B]$ is swept from the starting frequency f_0 over the sweep bandwidth B, for the duration of the sweeping time T_C.

$$f(t) = \mu t + f_0, \quad \mu = \frac{B}{T_C} \tag{2.37}$$

where μ is called the chirp rate.

The Tx signal argument then becomes:

$$2\pi \int f(t)dt \tag{2.38}$$

$$= 2\pi \left[\frac{1}{2}\mu t^2 + f_0 t + C\right] \tag{2.39}$$

$$= 2\pi f_0 t + \pi \mu t^2 + \phi \tag{2.40}$$

The complete complex Tx signal, including phase ϕ:

$$x(t) = e^{j\left(2\pi f_0 t + \pi \mu t^2 + \phi\right)} \tag{2.41}$$

The reflected Rx signal has a time delay τ caused by the range r of the reflection:

$$\tau = \frac{2r}{c} \Longrightarrow \tag{2.42}$$

$$y_{\text{raw}}(t) = e^{j\left(2\pi f_0 (t-\tau) + \pi \mu (t-\tau)^2 + \phi\right)} \tag{2.43}$$

where c is the speed of light. As the time delay is in order of nanoseconds, it is hard to measure. However, because of the chirp, the time delay also means a shift between the Rx and Tx frequencies, called a beat frequency f_b:

$$f_b = \tau \mu = \frac{2\mu r}{c} \tag{2.44}$$

To measure the beat frequency, the Rx is first mixed with the Tx conjugate, $x^*(t)$. This corresponds to a convolution in frequency domain.

$$y(t) = y_{\text{raw}}(t)x^*(t) \tag{2.45}$$

$$= e^{\left(j2\pi f_0(t-\tau-t) + j\pi\mu(t-\tau)^2 - j2\pi\mu t^2 + j\phi - j\phi\right)} \tag{2.46}$$

$$= e^{j\left(-2\pi f_b t + \pi\mu\tau^2 - 2\pi f_0\tau\right)} \tag{2.47}$$

The argument terms of the signal are divided into the frequency part $(\pm)2\pi f_b t$, as it is dependent on time t, and the phase part $\pi\mu\tau^2 - 2\pi f_0\tau = \phi_y$. This sinusoid with frequency f_b in time domain corresponds to a peak at f_b in frequency domain. To distinguish between multiple reflections, a Discrete Fourier Transform (DFT) is used.

Relative velocity v of the target adds another, much smaller range Δ_r that is detectable over multiple chirps:

$$\Delta_r = vT_C \tag{2.48}$$

$$\Delta_\tau = \frac{2\Delta_r}{c} = \frac{2vT_C}{c} \tag{2.49}$$

Thus each received signal will have an additional delay of Δ_τ:

$$\phi_{y,1} = \pi\mu\tau^2 - 2\pi f_0\tau \tag{2.50}$$

$$\phi_{y,2} = \pi\mu(\tau+\Delta_\tau)^2 - 2\pi f_0(\tau+\Delta_\tau) \tag{2.51}$$

$$\Delta\phi = \phi_{y,2} - \phi_{y,1} = 2\pi\mu\tau\Delta_\tau + \pi\mu\Delta_\tau^2 - 2\pi f_0\Delta_\tau \tag{2.52}$$

$$\approx -2\pi f_0\Delta_\tau \tag{2.53}$$

$$= \frac{4\pi T_C f_0 v}{c} \tag{2.54}$$

The phase shift over n chirps:

$$\phi_n = \phi - 2\pi f_0 n\Delta_\tau \tag{2.55}$$

The beat frequency f_b changes slower than the phase ϕ_n and can be considered constant. Thus, a second DFT can be performed on the samples in each frequency bin of the former DFT. The spectrum from the first, "fast-time", DFT shows the f_b peaks that correspond to separate reflection ranges. The spectrum of the second, "slow-time", DFTs show the ϕ_n peaks that correspond to the relative velocity of each range bin. This is referred to as a Range-Doppler Map (RDM).

Furthermore, for a radar with multiple Rx antennas, a third DFT can be applied to find the direction of arrival angle of the corresponding reflection, to distinguish reflections with the same range and velocity. This is then called the radar cube. [12]

2.6 State - measurement relationships

The radar measurement contains two quantities, range and relative velocity, as described in Section 2.5. To develop an estimator, it needs to be decided which measurements to associate with which states. An intuitive approach would be to pair the time derivatives:

Quantity	Unit
Ego-pose	m, rad
Ego-velocity	m/s, rad/s
Range	m
Doppler	m/s

Table 2.3: States and measurement quantities

It could be possible to utilize the coupling of various states, e.g. that the vehicle generally has a negligible pitch angle at constant longitudinal velocity, to model other state-measurement relations. But these couplings are more difficult to derive, and likely do not cover all the states.

As the measurements are collected over time, it is also possible to integrate or differentiate them to produce accumulated radial distance y^{i} [m] from Doppler, or rate of radial distance change y^{d} [m/s] from range:

$$y^{\mathrm{i}}(t) = \int_{t-\Delta t}^{t} y_{\mathrm{Doppler}}(\tau) d\tau \qquad \Longleftrightarrow \qquad y_k^{\mathrm{i}} = \sum_{i=0}^{n} y_{k-i}^{\mathrm{Doppler}} \Delta t \tag{2.56}$$

$$y^{\mathrm{d}}(t) = \frac{d}{dt} y^{\mathrm{range}}(t) \qquad \Longleftrightarrow \qquad y_k^{\mathrm{d}} = \frac{y_k^{\mathrm{range}} - y_{k-1}^{\mathrm{range}}}{\Delta t} \tag{2.57}$$

This is however tricky, as the measurement noise produces a drift in y^{i} and is greatly amplified in y^{d}:

$$w_k \sim \mathcal{N}(0, \sigma^2) \tag{2.58}$$

$$y_k^{\mathrm{i}} = \sum_{i=0}^{n} \left(h(\mathbf{x}_{k-i}) + w_{k-i} \right) \Delta t = \cdots + \sum_{i=0}^{n} w_{k-i} \Delta t \tag{2.59}$$

$$y_k^{\mathrm{d}} = \frac{h(\mathbf{x}_k) + w_k - \left(h(\mathbf{x}_{k-1}) + w_{k-1} \right)}{\Delta t} = \cdots + \frac{w_k - w_{k-1}}{\Delta t} \tag{2.60}$$

Also, as y^{i} is a relative measurement, the initial radial distance would have to be known beforehand.

Furthermore, as the range and Doppler measurements can effectively be treated as separate sensors, it is desired to utilize both. Otherwise, to estimate all 9 motion states, 9 radars would be required.

In summary, it would appear worthwhile to first attempt modeling range as a function of ego-pose, and Doppler as a function of ego-velocity.

2.7 TruckMaker

IPG TruckMaker (TM) is a high fidelity vehicle dynamics simulation software, used for rapid prototyping, Hardware in the Loop (HIL) and Software in the Loop (SIL). Various sensors can be used in TruckMaker through its Raw Signal Interfaces (RSIs),

that simulate physical effects by propagation of the respective signal in the environment [13]. This allows for the extraction of range and Doppler velocity measurements. TruckMaker was used in this thesis project to enable testing several road scenarios and driving maneuvers, with various radar configurations and processing.

3
Method

3.1 Radar configuration

A radar R_j can be mounted on the ego-vehicle by defining its x-, y- and z-position with respect to vehicle body frame $\{V\}$. The position vector $\mathbf{t}_{\mathrm{j}R}^{V}$ describes this relation for each radar. The radar orientation is defined by an elevation angle α_j, i.e. the angle around the vehicle y-axis, and an azimuth angle β_j, i.e. the angle around the vehicle z-axis. This yields a unit vector $\hat{\mathbf{u}}_{\mathrm{j}}$ in Cartesian coordinates These vectors are depicted in Figure 3.1. A radar configuration is the pose (position and orientation) for each radar R_j for some number of radars $j = 1 \dots M$.

$$\mathbf{t}_{\mathrm{j}R}^{V} = \begin{bmatrix} x_{j_R}^{V} \\ y_{j_R}^{V} \\ z_{j_R}^{V} \end{bmatrix} \tag{3.1}$$

$$\hat{\mathbf{u}}_{\mathrm{j}} = \begin{bmatrix} cos(\alpha_j)cos(\beta_j) \\ cos(\alpha_j)sin(\beta_j) \\ -sin(\alpha_j) \end{bmatrix} \tag{3.2}$$

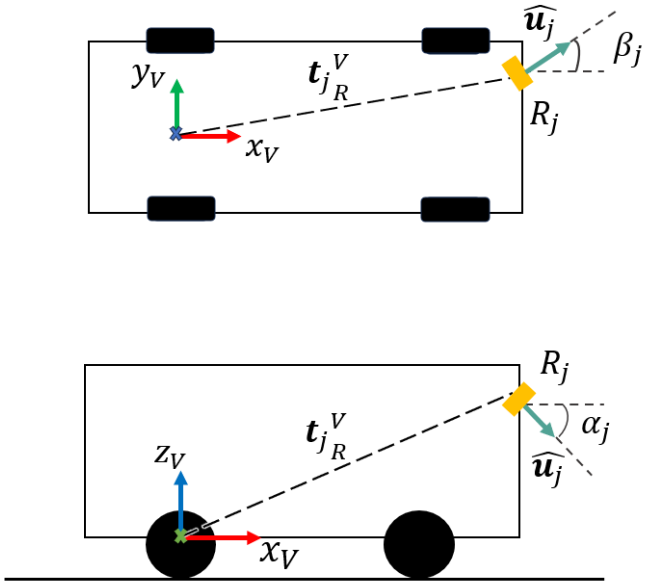

Figure 3.1: Illustration of azimuth β_j (upper) and elevation α_j (lower).

3.2 Measurement models

To estimate anything, a relationship between the state and the measurements need be modeled. For this thesis, the states of interest are the ego-pose and -velocity, and the measured quantities available are the radar range and Doppler velocity.

3.2.1 Doppler - ego-velocity

For the ego-velocity estimation the radar Doppler measurements is utilized. The radar measures the radial velocity of the reflected target and since it will measure a point on the ground surface, the reflected target will be stationary with respect to the radar. Assuming that the radar sensor and the vehicle body is one rigid object where each radar is mounted as defined in Section 3.1.

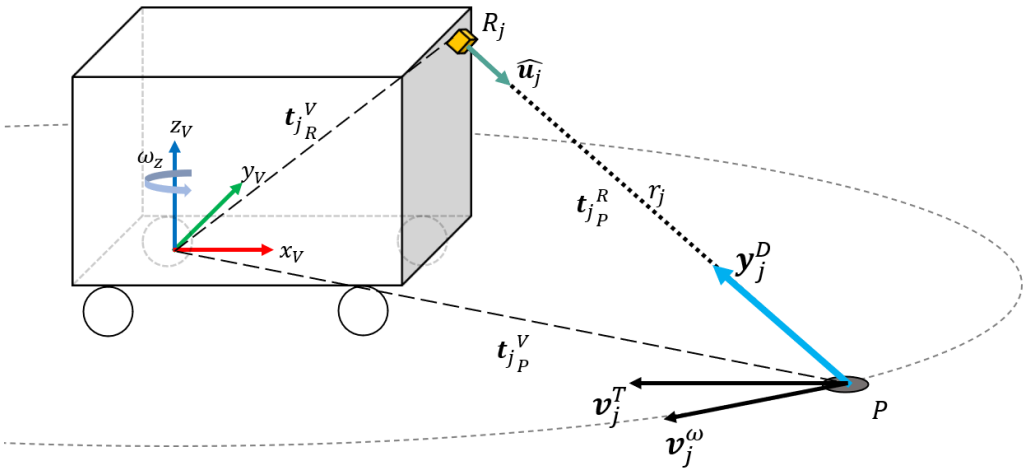

Figure 3.2: Geometric relationships between ego-vehicle, radar and observed point.

Using Equation 3.2 the translation vector from radar sensor to point is

$$\mathbf{t}_{\mathrm{j}P}^{R} = r_j \hat{\mathbf{u}}_{\mathrm{j}} \tag{3.3}$$

where r_j is the scalar range measurement. The total translation vector from vehicle frame to point is

$$\mathbf{t}_{\mathrm{j}P}^{V} = \mathbf{t}_{\mathrm{j}R}^{V} + \mathbf{t}_{\mathrm{j}P}^{R} \tag{3.4}$$

The observed point P experiences a translational velocity $\mathbf{v}^T$, and a tangential velocity $\mathbf{v}^\omega$, resulting from translational and angular ego-velocity respectively. The radar can only measure the projections of the sum of these velocities onto the Doppler velocity vector $\mathbf{y}_j^D$.

In this model, the three axes of rotation are approximated to intersect in a single point: the Center of Rotation (CoR). While the CoR depends on vehicle properties such as load and relative tire friction, the origin of the Vehicle frame can be chosen

arbitrarily, so the two do not necessarily coincide. That is the case in TruckMaker, where the vehicle-fixed frame, Fr1, is defined as the lowest rear point of the truck. For the model to be valid in such cases, an offset $\mathbf{t}^V_{\text{CoR}}$ needs to be defined. The vector $\mathbf{t}^{\text{CoR}}_P$ from the CoR to the reflection point P:

$$\mathbf{t}^V_P = \mathbf{t}^V_{\text{CoR}} + \mathbf{t}^{\text{CoR}}_P \tag{3.5}$$

$$\iff \mathbf{t}^{\text{CoR}}_P = \mathbf{t}^V_P - \mathbf{t}^V_{\text{CoR}} \tag{3.6}$$

If the Vehicle frame coincides with the CoR, then $\mathbf{t}^V_{\text{CoR}} = (0, 0, 0)$ and $\mathbf{t}^{\text{CoR}}_P = \mathbf{t}^V_P$.

The tangential velocity $\mathbf{v}^\omega_j$ is

$$\mathbf{v}^\omega_j = \mathbf{t}_{\text{j}P}^{\text{CoR}} \times \boldsymbol{\omega} \tag{3.7}$$

where $\boldsymbol{\omega}$ is the angular ego-velocities defined in Table 2.2. The order of the cross product reflects that a rotation of the vehicle corresponds to an opposite orbit of the stationary point.

