Introduction

This week I tried a different approach to tackle the problem: in stead of using the parameters exposed over UDP I modified the code of GNSS-SDR iself. The main idea was to retrieve the missing amplitude parameter from the Acquisition block and incorporate it into the Gnss-Synchro object which propagates upto the PVT block. This way all the information needed for assembling the features needed for a classifier are available in the PVT block.
The advantages of this approach is that it guarantees that all features correspond to the same set of measurements, and the classification results can be included in the outputs of the PVT block. //
The modified code is available on GitHub on:

`https://github.com/robix82/GNS-SSDR-mod.git`
or over ssh:
`git@github.com:robix82/GNS-SSDR-mod.git`

Branch `master` contains the original unmodified code, `dev` contains the latest stable version of the modifications, `experimental` contains the work in progress.

1 Feature retrieval

1.1 Find and propagate the amplitude

In order to include the amplitude information into the Gnss_Synchro objects I modified `/src/core/system_parameters/gnss_synchro.h` at lines: 59 and 105 by adding a field for containing the value.
The value is set in `/src/algorithms/acquisition/gnuradio_blocks.pcps_acquisition.cc` at line 687.
By the way I am not completely sure that the assigned value is the correct one; we will need to check this.

If this approach turns out to be useful, an analogous adaptation needs to be applied to all Acquisition Block implementations.

1.2 Assemble the features in the PVT block

For later processing the features I created the class FeatureSet in `/src/algorithms/PVT/gnuradio_blocks` as `feature_set.h`, which holds the values as well as methods for setting the features that need to be computed. Some information comes from the `Gnss_Synchro` objects which arrive as input to the PVT block, other other information comes from the `d_pvt_solver` pointer.

The features are assembled as follows:

Number of valid satellites: Already set by GNSS-SDR.

satellites changed: from Gnss-Synchro.

Signal over noise: from Gnss-Synchro.

Carrier phases: from Gnss-Synchro.

Gap from last pos: from pvt_solver.

Easting from real pos: computed from average over last 10 positions.

Northing from real pos: computed from average over last 10 positions.

Height from real pos: computed from average over last 10 positions.

velocity: from pvt_solver.

acceleration: from pvt_solver.

Time without lock: computed using timer.

1.3 Insert a dummy classifier

A classifier can take as input an instance of the FeatureSet class and output a Classification. I implemented a dummy version and connected it to the PVT block.

1.4 Installation on Raspberry Pi 3 B+

With Raspbian Stretch Lite installed on a 16 GB SD card I cloned the modified source from my Github repo and followed the GNSS-SDR installation guide.

The installation succeeded and the program runs the same way as on the laptop, just a bit slower.