Dataset to "Ultra-wideband angle of arrival estimation based on angle-dependent antenna transfer functions"

Anton Ledergerber (antonl@ethz.ch)

Institute for Dynamic Systems and Control (IDSC), ETH Zurich, Zurich, Switzerland
compiled on September 27, 2019

1 Introduction

This document describes the data presented in

> A. Ledergerber, R. D'Andrea, "Ultra-wideband angle of arrival estimation based on angle-dependent antenna transfer function, " submitted to *Sensors*, 2019,

which can be cited if the data is used in another publication.

The data of one experiment consists of:

- A Protobuf log containing the raw onboard log, named "logData.dat"
- A Protobuf log containing the raw offboard log, named "logOffboardData.dat"
- A numpy log of the processed onboard and offboard log, named "processedLog.npz"
- A ini file of some configuration parameters, named "setupConfig.ini"
- A picture of the anchor and obstacle setup
- If captured at the time, a video of the first few minutes of an experiment

In the following each of these logs is described in detail. The Protobuf message definition for the Protobuf logs is given in the file "logCIR.proto". Protobuf version 3.5.1 was used for logging, but newer versions can open the log files as well.

For the impatient ones who simply want to get the measured CIR, the corresponding measured range, and the corresponding ground truth angle of arrival, a python code snipped is given in "example.py".

2 Onboard log

The onboard log contains the UWB measurements and the odometry measurements. Both are timestamped with the same clock of the host microcontroller.

2.1 UWB measurement

A UWB measurement is visualized in Figure 1 and is composed of the following data fields (all contained in the Protobuf message "UWBRangingExtTopic"):

- **uint64 timestamp** The time recorded on the host microcontroller when it received the UWB measurement

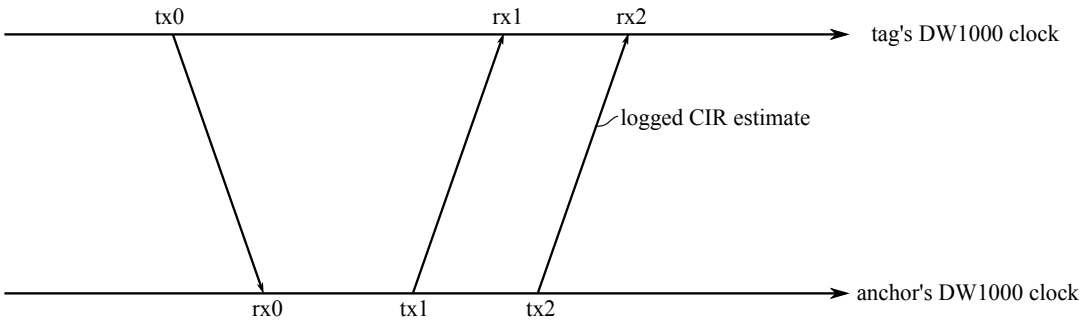

Figure 1: The one-way repeated ranging protocol as presented in [1]

- **uint64 tx0** The time in the tag's DW1000 clock when the first signal of the one-way repeated ranging (OWRR) as presented in [1] was transmitted
- **uint64 rx0** The time in the anchor's DW1000 clock when the first signal of the OWRR was received
- **uint64 tx1** The time in the anchor's DW1000 clock when the second signal of the OWRR was transmitted
- **uint64 rx1** The time in the tag's DW1000 clock when the second signal of the OWRR was received
- **uint64 tx2** The time in the anchor's DW1000 clock when the third signal of the OWRR was transmitted
- **uint64 rx2** Time in the tag's DW1000 clock when the third signal of the OWRR was received
- **float absolute_time** Unused/Deprecated
- **float fp_index** The first path index in the accumulator memory
- **uint32 fp_index_raw** The first path index as read from the register (FP_INDEX)
- **uint32 fp_ampl1** The amplitude of the first path as as read from the register (FP_AMPL1)
- **uint32 lde_ppindx** The leading edge detection peak path index as read from the register (LDE_PPAMPL)
- **uint32 lde_ppampl** The leading edge detection peak path amplitude as read from the register (LDE_PPINDEX)
- **uint32 cir** The channel impulse response power as read from the register (CIR_PWR)
- **uint32 rxpacc** The preample accumulation count as read from the register (RXPACC)
- **uint32 partner** The ID of the transmitting module
- **bytes p_acc_mem** The accumulator data memory as read from the register (ACC_MEM), starting 4 samples before the first path index and ending 25 samples after the first path index
- **uint32 rcphase** The receive carrier phase adjustment as read from the register (RCPHASE)

All mentioned DW1000 register values are read for the last message exchanged in the one-way repeated ranging protocol [1]. For a more detailed description of the registers, the reader is referred to [2].