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{3.8}$$

The translational velocity $\mathbf{v}^T_j$ is equal to the opposite velocity $\mathbf{v}$ at the body frame

$$\mathbf{v}^T_j = -\mathbf{v} \tag{3.9}$$

where $\mathbf{v}$ is the linear ego-velocities defined in Table 2.2.

$$\mathbf{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{3.10}$$

The projection of $\mathbf{v}^\omega_j$ and $\mathbf{v}^T_j$ on the Doppler measurement vector is

$$\mathbf{y}^D_j \hat{\mathbf{u}}_{\text{j}} = \mathbf{v}^\omega_j + \mathbf{v}^T_j \implies \tag{3.11}$$

$$\mathbf{y}^D_j = \hat{\mathbf{u}}_{\text{j}}^\top \cdot (\mathbf{v}^\omega_j + \mathbf{v}^T_j) \tag{3.12}$$

Simplifying (3.12), r_j drops out and left is a linear combination of the ego-velocity states:

$$\mathbf{y}^D_j = \mathbf{C}_j \mathbf{x}_{\text{velocity}} \tag{3.13}$$

where $\mathbf{C}_j$ is a 1×6 row vector. Using M radars, the measurements and $\mathbf{C}_j$ vectors are then concatenated vertically, $j = 1..M$:

$$\mathbf{y}^D = \begin{bmatrix} y^D_1 \\ \vdots \\ y^D_M \end{bmatrix} = \begin{bmatrix} C_1 \\ \vdots \\ C_M \end{bmatrix} \mathbf{x}_{\text{velocity}} = \mathbf{C}\mathbf{x}_{\text{velocity}} \tag{3.14}$$

Thus the Doppler measurement is a linear function of the ego-velocity state, represented by the measurement matrix $\mathbf{C}$. One row of $\mathbf{C}$, corresponding to one radar j, transposed for readability:

$$\mathbf{C}_j = \begin{bmatrix} -\cos(\alpha_j)\cos(\beta_j) \\ -\cos(\alpha_j)\sin(\beta_j) \\ \sin(\alpha_j) \\ (y_j - y_{\text{CoR}})\sin(\alpha_j) + (z_j - z_{\text{CoR}})\cos(\alpha_j)\sin(\beta_j) \\ -(x_j - x_{\text{CoR}})\sin(\alpha_j) - (z_j - z_{\text{CoR}})\cos(\alpha_j)\cos(\beta_j) \\ -(x_j - x_{\text{CoR}})\cos(\alpha_j)\sin(\beta_j) + (y_j - y_{\text{CoR}})\cos(\alpha_j)\cos(\beta_j) \end{bmatrix}^\top \tag{3.15}$$

Note that each row in the $\mathbf{C}$ matrix is only dependent on the radar mounting parameters for that corresponding radar.

3.2.2 Range - ego-pose

For the ego-pose estimation, the road is assumed to be planar. The proposed measurement model relates the transformation between the Vehicle frame $\{V\}$ and the Ground frame $\{G\}$, to the radar range measurements. Due to how $\{G\}$ is defined, only three ego-pose states are relevant: the heave position z, the roll angle φ_x and the pitch angle φ_y. Thus, the range-pose measurement model will be the range measurement $r_k = h(\mathbf{x}_{\text{pose}}) = h(z, \varphi_x, \varphi_y)$.

3.2.2.1 General model

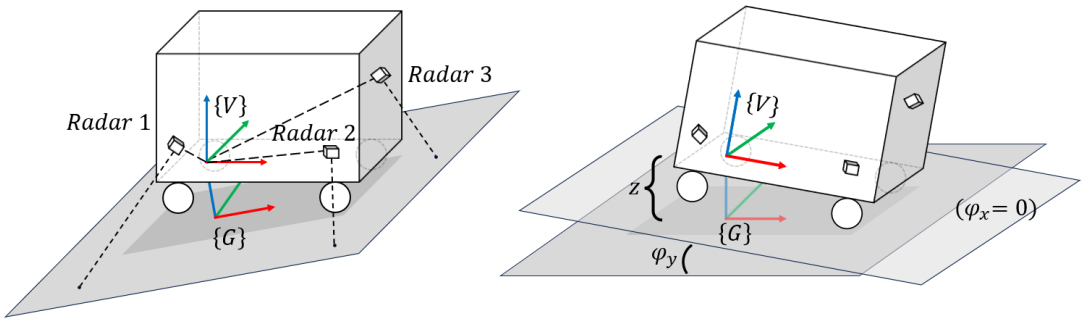

Figure 3.3: Range - ego-pose model

As per (3.1 - 3.2), a range measurement r_j can be interpreted as a point P_j in the road plane by

$$\mathbf{t}^V_{P_j} = \mathbf{t}^V_{R_j} + \mathbf{t}^{R_j}_{P_j} \tag{3.16}$$

$$= \mathbf{t}^V_{R_j} + r_j \hat{\mathbf{u}}_j \tag{3.17}$$

Three such points are needed to define the road plane w.r.t. the vehicle frame. The points cannot lie on the same line or point, else the solution is not unique. The

measured road plane is spanned by the xy axes of the ground frame. The ego-pose of the vehicle frame w.r.t. the ground frame can thus be calculated from a minimum of three range measurements.

The position of radar R_j expressed in $\{G\}$:

$$\mathbf{t}_{R_j}^G = \mathbf{t}_V^G + \mathbf{R}\mathbf{t}_{R_j}^V \tag{3.18}$$

$$= \begin{bmatrix} 0 \\ 0 \\ z \end{bmatrix} + \mathbf{R}\mathbf{t}_{R_j}^V \tag{3.19}$$

where the ego-orientation rotation matrix $\mathbf{R}$ is defined in (2.4).

The radar observation direction unit vector $\hat{\mathbf{u}}_\mathrm{j}$ w.r.t. $\{G\}$:

$$\hat{\mathbf{u}}_\mathrm{j}^{\{G\}} = \mathbf{R}\hat{\mathbf{u}}_\mathrm{j} \tag{3.20}$$

To get the projection of $\hat{\mathbf{u}}_\mathrm{j}^{\{G\}}$ onto the road plane, its z component is simply zeroed out. The resulting vector is then normalized to get the horizontal observation direction unit vector $\hat{\mathbf{v}}_j$:

$$\mathbf{v}_j = \hat{\mathbf{u}}_\mathrm{j}^{\{G\}} * \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \tag{3.21}$$

$$\hat{\mathbf{v}}_j = \mathbf{v}_j \frac{1}{\|\mathbf{v}_j\|} \tag{3.22}$$

where $*$ denotes element-wise multiplication.

The vertical position $z_{R_j}^G$ and elevation angle $\alpha_j^{\{G\}}$ of the radar w.r.t. $\{G\}$ can then be computed:

$$z_{R_j}^G = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{t}_{R_j}^G \tag{3.23}$$

$$\alpha_j^{\{G\}} = \arccos(\hat{\mathbf{v}}_j \cdot \hat{\mathbf{u}}_\mathrm{j}^{\{G\}}), \tag{3.24}$$

where $\cdot$ denotes inner product.

The range measurement r_j is then the hypotenuse of a right triangle with leg $z_{R_j}^G$ and opposing corner $\alpha_j^{\{G\}}$:

$$r_j = \frac{z_{R_j}^G}{\sin(\alpha_j^{\{G\}})} \tag{3.25}$$

Which is concatenated for multiple radars:

$$\begin{bmatrix} r_1 \\ r_2 \\ \vdots \end{bmatrix} = h\left(\mathbf{x}_\mathrm{pose}\right) = h\left(\begin{bmatrix} z_{R_1}^G / \sin(\alpha_1^{\{G\}}) \\ z_{R_2}^G / \sin(\alpha_2^{\{G\}}) \\ \vdots \end{bmatrix}\right) \tag{3.26}$$

3.2.2.2 Differential Janus model

Another approach would be to mount two radars in a pair on opposite sides of the ego-vehicle, with the same height and elevation angle. This configuration is referred to as a "Janus configuration". E.g. in the xz plane:

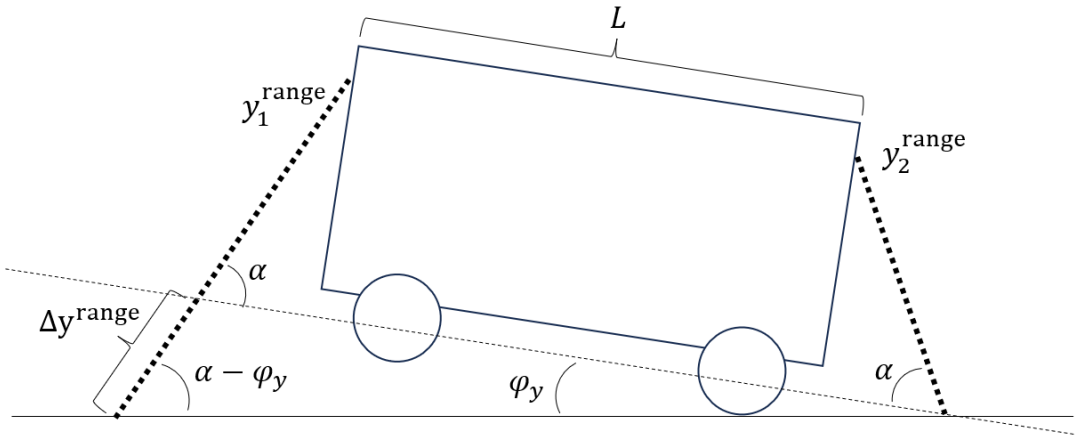

Figure 3.4: Geometric derivation of a 2D differential Janus model

The difference between the range measurements from the radar pair is a function of the ego-pitch angle φ_y:

$$\frac{\sin(\varphi_y)}{\Delta y^{\text{range}}} = \frac{\sin(\alpha - \varphi_y)}{L + 2\cos(\alpha)\min(y_1^{\text{range}}, y_2^{\text{range}})} \tag{3.27}$$

$$\iff \Delta y^{\text{range}} = y_1^{\text{range}} - y_2^{\text{range}} = \begin{cases} y_1^{\text{range}} \leq y_2^{\text{range}} : \dfrac{\sin(\varphi_y)(2\cos(\alpha)y_1^{\text{range}})}{\sin(\alpha - \varphi_y)} \\ y_1^{\text{range}} > y_2^{\text{range}} : \dfrac{\sin(\varphi_y)(2\cos(\alpha)y_2^{\text{range}})}{\sin(\alpha - \varphi_y)} \end{cases} \tag{3.28}$$

This is known as a differential measurement model $\Delta y = h(\mathbf{x})$. The above can be applied similarly for the ego-roll angle. However, as this method requires the radars to be mounted in a specific configuration, the general approach is favored.

3.2.3 Model validation

The range measurement model relates the range measurements to the ego-pose state and is described by Equation 3.26. There were no real range measurements available for comparison, hence TruckMaker range measurements were used.

Matlab simulated range measurements can be generated by propagating groundtruth pose states through the non-linear measurement model,

$$\mathbf{y}_{sim}^{range} = h_{range}(\mathbf{x}_{pose}) \tag{3.29}$$

where the groundtruth states can be obtained from TruckMaker

$$\mathbf{x}_{pose} = \begin{bmatrix} z \\ \varphi_x \\ \varphi_y \end{bmatrix} \tag{3.30}$$

Having a single radar mounted in the front during an accelerating maneuver, Figure 3.5 shows the corresponding groundtruth longitudinal velocity and pitch angle during the maneuver. A comparison between TM range measurements and Matlab simulated measurements can be seen in Figure 3.6. Note that when the pitch increases due to gear shifts, the range decreases, which is expected since a greater positive pitch makes the radar mounting position "lean" closer to the ground.

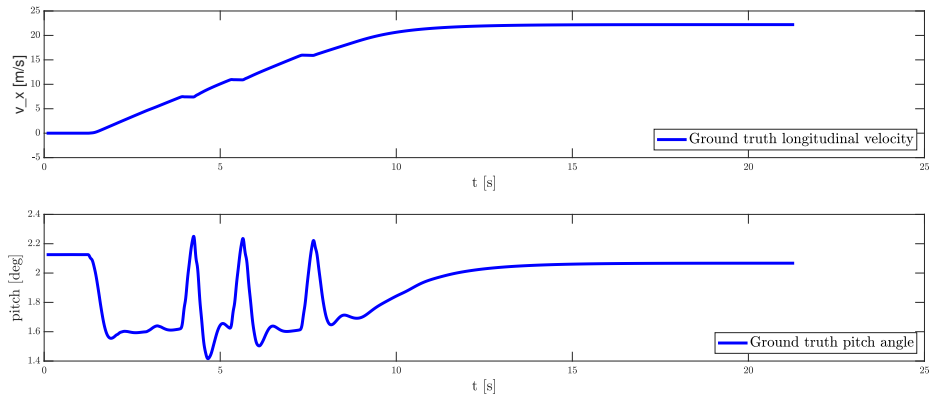

Figure 3.5: Groundtruth for longitudinal velocity v_x and pitch angle φ_y of vehicle during acceleration.

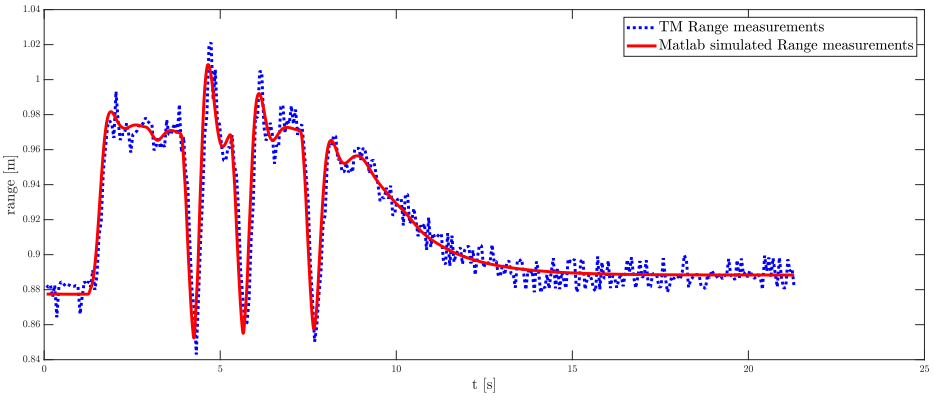

Figure 3.6: TM range measurements (dotted blue) and Matlab simulated measurement (red) comparison for one radar mounted in the front of the vehicle.

The Doppler measurement model relates the relative velocity measurements to the ego-velocity state, described by Equation 3.15. To investigate if the derived measurement model is correct, the Doppler velocity measurements $\mathbf{y}^D$ and the velocity

states are needed. The Doppler measurements can come from three sources:

1. Simulated measurements in Matlab, $\mathbf{y}_{sim}^{D}$
2. Real measurements, $\mathbf{y}_{real}^{D}$
3. Simulated measurements in TM, $\mathbf{y}_{TM}^{D}$, validated and described in 3.7.4

Matlab simulated Doppler measurements can be generated with groundtruth velocity

$$\mathbf{y}_{sim}^{D} = \mathbf{C}\mathbf{x}_{\text{velocity}} \tag{3.31}$$

where the groundtruth states can be obtained from either TruckMaker or the RT3000 groundtruth sensor unit.

