Homodyne Data Analysis

Klaus Zimmermann

December 10, 2015

Contents

1 Introduction

The analysis of homodyne data proceeds in stages.

1. First the data is brought into a standard HDF5 format. This can be done from the current ascii data by means of the `convert_raw_quadratures.py` program.

2. Then the quadratures are standardized by the `standardize_raw_quadratures.py` program.

3. Next the reconstruction of the Wigner functions is performed by the `tomography.py` program.

4. Subsequent analysis takes place. Right now this can be:

- Plotting of the averaged Wigner functions by `plot_reconstructions.py`.
- Fitting of Gaussian functions by `fit_gaussians.py`.

The general philosophy of these programs is to keep all data for one measurement together in one big HDF5 file. This may be either directly as datasets or, if several pieces of data are connected strongly, in a group containing datasets. All groups and datasets have an attribute `git_version` that uniquely identifies the version of the program that was used to create the object. Due to the different nature of different datasets their dimensions vary. There is, however, a common theme: Usually the order of dimension for quadrature data is as follows:

1. Scans
2. Steps, that is pump-probe time delay steps
3. Angles, that is local oscillator steps
4. Pulses.

For phase space data we have:

1. Scans
2. Steps, that is pump-probe time delay steps
3. Q
4. P.

Not all dimensions are present in all datasets. This is explained in detail with the individual datasets below.

The stages of the analysis are implemented in separate programs. Every program can be controlled by command line arguments. Generally, `-h` or `--help` will give a short overview of the available command line arguments. One argument that is present in almost all programs is `-f` or `--force`. Without this switch programs do not destroy present information. With this switch activated, already present analysis will be overwritten. This is useful for development and to check the effect of different parameters, for example a different resolution in the reconstruction of the Wigner functions.

In the following the analysis steps are briefly described.

2 Conversion of Raw Quadratures

Raw quadratures, present in text form, can be converted to the HDF5 format. The result will be a `.h5` file, the inputs for this are the measured quadratures and a set of measured vacuum quadratures. Optionally some info files can be present, but since that information seems to be unreliable, the conversion program only uses it for consistency checks and only outputs warnings if inconsistencies are found.

2.1 Usage

```
convert_raw_quadratures.py [-h] [-f] -v VACUUM [-e ETA] basename
```

2.1.1 Command Line Arguments

- `-h, --help`: Gives a short help
- `-f, --force`: Normally the program will abort if the output file already exists. With `-f` the output file will be overwritten, DELETING earlier imports of the data with all analysis inside the file.
- `-v VACUUM, --vacuum VACUUM`: Specifies the file name of the vacuum data. This is a single step of quadrature data, acquired under vacuum conditions. Necessary for the standardization, hence REQUIRED.
- `-e ETA, --eta ETA`: The detector efficiency. Will be saved as an attribute in the quadrature data and used for the standardization and reconstruction.
- `basename`: Base name of the quadrature data. This is the common prefix of all the text files containing this measurement. Also determines the name of the output file.

2.1.2 Input

The input consists in the text files making up the measurement. The bulk of it is a number of files with names of the form `<basename>-scan-<i>-step-<j>`. Here `i` is the number of the scan and `j` is the number of the step in the scan. Both are expected to be consecutive and to start from 0. Furthermore, all scans should have the same number of steps. One file contains the quadratures for all angles of the scan and step specified in its name. Every line in the file corresponds to one angle; every column to one pulse. The columns are separated by semicolons and given as floating point numbers with comma as decimal separator.

The second mandatory input is the vacuum measurement. It is given as a file of the same format; the name must be specified using the -v argument (see above). For the vacuum measurement, the scan and step information from the file name are ignored.

2.1.3 Output

The output takes the form of a single HDF5 file that is named `basename.h5`. It contains two datasets:

- `raw_quadratures` with dimensions (scan, step, angle, pulse) and
- `vacuum_quadratures` with dimensions (angle, pulse).

3 Standardization of Quadratures

The quadratures as acquired from the experimental setup really are voltages. On top of that they suffer from a number of problems that are consequences of the setup. Some of the problems are unavoidable, others we hope to improve in future iterations of the experiment. To make best use of the data we have right now, we must apply a number of corrections before we enter into the reconstruction itself. Additionally, some analysis is needed to map the steps of the local oscillator to the angles of the quadrature operators. Both tasks are performed by the `standardize_raw_quadratures.py` program. Namely it does the following steps:

1. Correct for intrastep drift. This is a defect that occurs if the moving stage of the local oscillator has not come completely to rest before the measurement is started. In this case the points for even one fixed angle are coming from slightly different angles, effectively broadening the distribution; however, due to the high repetition rate the movement of the stage is still slow compared with the acquired points, so that a polynomial fit can be used to correct for it.

2. Center distributions on a cosine. This step serves two purposes. On the one hand it corrects an imbalance that can shift the voltages from one angle to the next. On the other hand it establishes the relation between the local oscillator steps and the quadrature angles. This is

done by fitting the function

$$V(\phi) = V_0 + A\cos(\omega\phi + \phi_0)$$

to the quadrature data (where V is for voltage). The quadrature distributions are then centered, angle for angle, on this cosine. Furthermore the whole curve is centered around zero. The fit parameters V_0 and A are then discarded, while ω and ϕ_0 are recorded for future analysis.

3. Finally, a vacuum correction is applied. This turns the voltages into quadratures according to (5) of New J. Phys. 16 (2014) 043004.

3.1 Usage

```
standardize_raw_quadratures.py [-h] [-f] [-s SCANS] filename
```

3.1.1 Command Line Arguments

- `-h, --help`: Gives a short help
- `-f, --force`: Normally the program will abort if the pertinent datasets are already present in the HDF5 file. With `-f`, the selected scans will be overwritten.
- `-s SCANS, --scans SCANS`: Which scans to treat. The output (see section Output below) will only be calculated for the selected scans. SCANS can either be "all", or a comma separated list of ranges. Examples:
  - `-s all`
  - `--scans 4`
  - `-s 4-6,9,12-14`
- `filename`: Name of the HDF5 file. Contains both, input and output.

3.1.2 Input

The input is contained in the HDF5 specified by the filename parameter. It consists in the `raw_quadratures` and `vacuum_quadratures` datasets as specified in the Output section of `convert_raw_quadratures.py`.

3.1.3 Output

The output will also be placed in `<filename>.h5`, and comprises the following datasets:

- `angles` for all scans specifies the angle for every local oscillator step. Dimensions are (scan, angle).
- `phi_0` holds the aforementioned fit parameter ϕ_0. Dimensions are (scans, steps).
- `standardized_quadratures` is the main output. It contains the standardized quadratures suitable for the reconstruction process. Dimensions are as the raw quadratures, that is (scan, step, angle, pulse).