

ReadMe

🕒 Created	@February 1, 2022 5:42 PM
🕒 Last Edited Time	@February 1, 2022 6:25 PM
▼ Type	Technical Spec
▼ Status	
👤 Created By	
👤 Last Edited By	
👥 Stakeholders	

Dependencies

Getting Started

Data Folder

Processing

Analysis

samplingFrequencyAnalysis.m
samplingFrequencyAnalysis_perCondition.m
compareNominalParametersToOptimal.m
compareGDFextensions.m

Optimization

defaultOptimOpts.m
estimateAttitude2.m
optimizeAHRSParameters.m
optimizeAHRScrossDataset.m


Dependencies

OpenSim Matlab library required for opening .c3d files

Scripting with Matlab



These instructions assume that you've already installed OpenSim version 4.0 or later (see Installation Guide for more info). As noted above, OpenSim 4.0 requires 64-bit Matlab. Launch MATLAB. If

 <https://simtk-confluence.stanford.edu:8443/display/OpenSim/Scripting+with+Matlab#ScriptingwithMatlab-MatlabSetupSettingupyourMatlabScriptingEnvironment>

Getting Started

1. You need access to the data folder. If you are accessing this repository from the UQ rdm server, this folder should be located at:

```
| \\shares01.rdm.uq.edu.au\DORSAVIVAL-A1680\Ryan\data
```

2. Change your current directory in matlab to the top level of the repository (should have the function SetPaths.m in the current directory).
3. Run the command SetPaths to set the matlab path to use this repository


















```
SetPaths
```

4. You need to point to where the data folder is stored on your computer. This is done in the file `setFileNames.m`. In this file, you would then set the variable `dataDir` to where your data directory is, for example:

```
dataDir = 'C:\Users\ryrid\Documents\Work\painData\validation2021\';
```

Data Folder

The data folder should look something like this

 figuresAndTables	1/02/2022 6:22 PM	File folder	
 filterParameters	1/02/2022 6:22 PM	File folder	
 s1	27/01/2022 4:05 PM	File folder	
 s2	27/01/2022 4:06 PM	File folder	
 s3	27/01/2022 4:06 PM	File folder	
 s4	27/01/2022 4:06 PM	File folder	
 s5	27/01/2022 4:06 PM	File folder	
 s6	27/01/2022 4:07 PM	File folder	
 s7	27/01/2022 4:07 PM	File folder	
 s8	27/01/2022 4:07 PM	File folder	
 s9	27/01/2022 4:07 PM	File folder	
 s10	27/01/2022 4:05 PM	File folder	
 s11	27/01/2022 4:06 PM	File folder	
 s12	27/01/2022 4:06 PM	File folder	
 condition_filenumbers.xlsx	29/06/2021 1:36 PM	Microsoft Excel W...	13 KB
 condition_filenumbers_s1corrected.xlsx	22/12/2021 5:44 PM	Microsoft Excel W...	13 KB
 validation2021_dataset.mat	1/02/2022 6:23 PM	MATLAB Data	760,509 KB

The raw data files in addition to pre-processed data are stored in participant-specific folders s1, s2, ..., s12.

- `validation2021_dataset.mat` has the entire dataset that is used for the analysis
- The results of optimizations to select filter parameters are stored in `filterParameters/`
- Figures and tables generated from analyses are stored in `figuresAndTables/`

Processing

The pre-processing includes syncing between mocap and the sensors, magnetometer calibration, removing gyro offset, computing the baseline magnetometer heading, etc.

For more detailed information of how the data is pre-processed, see the `/processing` folder. To re-process all data, you can run `extractData.m` with your desired options. This file calls nested functions for processing: `processSubject.m` → `processCondition.m` → `processSensor.m`.

Analysis

The folder `analyses` contains the main analyses used for the paper

`samplingFrequencyAnalysis.m`

This file outputs a data table and figures showing how various orientation filtering algorithms perform versus sampling frequency. Optimizations were performed to select optimal filter parameters at each sampling frequency across all conditions.

To re-run the optimization, run the command `runSampleFrequencyOptimizations.m`. This may take a while.

`samplingFrequencyAnalysis_perCondition.m`

The same as `samplingFrequencyAnalysis.m` except an optimization is performed separately for each condition. `runSampleFrequencyOptimizations.m` also runs the optimizations for this file.

`compareNominalParametersToOptimal.m`

This file generates figures and tables showing how using nominal parameter values compare to parameter values that were selected from optimization across all conditions (as in `samplingFrequencyAnalysis.m`).

`compareGDFextensions.m`

This file compares a baseline Gradient Descent Filter (Wilson et. al) to Riddick's adaptations which incrementally show how these adaptations improve performance especially at lower frequencies.

Run the command `runOptimizations_RiddickVariations.m` to re-do the optimizations for this file.

Optimization

Optimization is performed in 2 separate methods

1. MultiStart: used for all filters besides the fast kalman filter. Starting from 8 guesses for initial parameters, run a local fmincon to find the local minima, and then pick the overall global solution. Most GDF filter optimizations are convex so we don't need many starting guess points
2. surrogateopt: used for the fast kalman filter since the cost function is highly non-convex with regards to the variance parameter in the magnetometer. This evaluates at 80 random points and then does 20 rounds of adaptation from the best point

See the `optimization/` folder for functions used to optimize.

`defaultOptimOpts.m`

Sets the default optimization options for various filters. The analysis files use this as baseline options and then change the relevant options for optimization.

`estimateAttitude2.m`


















Computes the orientation for a given filter and the error with respect to ground truth, which is used as the cost function in the optimizations.

`optimizeAHRSParameters.m`

Sets the options for various optimization solvers based on the options provided by the user.

`optimizeAHRScrossDataset.m`

Handles organizing many optimizations to be run sequentially, across various subsets of data. These subsets can be based on participant, condition, sensor, sampling frequency, and filtering algorithm.

 figuresAndTables	1/02/2022 6:22 PM	File folder	
 filterParameters	1/02/2022 6:22 PM	File folder	
 s1	27/01/2022 4:05 PM	File folder	
 s2	27/01/2022 4:06 PM	File folder	
 s3	27/01/2022 4:06 PM	File folder	
 s4	27/01/2022 4:06 PM	File folder	
 s5	27/01/2022 4:06 PM	File folder	
 s6	27/01/2022 4:07 PM	File folder	
 s7	27/01/2022 4:07 PM	File folder	
 s8	27/01/2022 4:07 PM	File folder	
 s9	27/01/2022 4:07 PM	File folder	
 s10	27/01/2022 4:05 PM	File folder	
 s11	27/01/2022 4:06 PM	File folder	
 s12	27/01/2022 4:06 PM	File folder	
 condition_filenumbers.xlsx	29/06/2021 1:36 PM	Microsoft Excel W...	13 KB
 condition_filenumbers_s1corrected.xlsx	22/12/2021 5:44 PM	Microsoft Excel W...	13 KB
 validation2021_dataset.mat	1/02/2022 6:23 PM	MATLAB Data	760,509 KB