To validate if the measurement model, Equation 3.15, is correct, comparing Matlab simulated measurements $\mathbf{y}_{sim}^{D}$ with real Doppler measurements $\mathbf{y}_{real}^{D}$ is one approach. This means modeling the same radar configuration as the real test and generating Matlab simulated measurements through Equation 3.31, and comparing the results.

In addition to those test, it is also possible to validate using TruckMaker. Figure 3.7, shows the Matlab simulated Doppler measurements, $\mathbf{y}_{sim}^{D}$, along with the TruckMaker measurements. It is the same maneuver as described by Figure 3.5. Since the radar is mounted in the front of the vehicle, and the maneuver is just straight acceleration, the Doppler measurement will have the opposite sign of the groundtruth longitudinal velocity. This can be seen by comparing to the top plot of Figure 3.5.

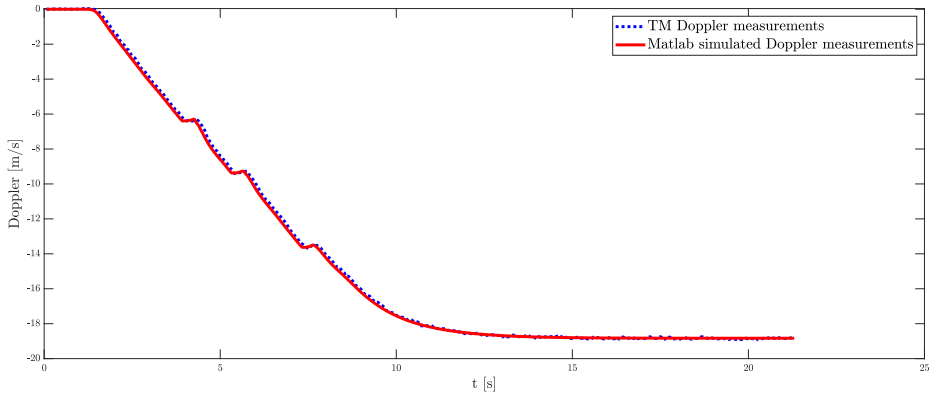

Figure 3.7: TM Doppler measurements (dotted blue) and Matlab simulated measurement (red) comparison for one radar mounted in the front of the vehicle.

3.3 Estimators

The two derived measurement models, Equation 3.26 and 3.15 allow for the latent variables, i.e. the ego-pose and -velocity states, to be estimated from the radar measurements. Which estimation technique that is chosen is due to its performance on a particular system.

3.3.1 Pseudo inversion

To invert a square matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, its inverse, $\mathbf{A}^{-1}$, exists if and only if the determinant $|\mathbf{A}| \neq 0$. For a non-square matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ the determinant does not exist and hence the inverse does not exist. Instead, the pseudo inverse $A^{\dagger}$ of A can be computed. If $\mathbf{A}$ is square and $|\mathbf{A}| \neq 0 \implies \mathbf{A}^{\dagger} = \mathbf{A}^{-1}$ i.e. the pseudo inverse gives the same answer as the regular inverse.

The pseudo inverse can act as a least squares solution when applied to a system of equations on the form $\mathbf{y} = \mathbf{A}\mathbf{x}$ and when $\mathbf{A}$ is not invertible. Considering an over-determined system of equations, i.e. $\mathbf{A} \in \mathbb{R}^{mxn}$ where $m > n$ and full column rank, i.e. its columns are linearly independent, the pseudo inverse is defined as

$$\mathbf{A}^{\dagger} = (\mathbf{A}^{\top}\mathbf{A})^{-1}\mathbf{A}^{\top} \tag{3.32}$$

which is called the left inverse of A since

$$\mathbf{A}^{\dagger}\mathbf{A} = (\mathbf{A}^{\top}\mathbf{A})^{-1}\mathbf{A}^{\top}\mathbf{A} = \mathbf{I} \tag{3.33}$$

Using the equation, referred to in literature as the normal equation, $\mathbf{A}^{\top}\mathbf{A}\mathbf{x} = \mathbf{A}^{\top}\mathbf{y}$ the least squares solution can be computed

$$\mathbf{x}_{LS} = (\mathbf{A}^{\top}\mathbf{A})^{-1}\mathbf{A}^{\top}\mathbf{y} = \mathbf{A}^{\dagger}\mathbf{y} \tag{3.34}$$

Looking at Equation 3.15 one can notice that the rows of $\mathbf{C}$ will vary with the number of radars used, $\mathbf{C} \in \mathbb{R}^{mxn}$. If using more radars than states $m > n$ or fewer radars than states $m < n$, $\mathbf{C}$ will not be square. Therefore, the pseudo inverse can be used as a state estimator regardless of matrix dimensions.

3.3.2 EKF

The Kalman filter equations defined in Section 2.4 have several parameters that need to be set and tuned. The state space model representation requires the KF/EKF to be initialized with an initial prior mean $\mathbf{x}_0$, a prior covariance $\mathbf{P}_{0|0}$, a process model $\mathbf{A}_k$, process noise covariance $\mathbf{Q}_k$ and measurement noise covariance $\mathbf{R}_k$.

- **Process model**:
  The process model, $\mathbf{A}_k$, describes how the state evolves over time, i.e. information about x_k given x_{k-1}. The model can be designed in various ways depending on application. The constant velocity (CV) model is one. It assumes that the state evolves at a constant rate between the samples. The process model for the velocity estimation is

$$\begin{bmatrix} \mathbf{v}_k \\ \boldsymbol{\omega}_k \end{bmatrix} = \underbrace{\mathbf{I}_{6x6}}_{\mathbf{A}_k} \begin{bmatrix} \mathbf{v}_{k-1} \\ \boldsymbol{\omega}_{k-1} \end{bmatrix} \tag{3.35}$$

  The process model for pose estimation is

$$\begin{bmatrix} z_k \\ \varphi_{x_k} \\ \varphi_{y_k} \end{bmatrix} = \underbrace{\mathbf{I}_{3x3}}_{\mathbf{A}_k} \begin{bmatrix} z_{k-1} \\ \varphi_{x_{k-1}} \\ \varphi_{y_{k-1}} \end{bmatrix} \tag{3.36}$$

- **Process noise covariance**:
  The process noise covariance matrix $\mathbf{Q}_k$ is a parameter in the Kalman filter that needs to be selected and tuned. The noise is a measure of how much the state can deviate from the deterministic value determined by the process model.

  The process noise can be chosen and tuned from real data. Using a real drifting test (RD 7), the covariance can be computed

$$\mathbf{Q}_k = \begin{bmatrix} q_1 & & \\ & \ddots & \\ & & q_n \end{bmatrix} \cdot T \tag{3.37}$$

  where

$$[q_1, \ldots, q_n] = \text{diag}(\text{cov}(\text{diff}(\mathbf{x}^\top))) \tag{3.38}$$

  where $\mathbf{x}$ is a groundtruth sequence. The diff command computes the difference between each consecutive element in $\mathbf{x}$. The cov command computes the covariance matrix from which the diagonal can be extracted. Thus, the variation of change in each of the states is represented in $q_1, \ldots, q_n$.

  The process noise covariance for the constant pose model is chosen and tuned from TM simulations. The noise can be tuned according to a general scenario, i.e. a scenario containing accelerating, driving, turning and braking. The drawback to this approach is that not every state experiences a moderate change during the scenario. An initial approach is instead to tune the state from a scenario where it is known to change. For example, a hard brake would change the pitch angle and heave position noticeably.

  Using a TM-test with hard braking the covariance can be computed as explained by Equation 3.37, 3.38. This yields a pose process noise covariance that is too low. This is seen by visually inspecting how the filter performs. From that, it is clear that the filter reacts slowly to changes in the states which motivates increasing $\mathbf{Q}_k$.
- **Measurement noise covariance**:
  The measurement noise covariance matrix $\mathbf{R}_k$ is noise in the radar sensors. The filter is tested on two noise figures

$$\mathbf{R}_k^{realistic} = 0.05^2 \tag{3.39}$$

$$\mathbf{R}_k^{high} = 0.5^2 \tag{3.40}$$

3.4 Performance measures

To evaluate the performance of the estimators and the radar configuration of each test, it needs to be quantified. To be able to compare large test matrices, of various estimator parameters, maneuvers, radar configurations, uncertainties, noise etc.,

the number of measures were kept to a minimum. Apart from the total variance for each state of the estimate error, it was important to capture any bias introduced by model errors or simplifications. Thus, the following two measures were chosen:

Mean error, $\bar{\mathbf{e}}$:

$$\bar{\mathbf{e}} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{e}_k = \frac{1}{K}\sum_{k=1}^{K}(\hat{\mathbf{x}}_k - \mathbf{x}_k) \tag{3.41}$$

A non-zero mean error implies that the estimate suffers from bias.

The Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{K}\sum_{k=1}^{K}\mathbf{e}_k^2} \tag{3.42}$$

The RMSE indicates how large the spread of the estimation error is.

3.5 Estimate sensitivity

The estimate sensitivity is how much the estimate error is affected by various disturbances and parameters. It is a form of risk assessment that is an important step in the estimator design process. Some error sources require compromise, e.g. in how to mount the radars, for others it might be necessary to limit the set of use-cases.

3.5.1 General sensor noise

All sensors have some amount of measurement noise, stemming from e.g. electrical variations and disturbance from other components, temperature fluctuation and mechanical vibrations. Specifically for radar sensors, there are also many factors to the signal propagation such as diffraction, refraction and attenuation varying with both the propagation medium (the air) and the reflection surface [14]. These factors are grouped together as a single noise distribution, that cannot be influenced.

3.5.2 Vehicle body flex

The proposed measurement models assume a rigid vehicle body. In reality, there is considerable flex between different parts of a truck. In particular the truck cab suspension, allowing it to pitch separately from the vehicle frame. Estimating the motion in vehicle frame with measurements from radars mounted on the cab frame will thus have an error.

3.5.3 Number of radars

Both measurement models are multivariate functions $h_{\text{range}}(\mathbf{x}_{\text{pose}})$, $h_{\text{Doppler}}(\mathbf{x}_{\text{velocity}})$ that relate n ego-state variables to a single measurement: $h : \mathbb{R}^n \to \mathbb{R}$. To uniquely estimate the state variables at each time instance, measurements from $\geq n$ radars

are then needed, to get $h^{-1} : \mathbb{R}^{n+} \to \mathbb{R}^n$.

For the ego-pose estimate, the radar reflections need to uniquely define the road plane. Thus, the minimum number of radars needed for ego-pose estimation is three, and the radars cannot be mounted in such a way that the reflections lie in a single point or line.

The ego-velocity states can be divided into two subsets: the ones in the road plane, $\mathbf{x}_{2D} = \{v_x, v_y, \omega_z\}$, and the remaining $\mathbf{x}_{3D} = \{v_z, \omega_x, \omega_y\}$. In normal driving scenarios, most of the dynamics are in the 2D states, and they are the most important to estimate. For some applications, the 3D states estimate might not be required, or it is not accurate enough. Could then $\mathbf{x}_{2D}$ be estimated using just three radars? In that case, the 3D states could be considered a disturbance and the sensitivity of $\hat{\mathbf{x}}_{2D}$ from $\mathbf{x}_{3D}$ is of interest. Does having redundancy in the number of radars improve the estimate?

To isolate these cases, the following combinations were investigated for the Doppler measurements:

$$\mathbf{y}_{3\times1} = C_{3\times3}\mathbf{x}_{3\times1} \tag{3.43}$$
$$\mathbf{y}_{6\times1} = C_{6\times3}\mathbf{x}_{3\times1} \tag{3.44}$$
$$\mathbf{y}_{3\times1} = C_{3\times6}\mathbf{x}_{6\times1} \tag{3.45}$$
$$\mathbf{y}_{6\times1} = C_{6\times6}\mathbf{x}_{6\times1} \tag{3.46}$$

And for the 2D state estimate:

$$\hat{\mathbf{x}}_{2D} = (C_{3\times3})^{-1}\mathbf{y}_{3\times1} \tag{3.47}$$
$$\hat{\mathbf{x}}_{2D} = (C_{6\times3})^{\dagger}\mathbf{y}_{6\times1} \tag{3.48}$$
$$\hat{\mathbf{x}}_{2D} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} (C_{6\times6})^{-1}\mathbf{y}_{6\times1} \tag{3.49}$$

These investigations are presented in 4.2.

3.5.4 Frames of reference

The proposed models are of ego-pose in ground frame and of ego-velocity in vehicle frame. More often, but not always, it is desired to know the ego-velocity in ground frame. If an ego-pose estimate is available, the ego-velocity estimate can be transformed into ground frame by

$$\mathbf{x}^{\{G\}}_{\text{velocity}} = \begin{bmatrix} \mathbf{v}^{\{G\}} \\ \boldsymbol{\omega}^{\{G\}} \end{bmatrix} = \begin{bmatrix} R(\boldsymbol{\varphi})\mathbf{v}^{\{V\}} \\ R(\boldsymbol{\varphi})\boldsymbol{\omega}^{\{V\}} \end{bmatrix} \tag{3.50}$$

where $R(\boldsymbol{\varphi})$ is the rotation matrix defined in (2.4).

The ego-roll and -pitch angles are nominally quite small, so it is possible to disregard the ego-pose estimation and instead use the ego-velocity estimate in vehicle frame as ground speed. E.g. for a 5° pitch angle, the longitudinal velocity in vehicle frame is:

$$v_x^{\{V\}} = \cos(5^\circ) \cdot v_x^{\{G\}} \tag{3.51}$$

$$= 0.9962 \cdot v_x^{\{G\}} \tag{3.52}$$

i.e. just 0.38% off. However, this is just one of several sources of error.

3.5.5 Radar configuration

To get consistent radar measurements, the radars need to be mounted so that they get good reflections from the road. At too shallow an elevation angle the signal will be attenuated and the risk increases of reflecting on some obstacle other than the road. Doppler measurements from another vehicle going the same speed for example, would be zero and interpreted as the ego-vehicle standing still.