2.2 Odometry measurement

A odometery measurement is composed of the following data fields, (all contained in the Protobuf message "IRobot-DistanceAngle"):

- **uint64 timestamp** The time recorded on the host microcontroller when it received the odometry measurement
- **int32 distance** The distance that the Roomba robot has traveled in millimeters since the distance was last read
- **int32 angle** The angle in degrees that the Roomba robot has turned since the angle was last read

For a more detailed description the reader is refereed to [3].

3 Offboard log

The offboard log contains motion capture measurements. Such a measurement is composed of the following data fields (all contained in the Protobuf message "MotionCapturePose2"):

- **uint64 timestamp** The time recorded on the network computer when a motion capture package was received
- **int32 ID**. The motion capture object ID.
- **float p_x** The x-coordinate of the motion capture object's position
- **float p_y** The y-coordinate of the motion capture object's position
- **float p_z** The z-coordinate of the motion capture object's position
- **float q_w** The real part of motion capture object's orientation quaternion
- **float q_x** The x-coordinate of motion capture object's orientation quaternion
- **float q_y** The y-coordinate of motion capture object's orientation quaternion
- **float q_z** The z-coordinate of motion capture object's orientation quaternion

Note that the motion capture object's position does not correspond to the antenna position. The antenna position can be computed via the parameters given in the "Parameters.py" file.

4 Processed log

The processed log contains some processed data from the onboard and offboard log files.

- **realCIR** The real part (in-phase component) of the recorded CIRs
- **imagCIR** The imaginary part (quadrature component) of the recorded CIRs
- **offsetCIR** The times between the fourth recorded samples of the CIRs and the estimated first path locations in nanoseconds
- **t_uwb** The time the UWB measurements were taken in microseconds
- **ID_uwb** The IDs of the UWB modules/anchors with which the measurements were obtained
- **range_uwb** The ranges measured between the tag and the UWB anchor using the timestamps discussed in 2 and a constant antenna delay
- **pos_m_mc** The positions of the tag antenna as calculated from motion capture system data

- **heading_m_mc** The orientations of the tag antenna as calculated from motion capture system data
- **pos_s_mc** The positions of the anchor antennas as calculated from motion caputre system data
- **heading_s_mc** The orientations of the anchor antennas as calculated from motion capture system data
- **yaw_m_mc** The angles of arrival as calculated from motion capture system data
- **yaw_s_mc** The angles of departure as calculated from motion capture system data
- **pos_m_e** The positions of the tag antenna as estimated by the particle filter employing a Roomba process model and range measurement updates
- **heading_m_e** The orientations of the tag antenna as estimated by the particle filter employing a Roomba process model and range measurement updates
- **pos_s_e** Unused
- **heading_s_e** Unused
- **yaw_m_e** The angles of arrival as recorded by the particle filter employing a Roomba process model and range measurement updates
- **yaw_s_e** Unused
- **t_iRobot** The times the odometry measurements were taken in microseconds
- **diffDistance_iRobot** The distances the Roomba robots travelled in millimeters since the last time the odometry measurement were taken
- **diffAngle_iRobot** The angles in degrees that the Roomba robot turned since the last time the odometry measurements were taken

5 Setup configuration

The setup configuration file is composed of the following fields:

- **accMemLength** Number of logged CIR Samples
- **samplesBeforeFPIndex** Number of samples before the first path location in the CIR
- **antenna_s** Anchor antenna configuration
- **antenna_m** Tag antenna configuration
- **mounted_CP** Whether or not carbon plates were mounted in the antenna's vicinity
- **ID_mc_m** Tag's motion capture object ID
- **ID_mc_s** Anchors' motion capture object IDs
- **ID_uwb_s** Anchors' UWB ID
- **pos_antenna_s** Anchors' position as measured by the motion capture system
- **x_init_m** Tag's initial x coordinate as measured by the motion capture system
- **y_init_m** Tag's initial y coordinate as measured by the motion capture system
- **z_m** Tag's average z coordinate as measured by the motion capture system

- **yaw_init_m** Tag's initial orientation as measured by the motion capture system
- **loggedUWBTopic** The type of UWB message that is logged

References

[1] M. W. Mueller, M. Hamer, and R. D'Andrea, "Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1730–1736.

[2] *DW1000 USER MANUAL*, Decawave.

[3] *iRobot Create Open Interface*, www.irobot.com.