The radar transmits and receives signals within a vertical and horizontal Field of View (FoV), which makes up a cone-shape with the radar at its tip. The intersection of the radar cone and the road surface forms an ellipsis, with dimensions depending on the height, elevation angle and FoV of the radar. For a vertical FoV θ, the range span Δr is

$$\Delta r = \frac{z^{\{G\}}}{\sin(\alpha - \theta/2)} - \frac{z^{\{G\}}}{\sin(\alpha + \theta/2)} \tag{3.53}$$

Which becomes considerable for low elevation angles, as shown in Figure 3.8:

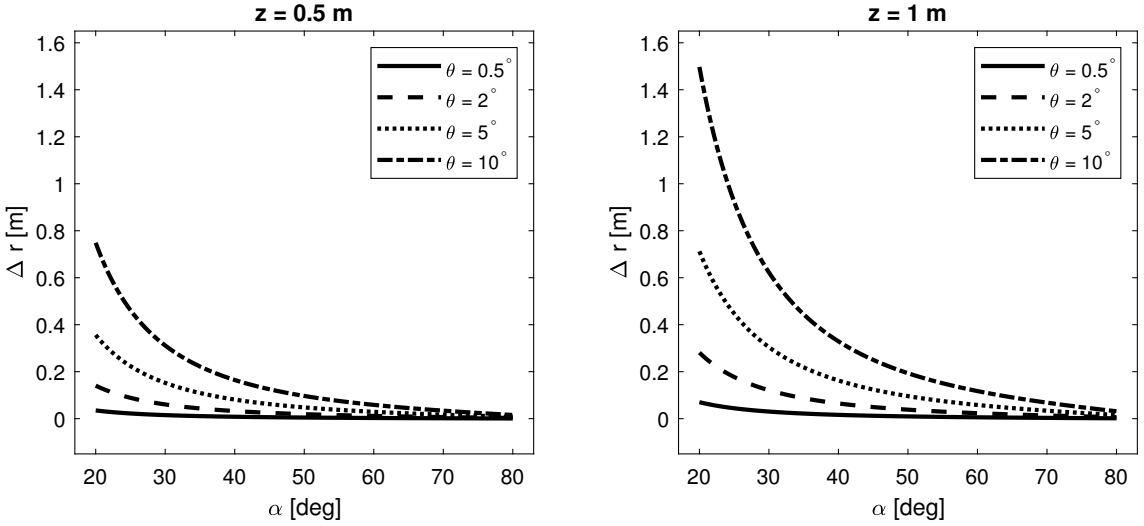

Figure 3.8: Range span for varying radar height, elevation angle and vertical FoV

However, by mounting the antennas in an 2D grid configuration, both elevation and azimuth angles of the received signal can be determined. Together with the range

information, this constitutes a 3D position of the point of reflection, in spherical coordinates. There is still a practical maximum angle resolution to consider.

For the Doppler-velocity measurement model, 3.15, the radars need to be mounted as to maximize the (absolute value of the) projections of the state variables. The projection is at its maximum when the observation vector of the radar is parallel to the measured velocity vector. Consequently, the projection is zero when they are perpendicular. If a state variable is not captured by any of the radars, the C matrix loses rank, as the column corresponding to that state will be $\mathbf{0}$.

As the projections depend on the radar poses but there is independent sensor noise, the radar configuration directly influences Signal-to-Noise Ratio (SNR):

$$y_{\text{Doppler}} = C\mathbf{x}_{\text{velocity}} + w, \quad w \sim \mathcal{N}(0, \sigma_{\text{Doppler}}) \tag{3.54}$$
$$\hat{\mathbf{x}}_{\text{velocity}} = C^{-1} y_{\text{Doppler}} \tag{3.55}$$
$$= C^{-1}\left(C\mathbf{x}_{\text{velocity}} + w\right) \tag{3.56}$$
$$= \mathbf{x}_{\text{velocity}} + C^{-1}w \tag{3.57}$$

As (3.57) shows, the influence of noise on the measurement depends on the properties of C^{-1}. It is therefore of interest to look at the condition number for inversion κ of the C matrix:

$$\kappa = \|C\|_2 \cdot \|C^{-1}\|_2 \tag{3.58}$$

The 2-norm, or spectral norm, of a matrix is defined as its largest singular value. The singular values describe how much the linear mapping C scales the state space X onto the observation space Y. Consequently, the condition number $\kappa \in [1, \infty)$ shows how much errors are amplified in the measurement.

There is no way to isolate every ego-velocity state, i.e. to make the C matrix diagonal. In fact, only v_z can be isolated, by:

$$x_j = x_{CoR} \wedge y_j = y_{CoR} \wedge \alpha_j = 90^\circ \tag{3.59}$$

which makes c_{j3} the only non-zero element of row C_j. To prove this, first note that due to $\alpha_j > 0$, no radar can be mounted parallel to v_x or v_y. For $\alpha_j = 90^\circ$, all terms containing a factor $\cos(\alpha_j)$ is zero. The remaining elements are then

$$c_{j4} = (y_j - y_{CoR})\sin(\alpha_j) + (z_j - z_{CoR})\cos(\alpha_j)\sin(\beta_j) \tag{3.60}$$
$$c_{j5} = -(x_j - x_{CoR})\sin(\alpha_j) - (z_j - z_{CoR})\cos(\alpha_j)\cos(\beta_j) \tag{3.61}$$

which are 0 for $y_j = y_{CoR}$ and $x_j = x_{CoR}$ respectively.

The performance of the ego-velocity estimate is highly dependent on the choice of radar configuration, which for 6 or more becomes non-trivial. An attempt at finding a good configuration is made in 3.6. The ego-pose estimate is less sensitive to radar configuration. To estimate both ego-roll and -pitch, it needs detections in both the longitudinal and lateral dimensions, but a radar configuration that yields an ego-velocity estimate should also yield an ego-pose estimate.

3.5.6 Miscalibration

The Doppler - ego-velocity model is a function of both the ego-velocity state variables and the radar configuration parameters $\mathbf{r}$:

$$\mathbf{y}_{\text{Doppler}}(t) = C\mathbf{x}_{\text{velocity}}(t) \tag{3.62}$$
$$= h(\mathbf{x}_{\text{velocity}}(t); \mathbf{r}) \tag{3.63}$$
$$\mathbf{r} = x_j, y_j, z_j, \alpha_j, \beta_j, \quad j = 1..M \tag{3.64}$$

Miscalibration is when the modeled radar configuration parameters $\mathbf{r}^{\text{model}}$ do not match the actual configuration $\mathbf{r}^{\text{actual}}$. As the function is linear in the state but nonlinear in the radar configuration parameters, the effect from parameter deviation $\Delta\mathbf{r}$ is not simply a superposition:

$$h(\mathbf{x}_{\text{velocity}}; \mathbf{r} + \Delta\mathbf{r}) \neq h(\mathbf{x}_{\text{velocity}}; \mathbf{r}) + h(\mathbf{x}_{\text{velocity}}; \Delta\mathbf{r}) \tag{3.65}$$

Consider the contribution from v_x to a radar with azimuth angle $\beta = 0$, actual elevation angle α and a model elevation angle deviation $\Delta\alpha$:

$$y_{\text{Doppler}} = c_{11}v_x + \cdots = \cos(\alpha)\cos(\beta)v_x + \ldots \tag{3.66}$$
$$= \cos(\alpha)v_x + \ldots \tag{3.67}$$
$$\hat{v_x} = {c_{11}}^{-1} y_{\text{Doppler}} = \frac{1}{\cos(\alpha + \Delta\alpha)} y_{\text{Doppler}} \tag{3.68}$$
$$= \frac{\cos(\alpha)}{\cos(\alpha + \Delta\alpha)} v_x \tag{3.69}$$

The normalized error e is then:

$$e = \frac{\hat{v_x} - v_x}{v_x} = \frac{\cos(\alpha)}{\cos(\alpha + \Delta\alpha)} - 1 \tag{3.70}$$

So the higher the elevation angle α, the more sensitive the estimate is to elevation deviation $\Delta\alpha$. This is because the change in the measurement function w.r.t. the elevation angle is convex and at its greatest at $\alpha = 0$:

$$\frac{\partial|h|}{\partial\alpha} = |-\sin(\alpha)|v_x + \ldots \tag{3.71}$$

To minimize the ego-velocity estimate sensitivity to miscalibration, the goal is to find a radar configuration that minimizes the Jacobian

$$J = \frac{\partial C}{\partial \mathbf{r}} \tag{3.72}$$

However as noted previously, there are other optimization objectives for the radar configuration. If possible, it is therefore preferable to reduce $\Delta\mathbf{r}$ by precise calibration than to reduce the sensitivity to miscalibration.

3.5.7 Road reflections

Electro-Magnetic (EM) waves can propagate through various media. Reflections occur when the wave passes between media with different relative permittivity $\epsilon(\omega)$ which depends on the frequency of the signal. The goal is to produce reliable reflections from targets of interest, while passing through clutter such as precipitation and dust. While automotive radars commonly use 77 GHz for this reason, they are designed for reflections from other vehicles, pedestrians and road features such as railings. For this application the desired target is only roads of varying material and Road Surface Conditions (RSC), i.e. dry/wet asphalt, ice, snow etc.

Another potential source of range measurement error is reflections inside the road materials. Road asphalt thickness can vary between 10 and 30 cm, and there could be additional layers such as water, ice or snow. If a signal bounces within the layers before returning to the receiver, the resulting range measurement would be erroneously large.

Obstacles or irregularities in the road would have an effect on the estimate. A pothole for example, would yield a faulty range from one or several radars, which would in turn yields a faulty ego-pose estimate. The effect on the estimate depends on the duration of the disturbance, and if there are correct measurements from other radars to compensate. The Doppler measurements should not be affected, as long as the obstacle has the same relative velocity as the road, i.e. stationary.

3.5.8 Signal processing and filtering

The radar signal processing has many aspects to consider. Firstly, with FFTs time and frequency resolution is inversely related. For a radar this translates to accuracy of the radar cube vs accuracy in time. Sacrificing time resolution by increasing the number of FFT samples will increase range resolution, but as the radars are mounted on a moving vehicle, the range will be an average over the road traversed in a longer time duration. There is also a limit on what resolutions are practically feasible with respect to computational requirements. For the estimates to be useful they need to be available in near real-time.

3.6 Radar configuration design

To optimize the radar configuration for ego-velocity estimation, the following heuristic process is proposed:

1. Maximize the absolute values of the diagonal elements of the C matrix
2. Choose the free parameters as to minimize the absolute values of the off-diagonals
3. When there is a trade-off between different off-diagonals, prioritize minimizing the nominally larger ones (e.g. yaw rate over roll rate)
4. Consolidate practical limitations

In practice:

Naturally, a radar can't be positioned below the road plane, and its observation line needs to intersect the road plane. To simplify the following equations, the radar configuration is here given with respect to a Vehicle frame with origin at the center lowest rear point of the truck. Given vehicle dimensions (L, W, H), the range of the radar pose is thus:

$$\begin{cases} x_j \in [0, L] \\ y_j \in [-W, W] \\ z_j \in (0, H] \\ \alpha_j \in (0, 90^\circ] \\ \beta_j \in (-180^\circ, 180^\circ] \end{cases}, \tag{3.73}$$

$$j = 1, .., 6 \tag{3.74}$$

The radars also need to be mounted on the outside of the vehicle. In a case where the CoR is not at the center of the vehicle, there will be a face closer to the CoR and an opposite face farther from the CoR. Denote the x position at the closer face $\Delta_{\min} x$, and at the farther face $\Delta_{\max} x$. I.e. if $x_{CoR} < L/2$, then $\Delta_{\min} x = 0, \Delta_{\max} x = L$ and vice versa. The same applies for y and z.

For the first diagonal element, the contribution to the Doppler measurement y_1^{Doppler} from longitudinal velocity v_x is

$$y_1^{\text{Doppler}} = c_{11} v_x + \cdots = -\cos(\alpha_1)\cos(\beta_1) v_x + \ldots \tag{3.75}$$

$$\text{argmax}|c_{11}| = \alpha_1 \to 0 \wedge \beta_1 = 90^\circ \pm 90^\circ \tag{3.76}$$

The free parameters of the first radar is $\{x_1, y_1, z_1, \alpha_1, \beta_1\} \setminus \{\alpha_1, \beta_1\} = \{x_1, y_1, z_1\}$.

$$C_1\Big|_{\alpha_1=0, \beta_1=0} = \begin{bmatrix} -1 & 0 & 0 & 0 & z_{CoR} - z_1 & y_1 - y_{CoR} \end{bmatrix} \tag{3.77}$$

A problem here is that $\alpha_1 \to 0 \implies z_1 \to 0$, i.e. that the lower the radar elevation angle is, the lower the radar height has to be, to avoid measuring too far from the vehicle. For now, introduce $z_{\min}$ and $\alpha_{\min}$. These limits will depend on external factors, as previously noted.

Continuing this process for each radar, produced (e.g.) the following radar con-

figuration:

$$\mathbf{r}^* = \begin{bmatrix} x_1 & y_1 & z_1 & \alpha_1 & \beta_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & \alpha_n & \beta_n \end{bmatrix} \tag{3.78}$$

$$= \begin{bmatrix} \Delta_{\min}x & y_{CoR} & z_{\min} & \alpha_{\min} & 0 \\ x_{CoR} & \Delta_{\min}y & z_{\min} & \alpha_{\min} & 90° \operatorname{sgn}(\Delta_{\min}y) \\ x_{CoR} & y_{CoR} & z_{\min} & 90° & \cdot \\ x_{CoR} & \Delta_{\max}y & z_{\max} & \min(\xi_1, 180° - \xi_1) & \xi_2 \\ \Delta_{\max}x & y_{CoR} & z_{\max} & \min(\xi_3, 180° - \xi_3) & \xi_4 \\ \Delta_{\min}x & \Delta_{\min}y & z_{\min} & \alpha_{\min} & \xi_7 \end{bmatrix} \tag{3.79}$$

$$\xi_1 = \left| \tan^{-1}\left(\frac{\Delta_{\max}y - y_{CoR}}{z_{\max} - z_{CoR}} \right) \right| \tag{3.80}$$

$$\xi_2 = \begin{cases} \xi_1 > 90° \wedge \Delta_{\max}y < y_{CoR} : 90° \\ \xi_1 \leq 90° \wedge \Delta_{\max}y \geq y_{CoR} : 90° \\ \text{else} : -90° \end{cases} \tag{3.81}$$

$$\xi_3 = \left| \tan^{-1}\left(\frac{\Delta_{\max}x - x_{CoR}}{z_{\max} - z_{CoR}} \right) \right| \tag{3.82}$$

$$\xi_4 = \begin{cases} \xi_3 > 90° \wedge \Delta_{\max}x < x_{CoR} : 0 \\ \xi_3 \leq 90° \wedge \Delta_{\max}x \geq x_{CoR} : 0 \\ \text{else} : 180° \end{cases} \tag{3.83}$$

$$\xi_5 = -\operatorname{atan2}\left(\frac{\Delta_{\min}x - x_{CoR}}{\Delta_{\min}y - y_{CoR}} \right) + \xi_6 \tag{3.84}$$

$$\xi_6 = \begin{cases} \Delta_{\min}x > x_{CoR} \wedge \Delta_{\min}y > y_{CoR} : 180° \\ \Delta_{\min}x \leq x_{CoR} \wedge \Delta_{\min}y \leq y_{CoR} : -180° \\ \text{else} : 0 \end{cases} \tag{3.85}$$

This configuration places the third radar underneath the vehicle. If this is not possible, it should be moved in the xy-plane to the closest valid position.

With an example CoR offset of $(L/2, 0, 1)$, both $\Delta_{\min}x$ and $\Delta_{\max}x$ can either be 0 or L, and both $\Delta_{\min}y$ and $\Delta_{\max}y$ can either be $-W$ or W. Assuming $z_{\min} \leq z_{CoR} \leq z_{\max}$:

$$\mathbf{r}^* = \begin{bmatrix} L & 0 & z_{\min} & \alpha_{\min} & 0 \\ L/2 & W & z_{\min} & \alpha_{\min} & 90° \\ L/2 & 0 & z_{\min} & 90° & 0 \\ L/2 & W & z_{\max} & \tan^{-1}\left(\dfrac{W}{z_{\max} - 1} \right) & 90° \\ L & 0 & z_{\max} & \tan^{-1}\left(\dfrac{L/2}{z_{\max} - 1} \right) & 0 \\ L & W & z_{\min} & \alpha_{\min} & 180° - \tan^{-1}\left(\dfrac{L/2}{W} \right) \end{bmatrix} \tag{3.86}$$

And for e.g. $L = 6, W = 1.3, z_{\min} = 0.5, z_{\min} = 1.5, \alpha_{\min} = 30°$, gives the C matrix:

$$C = \begin{bmatrix} -0.866 & 0 & 0.5 & 0 & -1.067 & 0 \\ 0 & -0.866 & 0.5 & 0.217 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -0.359 & 0.9333 & 1.3928 & 0 & 0 \\ 0.1644 & 0 & 0.9864 & 0 & -3.0414 & 0 \\ 0.3443 & -0.7946 & 0.5 & 0.2527 & -1.6722 & -2.8315 \end{bmatrix} \tag{3.87}$$

which has a condition number of 6.1942.

Some context is needed in order to assess this C matrix:

1. The first radar has greater contribution from ω_y than from v_x. However, nominally $v_x \in [0, 30]$ m/s while $\omega_y \in [-5°, 5°]$. At low speeds much pitch rate is not likely produced, but C_1 suggests that the v_x estimate risks bad performance at rapid acceleration or braking.
2. The last radar is the sole one with a contribution from yaw rate, but it is also disturbed by contribution from all the other states. A good yaw rate estimate is not likely.

3.7 TruckMaker

The vehicle model used is a 2-axle semi truck 4x2 with a rigid body and no trailer attached. It is one of the standard trucks provided in the program. Additional parameters like suspension and tires are left in their default settings. In TruckMaker, the vehicle frame origin is located in the bottom rear of the vehicle, shown in Figure 3.9. Its dimensions are $(6, 1.3, 3.8)$ m and its CoR offset was found empirically to be approximately $(3.9, 0, 1)$ m.

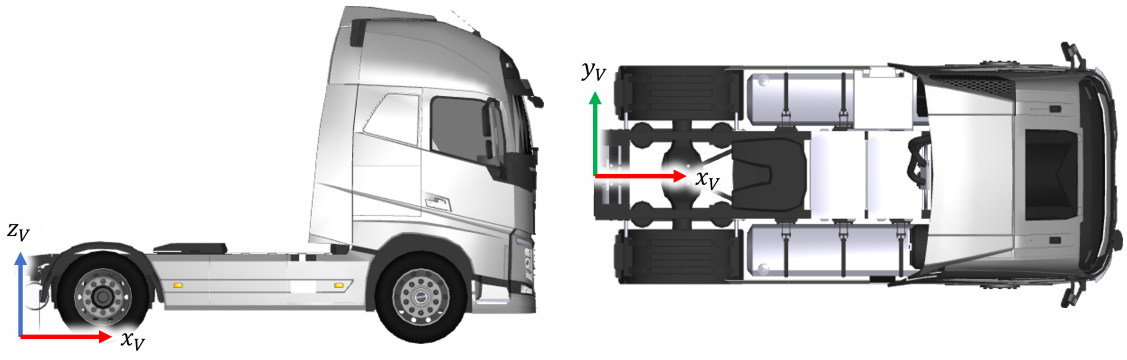

Figure 3.9: TruckMaker Vehicle frame definition

TruckMaker exposes its simulated states as User Accessible Quantities (UAQs). To get the groundtruth for both the ego-velocity states in Vehicle frame and the ego-pose states in Ground frame, some intermediate steps were taken:

Groundtruth	Intermediate	TruckMaker UAQ	Unit
z		$*$	m
φ_x		`Car.ConA.Roll_X`	rad
φ_y		`Car.ConA.Pitch_X`	rad
v_x		`Car.vx`	m/s
v_y		`Car.vy`	m/s
v_z		`Car.vz`	m/s
ω_x		`Car.RollVel`	rad/s
ω_x		`Car.PitchVel`	rad/s
ω_x		`Car.YawVel`	rad/s
	$\mathbf{t}_V^W$	`Vehicle.Fr1A.t_0`	m
	$\mathbf{t}_G^W$	`Vehicle.FrX.t_0`	m
	R_W^G	`Vehicle.FrX.Tr2Fr0`	

Table 3.1: TruckMaker groundtruth quantities

$*$ As the vehicle heave position in Ground frame was not available as a UAQ, it was calculated from:

$$z = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \left(R_W^G\right)^\top \left(\mathbf{t}_V^W - \mathbf{t}_G^W\right) \tag{3.88}$$

3.7.1 Radar RSI Sensor

The Radar RSI sensor provides realistic simulations of real FMCW radars. It models the whole sensor system, including physics-based polarimetric wave propagation, antenna characteristics and signal processing [13].

The radar cone is discretized by a number of rays at equidistant angles. The power of each ray is calculated from the transmitter gain map:

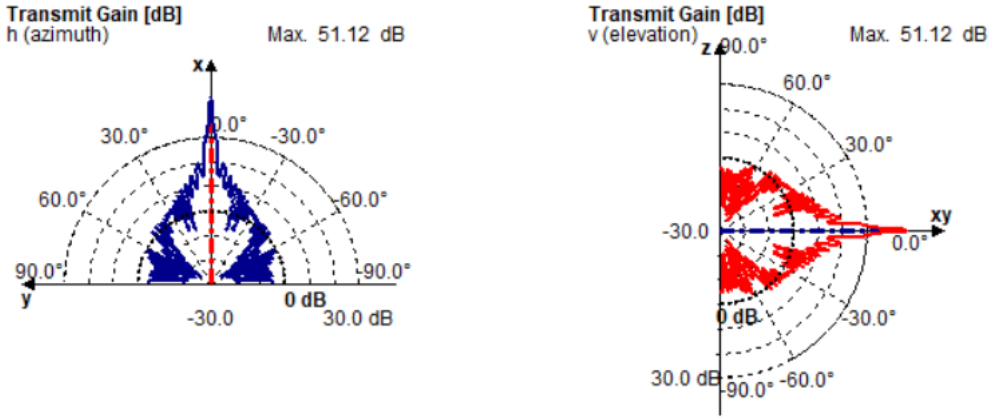

Figure 3.10: Transmitter gain map w.r.t. the radar frame

The radar model is configurable through various hardware and processing parameters. These were tuned to yield good range and Doppler measurements for the specific application of the thesis. The range of the parameter values were also limited, to keep the radar model realistically feasible. For example, avoiding zero noise or infinite max range. The parameters were tuned from their default values using tests from the test set, and inspecting the distribution of the measurements as well as the resulting state estimates. Ultimately, the values in Table 3.2 were used for all radars.

Horizontal transceiver FoV	0.5°
Vertical transceiver FoV	0.5°
Horizontal rays	8
Vertical rays	10
Central transmit frequency	78.1 GHz
Transmit power	12.5 dBm
Noise figure	14 dB
Receiver noise bandwidth	200 MHz
Receiver noise temperature	300 K
Minimum range	0 m
Maximum range	5.5 m
Minimum Doppler	-30 m/s
Maximum Doppler	30 m/s
Fast-time (range) FFT samples	512
Fast-time (range) FFT paddings	256
Slow-time (Doppler) FFT samples	512
Slow-time (Doppler) FFT paddings	128
OS-CFAR layers	5
OS-CFAR guards	2
OS-CFAR SnR	10 dB
OS-CFAR threshold	90 %

Table 3.2: TruckMaker radar sensor design parameters

A single radar was mounted on the front of the vehicle at $x_1 = 6, y_1 = 0, z_1 = 1, \alpha_1 = 45°, \beta_1 = 0$. Figure 3.11 shows the range measurements from a stationary test:

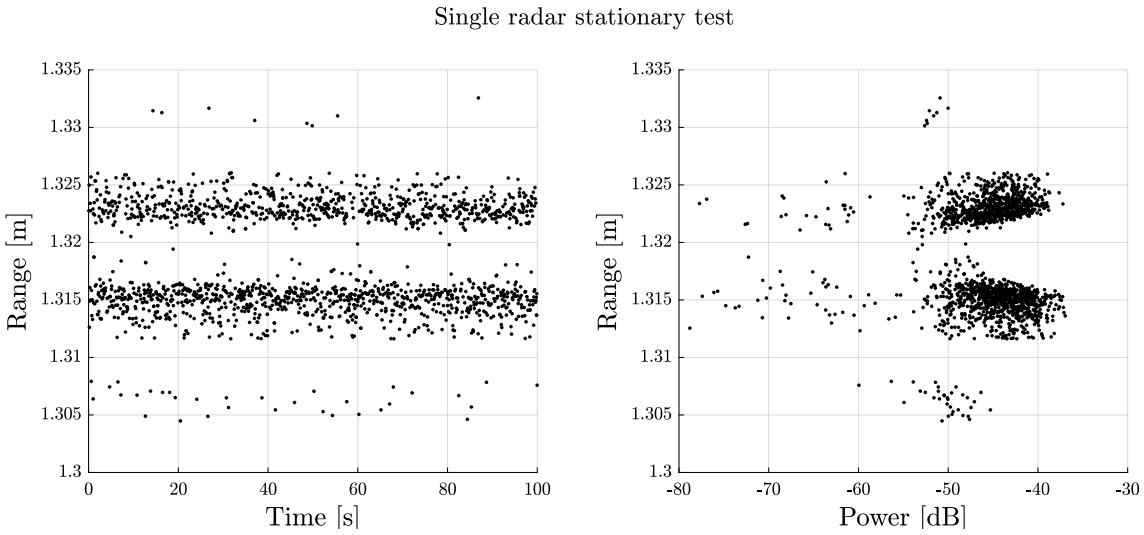

Figure 3.11: Range over time and power for a stationary test

Steady state groundtruth:

$$\mathbf{x}_{\text{velocity}} = \begin{bmatrix} 0.1895 & 0 & 2.1257 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^\top \tag{3.89}$$

Using the range - pose measurement model, the expected range is:

$$\begin{aligned} z_1^G &= -\sin(\varphi_y) \cdot x_1 + \cos(\varphi_y) \cdot z_1 + z && (3.90)\\ &= -\sin(2.1257^\circ) \cdot 6 + \cos(2.1257^\circ) \cdot 1 + 0.1895 && (3.91)\\ &= 0.9663 \text{ m} && (3.92)\\ \alpha_1^G &= \alpha_1 + \varphi_y = 45^\circ + 2.1257^\circ && (3.93)\\ &= 47.1257^\circ && (3.94)\\ y_{\text{range}} &= \frac{z_1^G}{\sin(\alpha_1^G)} = \frac{0.9663}{\sin(47.1257^\circ)} && (3.95)\\ &= 1.3186 \text{ m} && (3.96) \end{aligned}$$

Caution is warranted here, as this is a circular reasoning: the measurement model is derived to match the TruckMaker radar model, but at the same time the TruckMaker radar model is tuned to match the measurement model.

Figure 3.12 shows the Doppler measurements from a test where the truck drove with a constant ground speed of 30 km/h on a straight road:

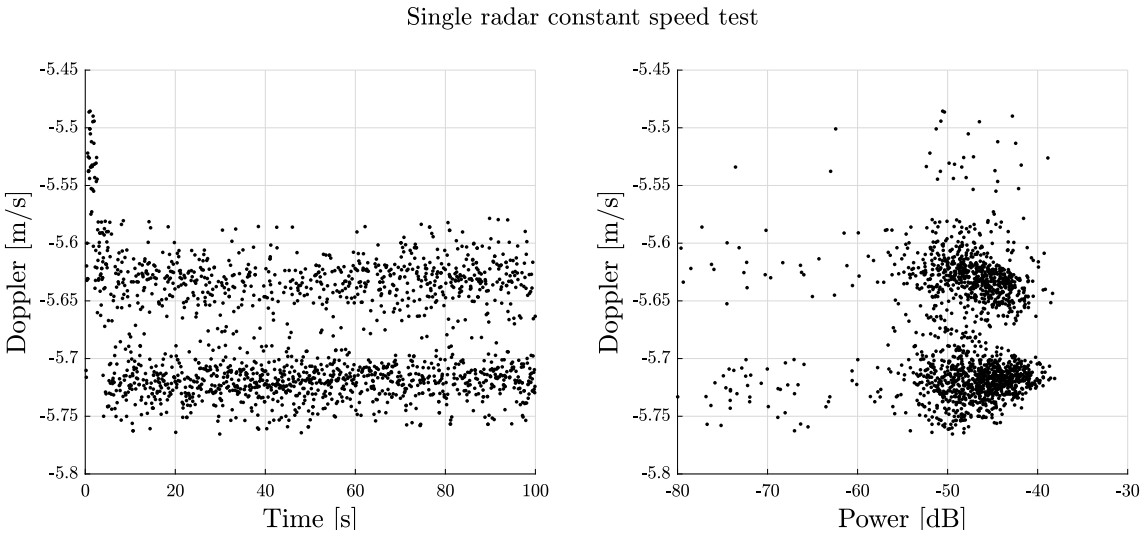

Figure 3.12: Doppler over time and power for a straight, constant velocity test

Steady state groundtruth:

$$\mathbf{x}_{\text{velocity}} = \begin{bmatrix} 0.1891 & 0 & 2.1085 & 8.3277 & 0 & 0.3065 & 0 & 0 & 0 \end{bmatrix}^\top \tag{3.97}$$

Note that $v_x = 8.3277$ m/s $= 29.9797$ km/h because the ego-translation is in vehicle frame.

$$\begin{aligned} v_x^G &= \cos(\varphi_y) \cdot v_x + \sin(\varphi_y) \cdot v_z && (3.98)\\ &= \cos(2.1085°) \cdot 8.3277 + \sin(2.1085°) \cdot 0.3065 && (3.99)\\ &= 8.3333 \text{ m/s} = 30 \text{ km/h} && (3.100) \end{aligned}$$

The resulting C "matrix":

$$C = \begin{bmatrix} -0.7071 & 0 & 0.7071 & 0 & -1.4849 & 0 \end{bmatrix} \tag{3.101}$$

Using the Doppler - velocity measurement model, the expected range is:

$$\begin{aligned} y_{\text{Doppler}} &= C\mathbf{x}_{\text{velocity}} && (3.102)\\ &= -0.7071 \cdot 8.3277 + 0.7071 \cdot 0.3065 && (3.103)\\ &= -5.6718 \text{ m/s} && (3.104) \end{aligned}$$

As noted in 3.5.5, it is important to know the position of the reflection. Simulating and processing the signals from multiple RSI radar sensors is however computationally expensive. To reduce simulation time, the radars were instead modelled with a single antenna but having an FoV corresponding to a reasonable angle resolution of 0.5°. This then removes the angle dimension from the radar cube, which means one fewer FFT needs to be performed (two for both elevation and azimuth).

The radar sensors were mounted on the ego-vehicle chassis by specifying each radar

pose w.r.t. the vehicle frame, as according to the radar configuration. Running a TruckMaker simulation then produces radar data in the form of one pointcloud per radar, containing a position, Doppler velocity, timestamp and relative power associated with each point. The pointcloud density is highly dependent on both the radar configuration and the signal processing and filtering.

The number of points is not constant over time, and varies between radars, but the estimators need full measurement vectors $\mathbf{y}_k^{\text{range}}$ and $\mathbf{y}_k^{\text{Doppler}}$. This is a synchronization problem, which can be solved in multiple ways, e.g.:

1. Only estimating the state at time instances where all radars have a measurement available. If t_j is the set of time instances for the measurements from radar j, then the measurement vector $\mathbf{y}(t)$ and consequently the estimate $\hat{\mathbf{x}}(t)$ is available at the intersection $t = t_1 \cap t_2 \cap \dots$.
2. Interpolating any missing measurements from the union $t = t_1 \cup t_2 \cup \dots$.

Interpolating fills out gaps in-between existing samples, which is fine for when the complete measurement sequence $\mathbf{y}_{1:K}$ is available, such as for offline simulations. But for a practical application, interpolation would mean that if one radar cannot provide a measurement at $t = 2$, the sensor system needs to wait until $t = 3$ to provide the measurement vector for $t = 2$ which introduces lag. Whether this is a viable solution depends on the sampling frequency for the individual radars, the availability of measurements, the number of radars and the required full measurement vector frequency.

A better solution would be extrapolation, where only previous samples are used to fill a missing sample. This is done using e.g. zero- or first-order hold. The order refers to the derivative order of the Taylor series expansion. For a zero-order hold, the previous sample's value is simply held until the next available sample. For a first-order hold, the new sample is generated by extending a line through the previous two samples, and so on.

A single radar can also have concurrent measurements, which can be handled in multiple ways. The simplest solution would be to select the point with the highest relative power. Higher power should mean lower attenuation which in turn should mean shorter distance traveled. However, for that assumption to hold, Figure 3.11 should display a clear downward slope. Selecting the highest power out of concurrent data could also result in selecting reflections from clutter between the sensor and the road.

3.7.2 Interfacing with TruckMaker

The state estimation and evaluation was implemented in Matlab and was only performed offline, i.e. after the simulation was completed. For a test, the groundtruth quantities in Table 3.1 were exported to file, as well as each radar pointcloud. The files were then parsed in Matlab. For each radar, first the point with the highest relative power was selected from concurrent points. Outlier rejection was performed

by a z-score threshold: if a sample had a z-score greater than 4 it was replaced by the nearest sample.

$$\left| \frac{y_k - \mu_y}{\sigma_y} \right| \tag{3.105}$$

The measurements from all radars were then synchronized by linear interpolation. Finally, the measurement sequence was synchronized with the groundtruth sequence by linear interpolation. For 6 radars, the resulting data is a $1 \times K$ time vector, a $9 \times K$ groundtruth matrix, a $6 \times K$ range measurement matrix and a $6 \times K$ Doppler measurement matrix. The sample period was determined by the average time step $T = \bar{\Delta}t$. The sample period and measurement sequences were then fed to the estimators, and the resulting estimates were compared to the groundtruth in evaluation.

To effectivize investigating estimate sensitivity, the TruckMaker test execution was automated in Matlab. A radar configuration class was implemented, with functions to generate the corresponding measurement models. Additional functions were written to convert and read/write radar configurations to/from the TruckMaker vehicle files. That way, tests could be repeated while e.g. tweaking elevation angles programmatically. Roads and maneuvers were designed in the IPG Scenario Editor GUI. From Matlab, all simulations were then run through the TruckMaker CLI and evaluation plots and tables were generated.

3.7.3 TruckMaker tests

The TruckMaker scenarios were designed to investigate the estimate's sensitivity to the different sources of error introduced in 3.5. As much as possible, the scenarios try to isolate a single behavior each. The results were then used as feedback to the design process, where the measurement models, radar configuration and other parameters were iteratively improved. The final test evaluations are presented in 4.3.

Road	Maneuver	Additional
Straight	Stationary	
Straight	Accelerate	
Straight	Brake	Soft vs hard
Straight	Constant speed	30 km/h vs 50 km/h vs 80 km/h
Straight	Constant speed	Flat vs bank vs slope vs both
Eight-shape	Constant speed	Rigid vs flexible truck body
Eight-shape	Constant speed	Asphalt vs snow vs custom road surface
Low friction curve	Side-slip	

Table 3.3: TruckMaker test scenarios

The road surface test compares different permittivity (p) and scattering (s) properties of the road material. Asphalt has low p and high s, and snow has low p and low s. A third, custom material is therefore included, which has high p and high s.

The low friction test is a scenario where estimates based on wheel speed encoders would fail due to wheel slip. It also captures higher yaw rate, which in normal driving is quite low.

3.7.4 Validate TruckMaker Doppler measurements

The Doppler measurements generated from TruckMaker need to be validated. It is important to make sure they match real measurements. This ensures that the simulation results are reliable, and that accurate conclusions are drawn from them.

The RT3000 groundtruth data contains longitudinal velocity v_x and steering wheel angle ψ from real tests performed on an open ice-track. These signals were exported as input to TruckMaker, that run on an open road model with appropriate friction coefficient μ. The simulated Doppler measurements could then be compared to the measurements from the real radar sensors.

4

Results

In this chapter the results from the developed methods are presented. First the radar configuration used is shown. Secondly, Matlab simulation results are presented, and then the TruckMaker results. The results are mainly displayed in tables, but some more interesting scenarios are shown in figures.

4.1 Radar configuration

The proposed radar configuration used in the result chapter is illustrated in Figure 4.1 and its parameters shown in Table 4.1.

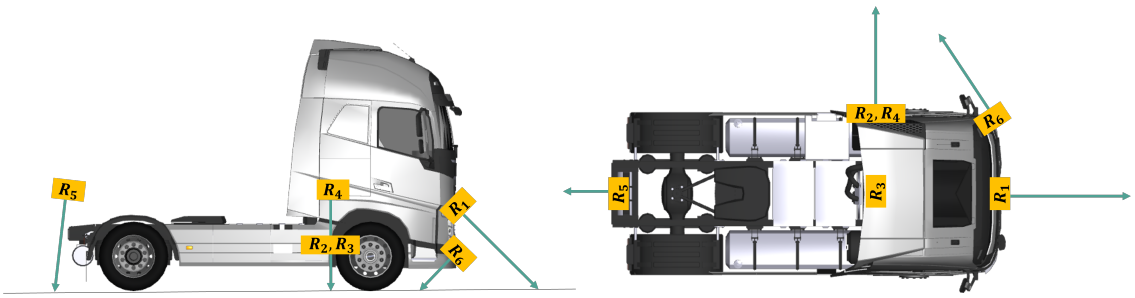

Figure 4.1: Illustration of radar configuration. Note that $\mathbf{R_2}$ and $\mathbf{R_4}$ have the same x- y-position.

Radar	x [m]	y [m]	z [m]	elevation [deg]	azimuth [deg]
$\mathbf{R_1}$	6	0	0.5	30	0
$\mathbf{R_2}$	3.9	1.3	0.5	30	90
$\mathbf{R_3}$	3.9	0	0.5	90	0
$\mathbf{R_4}$	3.9	1.3	1.5	68.962	90
$\mathbf{R_5}$	0	0	1.5	82.694	180
$\mathbf{R_6}$	6	1.3	0.5	30	121.76

Table 4.1: Radar parameters of proposed configuration.

The resulting Doppler - ego-velocity C matrix is:

$$C = \begin{bmatrix} -0.866 & 0 & 0.5 & 0 & -0.617 & 0 \\ 0 & -0.866 & 0.5 & 0.217 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -0.359 & 0.9333 & 1.3928 & 0 & 0 \\ 0.1272 & 0 & 0.9919 & 0 & 3.9319 & 0 \\ 0.4558 & -0.7364 & 0.5 & 0.2818 & -1.2779 & -2.1389 \end{bmatrix} \tag{4.1}$$

4.2 Matlab simulations

The Matlab simulations are results from Equation 3.43 - 3.49. The Doppler measurements were simulated without noise, with normal noise and with high noise. Additionally, Roll-Pitch-Scaling (RPS) was added, to upscale roll rate ω_x and pitch rate ω_y. This is to investigate how perturbing these velocity states affects the 2D velocity estimate. Additionally, uncertainty in the radar configuration was added, which is shown in Table 4.2. Results are shown in the following tables. The radar configuration used is described in Section 4.1 and groundtruth used was from a groundtruth log for a normal driving scenario.

		No noise	High noise	Uncertainty & Normal noise
v_x [m/s]	Mean	-0.0003	**0.0001**	**−0.56277**
	RMSE	0.0353	**0.0593**	**0.66795**
v_y [m/s]	Mean	-0.0000	**−0.0018**	**0.58114**
	RMSE	0.0861	**0.0993**	**0.67947**
ω_z [deg/s]	Mean	-0.0000	**0.0786**	**−31.5299**
	RMSE	1.0353	**1.8985**	**37.2526**

Table 4.2: 3 radars, 2D model, 2D velocity

If an uncertainty regarding the mounting is present, the velocity estimation will be greatly decreased, which can be seen in Table 4.2. The column with uncertainty and normal noise shows both a significant bias and a large RMSE. It is clear that the reason is the uncertainty added since the middle column has higher noise but no uncertainty and the mean and RMSE are still low.

		No noise	High noise	RPS = 10 Normal noise
v_x [m/s]	Mean	0.0231	0.0227	0.0239
	RMSE	0.0531	0.0693	0.0898
v_y [m/s]	Mean	0.0237	0.0247	0.0233
	RMSE	0.0725	0.0939	0.0825
ω_z [deg/s]	Mean	0.3753	0.3291	0.3896
	RMSE	0.9581	1.8854	6.1410

Table 4.3: 3 radars, 2D model, 3D velocity

		No noise	High noise
v_x [m/s]	Mean	-0.0003	**0.0001**
	RMSE	0.0353	**0.0567**
v_y [m/s]	Mean	0.0000	**−0.0007**
	RMSE	0.0861	**0.0970**
ω_z [deg/s]	Mean	0.0000	**0.0155**
	RMSE	1.0353	**1.8405**

Table 4.4: 6 radars, 2D model, 2D velocity

Having redundancy, i.e. more radars than states, does not show significant improvements. This can be seen by comparing Table 4.2 and Table 4.4 respective columns for High noise, where the only difference is the numbers of radars used. The reason for this can be the radar configuration used is derived by dedicating one radar per state. Hence, if the radars used to estimate v_z, ω_x, ω_y are the redundant ones, they will not deliver any useful information regarding v_x, v_y and ω_z.

		No noise	High noise	RPS = 10 Normal noise
v_x [m/s]	Mean	0.0162	0.0172	0.0169
	RMSE	0.0394	0.0686	0.1937
v_y [m/s]	Mean	0.0360	0.0357	0.0353
	RMSE	0.0852	0.0851	0.1485
ω_z [deg/s]	Mean	0.0449	0.0729	0.0690
	RMSE	1.0011	1.4786	8.6339

Table 4.5: 6 radars, 2D model, 3D velocity

		No noise	High noise
v_x [m/s]	Mean	-0.0007	-0.0019
	RMSE	0.0435	0.0682
v_y [m/s]	Mean	-0.0001	0.0016
	RMSE	0.0706	0.1017
ω_z [deg/s]	Mean	-0.0000	-0.0527
	RMSE	0.8717	2.0982

Table 4.6: 6 radars, 3D model, 2D velocity

		No noise	High noise	RPS = 10 Normal noise
v_x [m/s]	Mean	-0.0011	0.0001	-0.0006
	RMSE	0.0556	0.0641	0.0405
v_y [m/s]	Mean	-0.0001	0.0006	0.0001
	RMSE	0.0549	0.0839	0.0877
ω_z [deg/s]	Mean	-0.0001	-0.0153	**−0.0147**
	RMSE	0.6760	1.7813	**1.1893**

Table 4.7: 6 radars, 3D model, 3D velocity

Observing the tables with Roll-Pitch-Scaling present, i.e. Table 4.3, 4.5 and 4.7 it mainly affects ω_z. However, this is not true for Table 4.7 which show less error for ω_z even though the roll φ_x and pitch φ_y is up-scaled. This is due to that Table 4.7 depicts when it was simulated and estimated using a full 3D state velocity model with the full state vector $\mathbf{x}_{velocity}$. In other words, it shows that φ_x and φ_y will affect the 2D estimate quite little and only in very extreme, and arguably unrealistic, scenarios.

4.3 TruckMaker

The ego-motion estimate evaluation from the TruckMaker test set, 3.3, is compiled into Tables 4.8 and 4.9. The ”Asphalt” test, where the truck drives in an 8-shape with constant speed is selected and plotted in Figs. 4.2 - 4.5, as this test summarizes the performance quite well. As the vertical axes of the plots are not to the same scale, the plots effectively show the normalized error of the estimates.

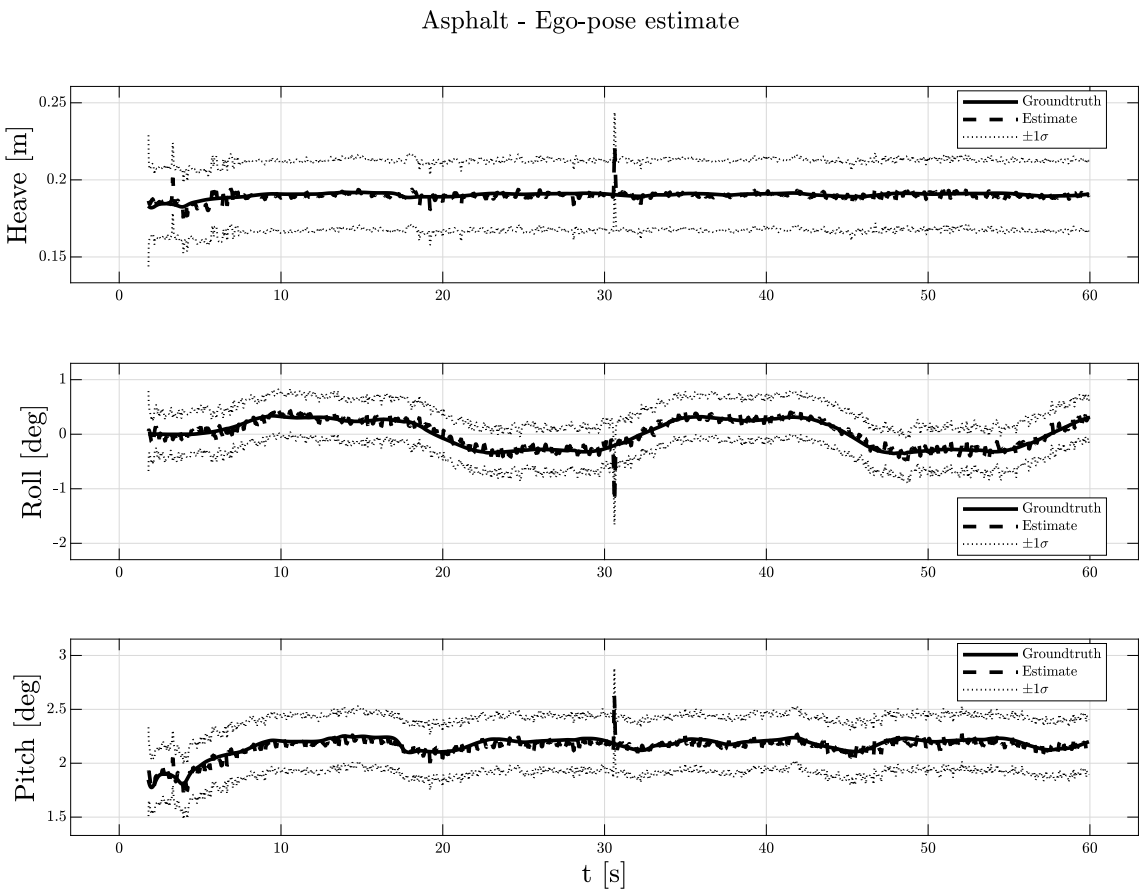

Figure 4.2: Ego-pose estimate and groundtruth, 8-shaped road

The ego-pose estimate shows low bias and overall error, on the order of centimeters and hundredths of a degree. The scenario is quite idealized: the truck body is rigid and there are no uncertainties such as an uneven load; the maneuver is low speed, simple and exact; the road surface is flat and even; no clutter is present.

The estimate is a Gaussian distribution, plotted by its mean and ± one standard deviation σ. As the groundtruth is well within 1σ at all times, it might seem natural to lower the measurement noise covariance in the EKF. That would however decrease the performance for other scenarios where the measurement quality is worse. In other words, higher confidence in the measurements would decrease the robustness of the estimate.

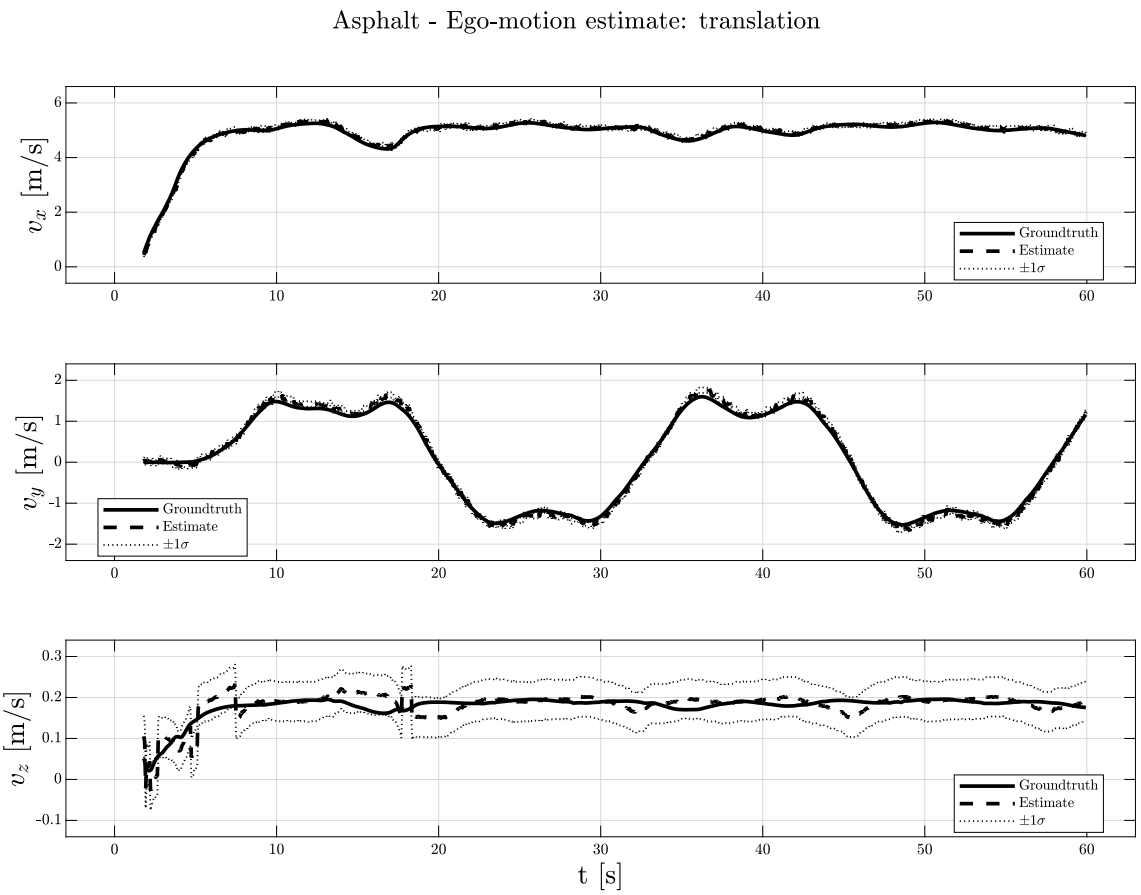

Figure 4.3: Translational ego-velocity estimate and groundtruth, 8-shaped road

The translational ego-velocity estimates show decent performance on the order of cm/s. The heave velocity estimate $\hat{v_z}$ has the lowest RMSE, but comparing mean normalized RMSE values, $\hat{v_x}$ is better:

$$\text{NRMSE}(\hat{v_x}) = \frac{\text{RMSE}(\hat{v_x})}{\bar{v_x}} = 0.0137 \tag{4.2}$$

$$\text{NRMSE}(\hat{v_y}) = \frac{\text{RMSE}(\hat{v_y})}{\bar{v_y}} = 1.9376 \tag{4.3}$$

$$\text{NRMSE}(\hat{v_z}) = \frac{\text{RMSE}(\hat{v_z})}{\bar{v_z}} = 0.1177 \tag{4.4}$$

The heave velocity estimate is interesting. Recall that according to the proposed measurement model it is the only state that can be isolated in the measurements from one radar. This is the motivation for putting the third radar vertically at the center of rotation of the vehicle, which in theory should give

$$y_3^{\text{Doppler}} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}_{\text{velocity}} + w = v_z + w, \quad w \sim \mathcal{N}(0, \sigma_y^2) \tag{4.5}$$

Comparing the groundtruth and the Doppler measurements in Figure 4.4 shows that it is not the case.

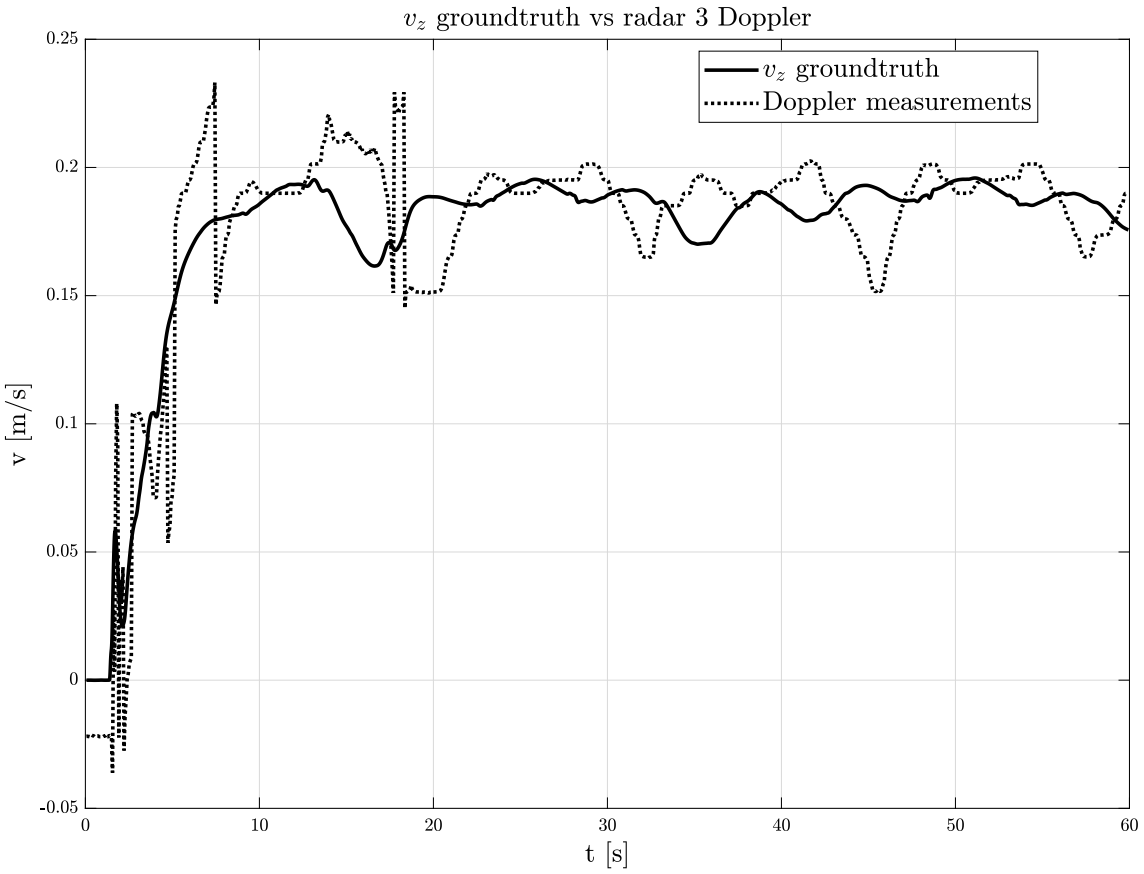

Figure 4.4: Doppler measurements from centered, vertical radar vs v_z groundtruth

The plots suggest that the discrepancy is not exclusively due to measurement noise, as it is continuous. The CoR in the measurement model could be off, which would give the false expectation $y_3^{\text{Doppler}} = v_z$. The measurements could not be recreated by changing the CoR however. Another potential explanation is that the measurement model could be too crude, in that it is missing some vital behavior. More likely though, it is a problem in the signal processing. This is supported by the fact that the fluctuation within 0.05 m/s is also seen in Figure 3.12.

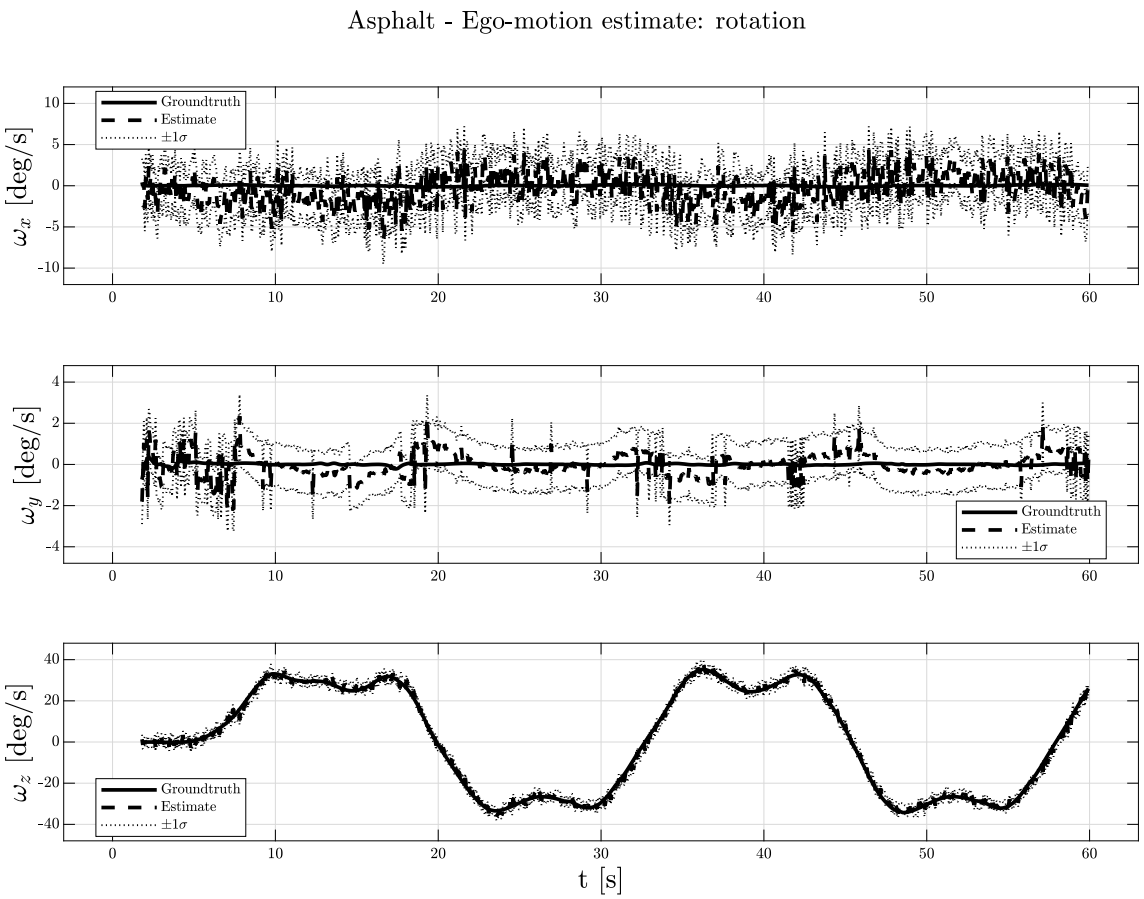

Figure 4.5: Angular ego-velocity estimate and groundtruth, 8-shaped road

The angular ego-velocities are generally harder to estimate which shows, with up to 2 deg/s error. The magnitude of those states are generally much smaller than the translations (note the upscaling from rad/s to deg/s in the results). Pitch rate ω_y has the lowest RMSE, but for the constant, quite slow 20 km/h of the scenario, very little pitch rate is expected. Instead, comparing the NRMSE as before shows that both the roll- and pitch rate estimates are quite abysmal:

$$\text{NRMSE}(\hat{\omega_x}) = 401.201 \tag{4.6}$$
$$\text{NRMSE}(\hat{\omega_y}) = 871.601 \tag{4.7}$$
$$\text{NRMSE}(\hat{\omega_z}) = 1.5167 \tag{4.8}$$

The following observations are categorized by state. The ego-pose estimate is quite robust as it keeps about the same performance for all scenarios. The pose model assumes a rigid vehicle body, so the test using a flexible one was expected to greatly reduce the performance. The effect is shown to be negligible, but it should be noted that minimal flex is expected at such low speeds. More aggressive maneuvers would be better suited for that investigation. The greatest impact came from bank and/or slope of the road. It is worth pointing out that these scenarios did not have constant bank or slope but rather sinusoidal bank/slope profiles. It could be the case that the ego-pose estimate is sensitive to change in bank/slope rather than to bank/slope.

		Stationary	Accelerating	Braking soft	Braking hard	Constant 30 km/h	Constant 50 km/h	Constant 80 km/h
z [m]	Mean	-0.0004	0.0001	0.0013	-0.0021	-0.0015	0.0024	0.0020
	RMSE	0.0050	0.0050	0.0062	0.0128	0.0046	0.0047	0.0050
φ_x [deg]	Mean	0.0507	0.0139	0.0092	0.0129	0.0331	-0.0390	-0.0068
	RMSE	0.0669	0.0629	0.0575	0.0589	0.0640	0.0664	0.0676
φ_y [deg]	Mean	0.0006	-0.0004	0.0082	-0.0122	-0.0108	0.0114	0.0120
	RMSE	0.0602	0.0638	0.0777	0.1800	0.0512	0.0456	0.0514
v_x [m/s]	Mean	**−0.0244**	-0.0912	0.1544	0.4362	-0.0558	-0.0036	-0.0092
	RMSE	**0.0298**	**0.1590**	**0.1723**	**0.5390**	**0.0683**	**0.0570**	**0.0558**
v_y [m/s]	Mean	**−0.0317**	0.0045	-0.0131	0.0119	-0.0673	-0.0177	0.0108
	RMSE	0.0328	0.0385	0.0419	0.0364	0.0696	0.0311	0.0375
v_z [m/s]	Mean	**−0.0218**	-0.0091	0.0034	0.0221	-0.0548	-0.0035	-0.0036
	RMSE	0.0218	0.0407	0.0303	0.0482	0.0590	0.0096	0.0398
ω_x [deg/s]	Mean	**0.3696**	-0.0559	-0.2669	-0.0311	0.9111	-0.4025	0.0825
	RMSE	0.7502	1.9996	1.6193	1.7803	1.3582	1.2951	2.1458
ω_y [deg/s]	Mean	**0.5329**	0.0963	0.2465	-0.3789	0.9110	0.0309	0.5548
	RMSE	0.5351	1.3310	1.0405	1.0571	1.1669	1.2013	1.4044
ω_z [deg/s]	Mean	**−0.3303**	0.0618	0.0992	0.2376	-0.3267	0.7176	0.0791
	RMSE	0.6809	1.5420	1.4305	1.1598	0.9516	1.4835	1.5139

Table 4.8: Ego-motion estimate performance for the TruckMaker test set, 1/2

		Flat	Bank	Slope	Both	Rigid	Flexible	**Asphalt**	Snow	High p high s	Side-slip
z [m]	Mean	0.0032	-0.0024	-0.0012	-0.0019	-0.0006	-0.0001	**−0.0005**	-0.0010	-0.0008	-0.0011
	RMSE	**0.0042**	**0.0383**	0.0080	0.0097	**0.0029**	**0.0035**	**0.0030**	0.0030	0.0029	0.0045
φ_x [deg]	Mean	-0.0651	0.1315	0.0100	0.0237	0.0106	0.0042	**0.0131**	0.0172	0.0201	0.0073
	RMSE	**0.0808**	**0.6451**	0.2522	0.3726	**0.0598**	**0.0694**	**0.0679**	0.0610	0.0641	0.0603
φ_y [deg]	Mean	0.0194	-0.0085	-0.0116	-0.0120	-0.0169	-0.0080	**−0.0145**	-0.0236	-0.0195	-0.0110
	RMSE	**0.0396**	**0.1443**	0.1257	0.1248	**0.0412**	**0.0417**	**0.0415**	0.0467	0.0431	0.0690
v_x [m/s]	Mean	-0.0212	-0.0058	0.0113	0.0101	0.0279	0.0501	**0.0283**	0.0270	0.0299	-0.0037
	RMSE	0.0574	0.0465	0.0506	0.0500	0.0725	0.0869	**0.0648**	0.0690	0.0654	0.1771
v_y [m/s]	Mean	0.0416	-0.0065	0.0049	0.0011	0.0146	0.0290	**0.0131**	0.0146	0.0117	0.0267
	RMSE	0.0458	0.0383	0.0501	0.0454	0.0888	0.0827	**0.0876**	0.0904	0.0902	**0.1649**
v_z [m/s]	Mean	-0.0127	-0.0244	0.0014	-0.0002	0.0024	0.0319	**0.0024**	0.0029	0.0024	-0.0029
	RMSE	0.0189	0.0355	0.0311	0.0315	0.0205	0.0530	**0.0205**	0.0211	0.0205	0.0389
ω_x [deg/s]	Mean	0.6006	0.5309	-0.2387	-0.3173	-0.2971	-1.1621	**−0.3611**	-0.3944	-0.3932	-0.2288
	RMSE	1.6314	1.8380	2.0121	2.0962	**1.9033**	**2.1886**	**1.8362**	1.8602	1.7977	2.1622
ω_y [deg/s]	Mean	1.1595	0.3234	-0.1747	-0.1895	-0.0339	-0.5109	**−0.0228**	-0.0316	-0.0105	-0.3997
	RMSE	1.5528	1.3030	1.3840	1.3551	**0.5768**	**1.0679**	**0.5325**	0.5350	0.5232	1.3748
ω_z [deg/s]	Mean	-1.1306	0.0758	0.3135	0.4264	0.1165	0.7235	**0.0892**	-0.0771	-0.0142	0.3709
	RMSE	1.6773	1.6056	1.5618	1.6867	**1.5084**	**1.7752**	**1.4159**	1.4690	1.4614	1.7803

Table 4.9: Ego-motion estimate performance for the TruckMaker test set, 2/2

From the whole of Table 4.8 it can be seen that rapid change has the biggest impact on the v_x estimate. This relates to the distribution of v_x, that has a large variance when considering different scenarios. Lateral and heave velocity v_y, v_z estimates appear robust as well, but those states are also generally quite small. It can be noted that RMSE(v_y) grows for the side-slip case, which has the highest lateral velocity.

Especially the angular ego-velocity estimate has a high bias in the stationary test. This reveals that the Doppler measurement noise is not zero mean, contrary to assumed.

$$\bar{\mathbf{e}} = \sum(\hat{\mathbf{x}} - \mathbf{0}) = \mathrm{E}[\hat{\mathbf{x}}] \tag{4.9}$$
$$= \mathrm{E}[C^{-1}\mathbf{y}_{\mathrm{Doppler}}] \tag{4.10}$$
$$= \mathrm{E}[C^{-1}(C\mathbf{0} + \mathbf{w})] \tag{4.11}$$
$$= C^{-1}\mathrm{E}[\mathbf{w}] \tag{4.12}$$
$$\bar{\mathbf{e}} \neq \mathbf{0} \iff \mathrm{E}[\mathbf{w}] \neq \mathbf{0} \tag{4.13}$$

The angular ego-velocity estimate also appears to be the most sensitive of the states to truck body flex.

5
Conclusion

This thesis has investigated whether it is possible to estimate the nine vehicle ego-motion states using radar range and Doppler measurements. The conclusion is that estimating ego-pose and translational ego-velocity is possible. Estimating angular ego-velocity could be possible for specific applications, as it shows poor performance in normal driving scenarios. For estimating ego-pose, three radars are needed. For ego-velocity, three radars can be used to estimate the 2D states, or six for the full ego-velocity vector. Using both range and Doppler velocity measurements from each radar, estimating the full ego-motion requires a total of six radars. These radars need to be mounted so that the points of their reflections span the road plane, as well as capture the projection of each ego-velocity state. The ego-pose estimate is reliable as long as the road is planar and the radars do not measure obstacles such as railings or other vehicles. The translational ego-velocity estimate is more accurate than the angular ego-velocity estimate for all cases tested. The ego-velocity estimate is highly sensitive to the choice of radar configuration, and the optimal radar configuration depends on the distribution of the ego-velocity state, i.e. what road, vehicle and maneuvers the estimation is used in. A combined estimate accuracy measure, defined by the maximum RMSE over all tests for each state, is presented in Table 5.1.

State	max RMSE	
z	0.0383	m
φ_x	0.6451	deg
φ_y	0.1800	deg
v_x	0.5390	m/s
v_y	0.1649	m/s
v_z	0.0590	m/s
ω_x	2.1886	deg/s
ω_y	1.5528	deg/s
ω_z	3.6338	deg/s

Table 5.1: Combined ego-motion estimate accuracy.

The validity of the conclusions rely on the simulated radars being fairly realistic. The simulated radar used in this thesis was not ideal, and due to the computational costs of the simulation, the signal processing was actually on the lower end of real radars. Still, it is possible that it is not practically feasible. The Doppler measurements were compared to real radars, but the verification results could not be published in this

thesis. Additionally, there were no real range measurements available to compare with.

5.1 Real experiments

Several real radar experiments were available to process which, in addition to the TruckMaker results, also proved that it is possible to estimate the 2D ego-velocities. The obtained results show low mean and RMSE for vx, v_y and ω_z. Research question 1 was formulated to contain all nine states. However, due to the radar configuration used during these tests, mainly the 2D states were possible to estimate. Extending the estimation to 3D states reveals that the radar configuration used in that particular test is not well suited.

5.2 Estimation method

The estimation method mainly used was a Kalman filter for the ego-velocity estimation and an Extended Kalman filter for the ego-pose estimation. The KF was compared to (pseudo) inversion and performed better overall, since it can adapt to noise. The KF/EKF were used with constant process models which proved to be enough to yield good estimates. The process noise covariance was tuned from real data and TruckMaker data. The tuning of the process noise was the main design parameter and could most likely be improved to get better estimation results.

5.3 Radar configuration

The main source of error was found to be the radar configuration. The measurement models and estimation methods only work if the configuration is known. Having large discrepancies would not work which is seen by the simulated result in Table 4.2. If used in a real vehicle this could be a big problem, e.g. if the radar mounting is unexpectedly altered, giving rise to a significant mismatch between the real configuration and the modeled configuration.

The proposed configuration does not take into account the mounting of a trailer or any other mechanical restrictions. Such limitations would probably force the configuration to change and thus the performance accuracy. In that sense, the configuration was only tested in a very specific and isolated environment.

The redundancy in radar configuration did not show significant improvements is the estimate in the Matlab simulations. However, redundancy could still be important since radars could fail while in operation. The proposed configuration does not take this into account. If the radar measuring v_x would break, no other radar would be able to properly account for that loss in information. This is a weakness of the

configuration.

5.4 Improvements and future work

Due to limited time, various investigations has been left out of scope. Some suggestions on improvements or future work are discussed here.

5.4.1 Inertial measurement unit

This thesis has only focused on the use of radar sensors. However, the sensor suite could be extended by incorporating an Inertial Measuring Unit (IMU). It would be an easy and inexpensive way to improve the estimate of $\boldsymbol{\omega}$.

5.4.2 Measurement outlier rejection

The ego-pose estimation relies on the assumption that the road is planar. Linear regression or RANSAC could be applied to find the road plane in the radar point-cloud, and remove any outliers. This could improve the range measurements, but more importantly avoid Doppler measurements from being collected from obstacles such as other vehicles or pedestrians.

5.4.3 Scenarios

The flexible vs rigid vehicle body scenario should have been tested with more aggressive maneuvers, to generate higher flex. The ego-pose estimate sensitivity to obstacles should be tested by scenarios containing e.g. railing, potholes and other vehicles. The bank and slope tests should be split into constant bank/slope and change in bank/slope. Scenarios that are both common and critical, such as lane changes, should be added.

Bibliography

[1] H. H. Meinel. "Evolving automotive radar — From the very beginnings into the future". In: *The 8th European Conference on Antennas and Propagation (EuCAP 2014)* (2014).

[2] T. M. Hyltin et al. "Vehicular Radar Speedometer". In: *SAE Transactions* (1973).

[3] W. Kleinhempel. "Automobile doppler speedometer". In: *Proceedings of VNIS '93 - Vehicle Navigation and Information Systems Conference* (1993), pp. 509–512.

[4] W. Menzel A. Hantsch. "A 76GHz Folded Reflector Antenna for True Ground Speed Measurement". In: *Proceedings of the German Microwave Conference* (2006).

[5] D. Kellner et al. "Instantaneous ego-motion estimation using Doppler radar". In: *IEEE International Conference on Intelligent Transportation Systems (ITSC)* (2013).

[6] D. Kellner et al. "Instantaneous ego-motion estimation using multiple Doppler radars". In: *IEEE International Conference on Robotics and Automation (ICRA)* (2014).

[7] C. Kollberg. *Radar-Based Two-Dimensional Ego-Motion Estimation for Heavy Duty Vehicles*. 2016.

[8] M. Sigonius. "Speed and Yaw Rate Estimation in Autonomous Vehicles Using Doppler Radar Measurements". In: (2018).

[9] S. H. Cen and P. Newman. "Radar-only ego-motion estimation in difficult settings via graph matching". In: *IEEE International Conference on Robotics and Automation (ICRA)* (2019).

[10] Simo Sarkka. *Bayesisan Filtering and Smooting*. Cambridge University Press., 2013.

[11] W. Burgard S. Thrun and D. Fox. *Probabilistic Robotics*. MIT Press, 2006.

[12] Qasim Chaudhari. *FMCW Radar*. URL: `https://wirelesspi.com/fmcw-radar-part-1-ranging`.

[13] IPG Automotive. *CarMaker Reference Manual 13.0*. IPG Automotive.

[14] David K. Barton. *Radar System Analysis and Modeling*. Artech House, 2004. ISBN: 9781580536813.

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY