Fast Orbit Response Matrix Measurements

G. Rehm, M. Furseman, I. Martin

DEELS workshop

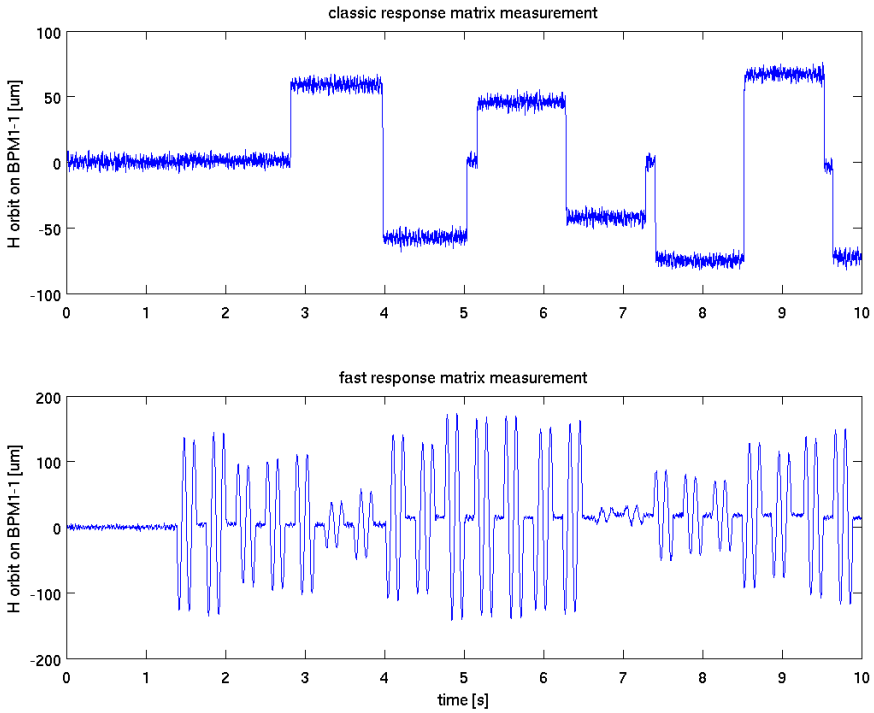
ESRF, 12-13 May 2014

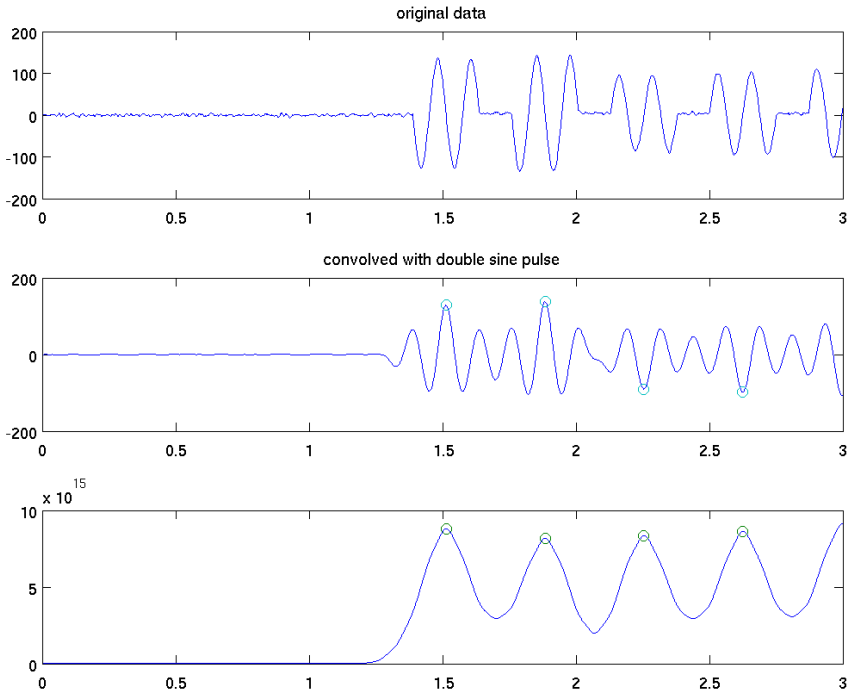
Overview

- ORM measurement normally done through Matlab/Middlelayer/AT
  - Steps one corrector up/down/back at a time
  - Takes a BPM readings at up/down from all BPMs
  - About 15 minutes for 344 correctors/BPMs
- Fast ORM
  - Excites correctors with single cycle 8 Hz
  - Steps through 8 correctors per second
  - All BPMs are recorded into fast archiver at 10072 S/s
  - About 43 seconds for 344 correctors/BPMs

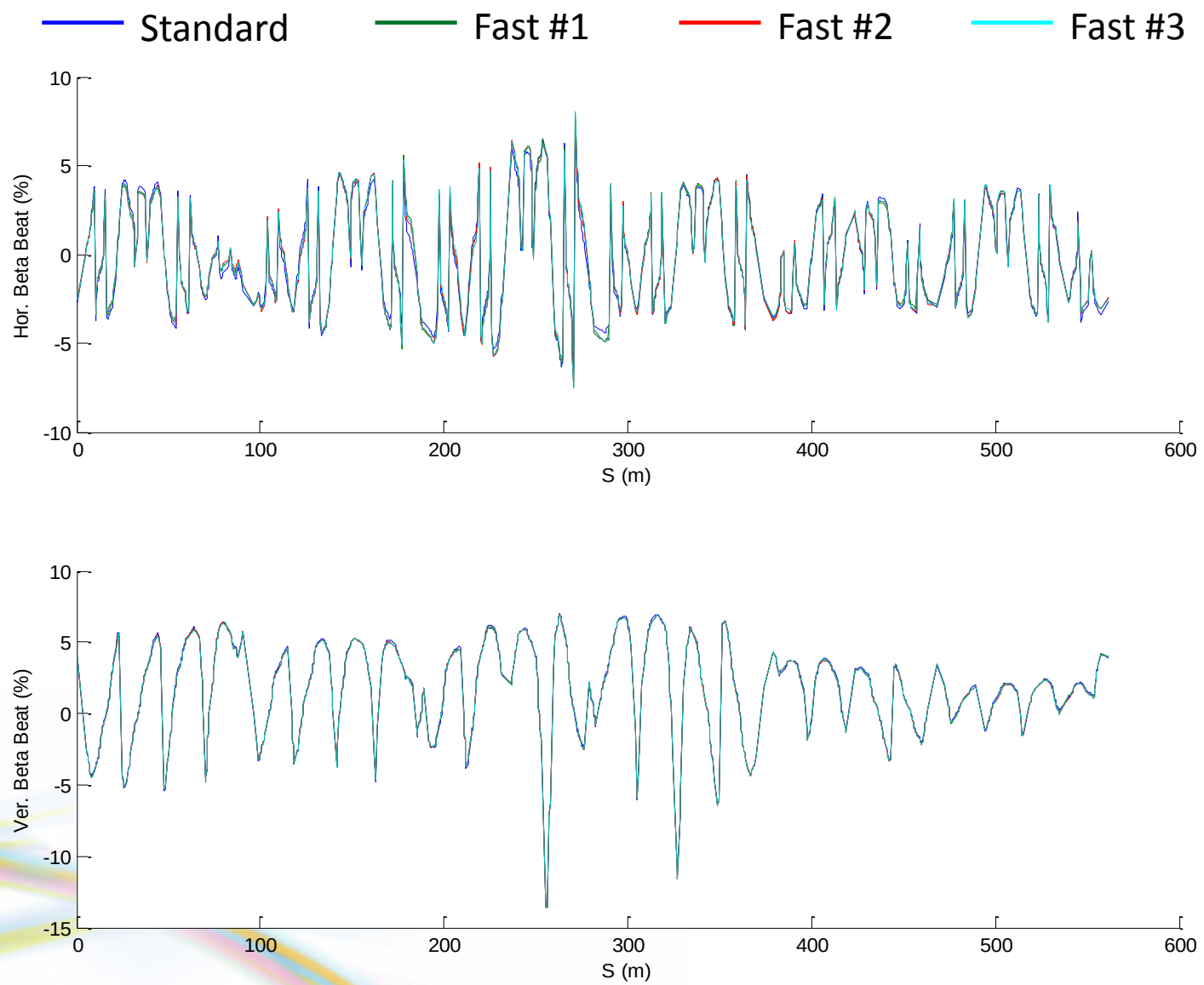
First Test

- Matlab script to drive on corrector at a time, then advance to the next
- Still using 2 cycles at 8 Hz
- Processing needs to identify position of excitations





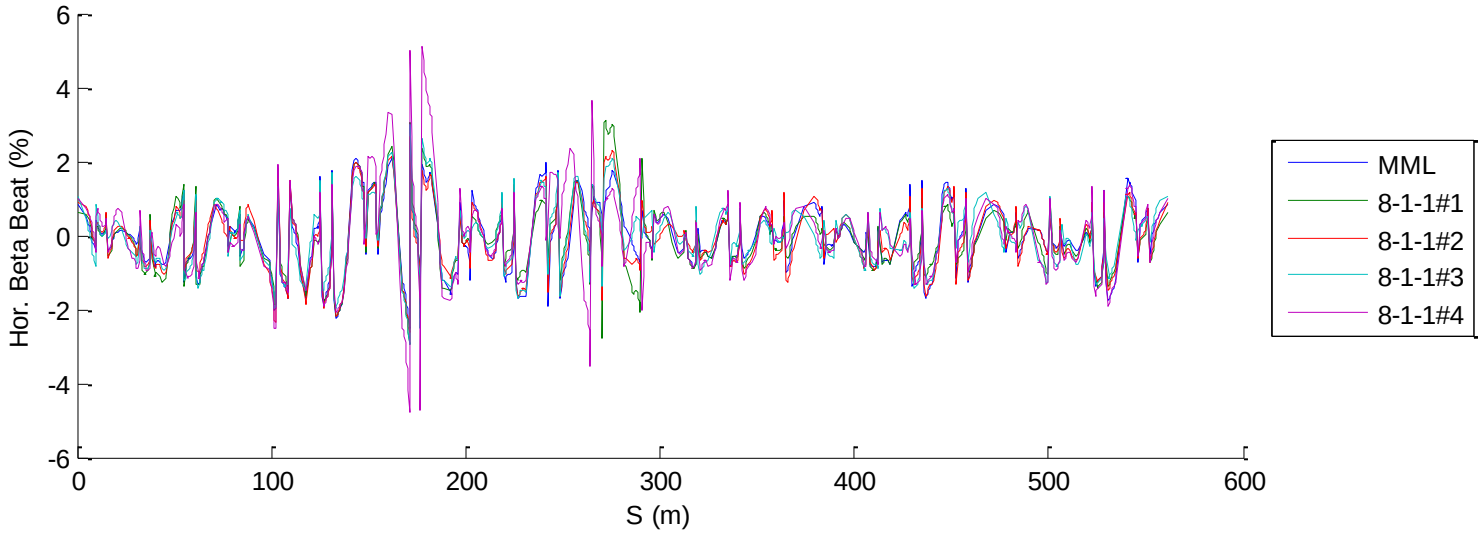
Compare LOCO fit with classic

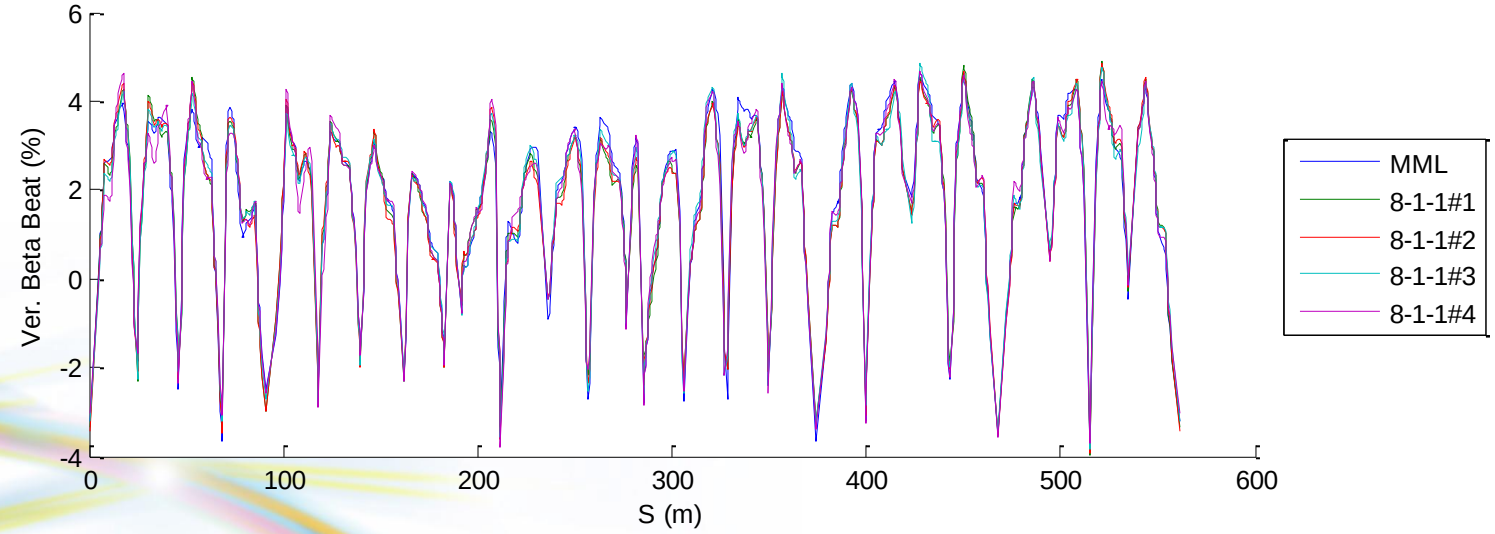


Final Implementation

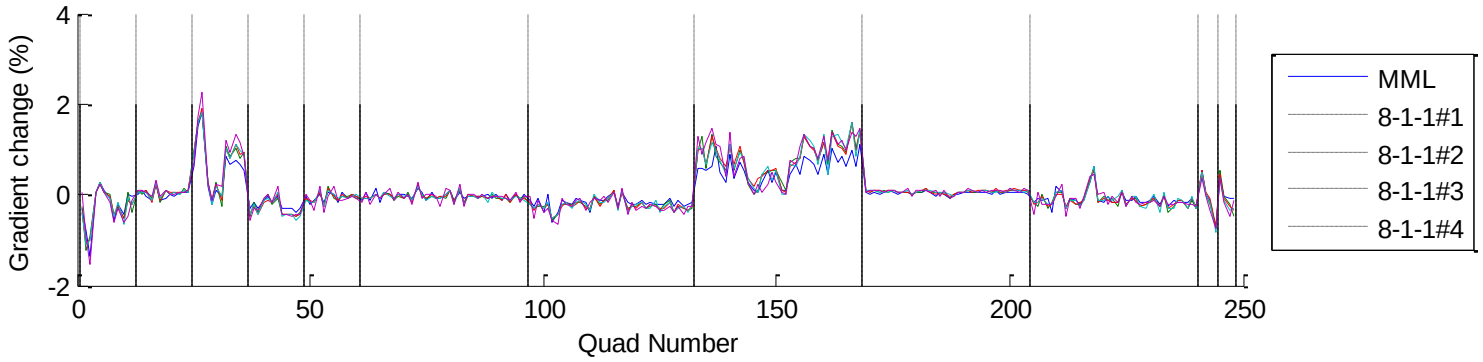
- Each feedback node can produce sine tones per corrector with programmable:
  - Amplitude and Frequency
  - Start time and Duration
- Communication Controller frame number acts as common time base between all nodes
- Python script enquires current time, then pre-programmes all correctors with their precise time and duration
- Fast Archiver records all BPM data
- Python script reads data from fast archiver once all complete and analyses using knowledge of time of excitation of each corrector, saves in LOCO compatible format

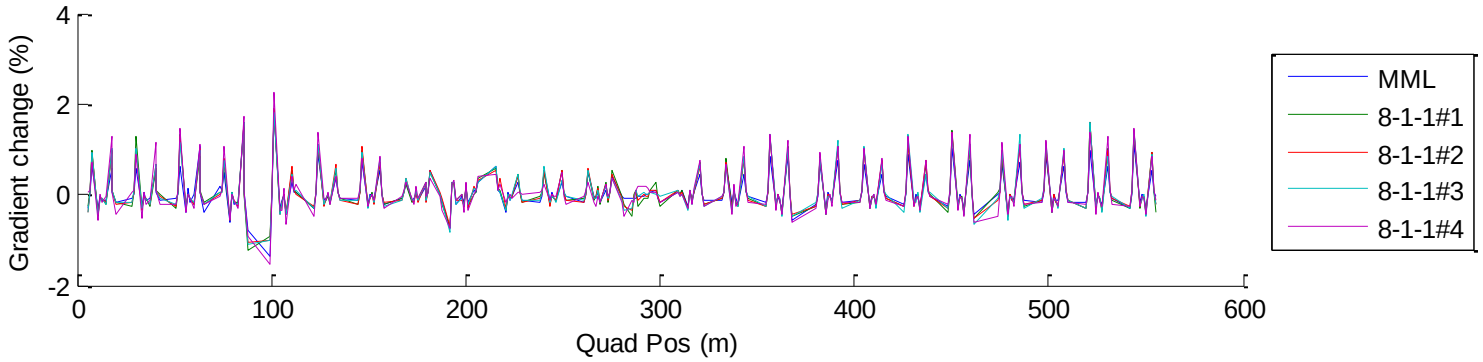
LOCO Results 1

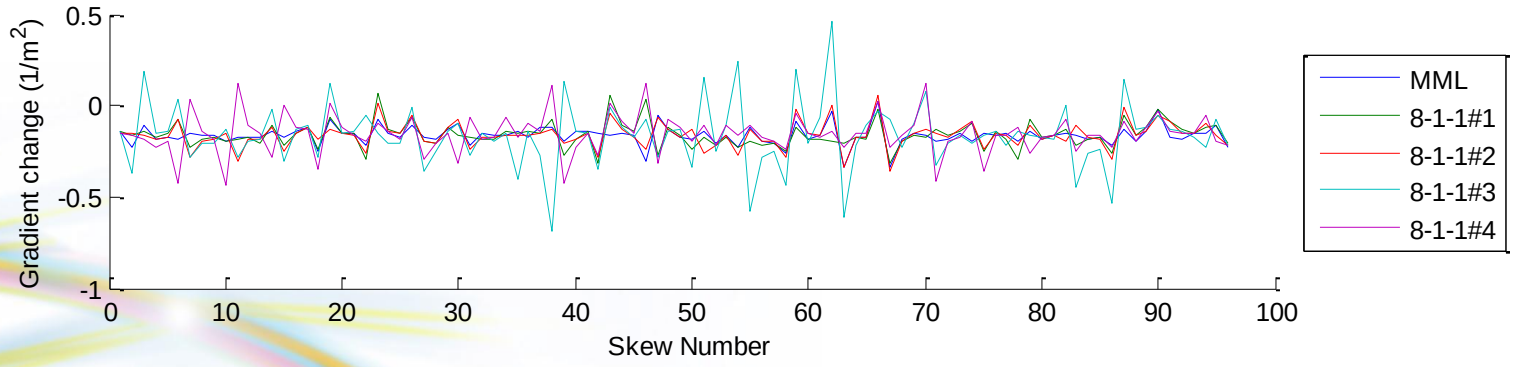




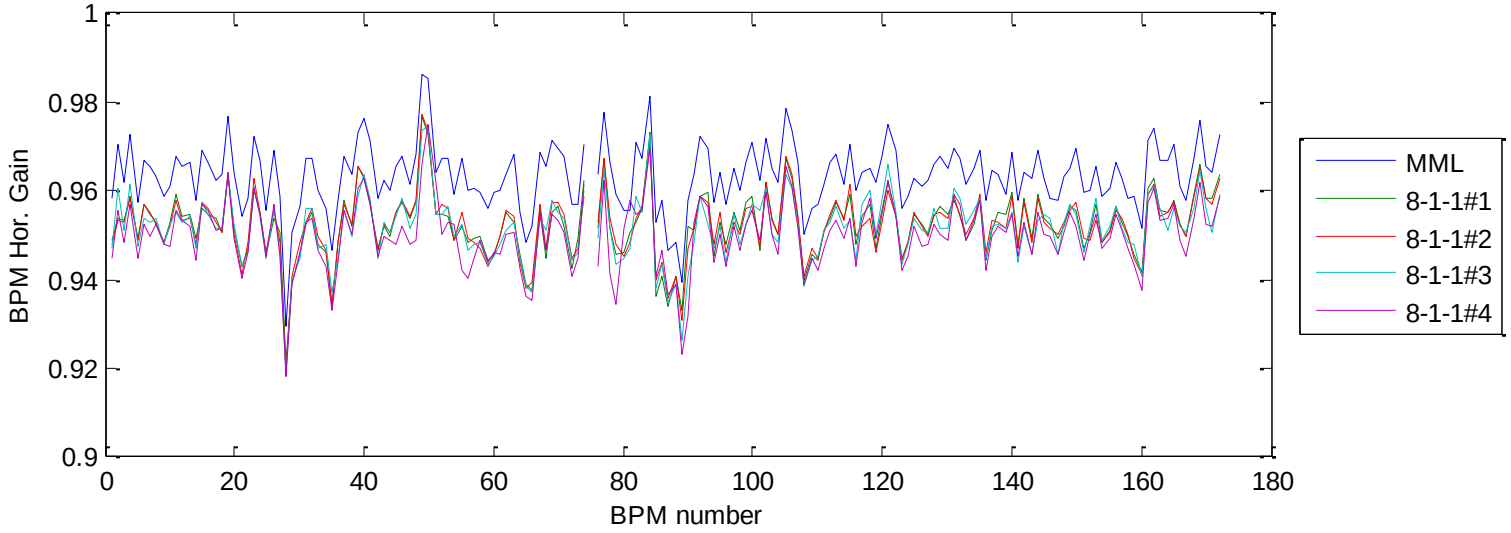
LOCO results 2

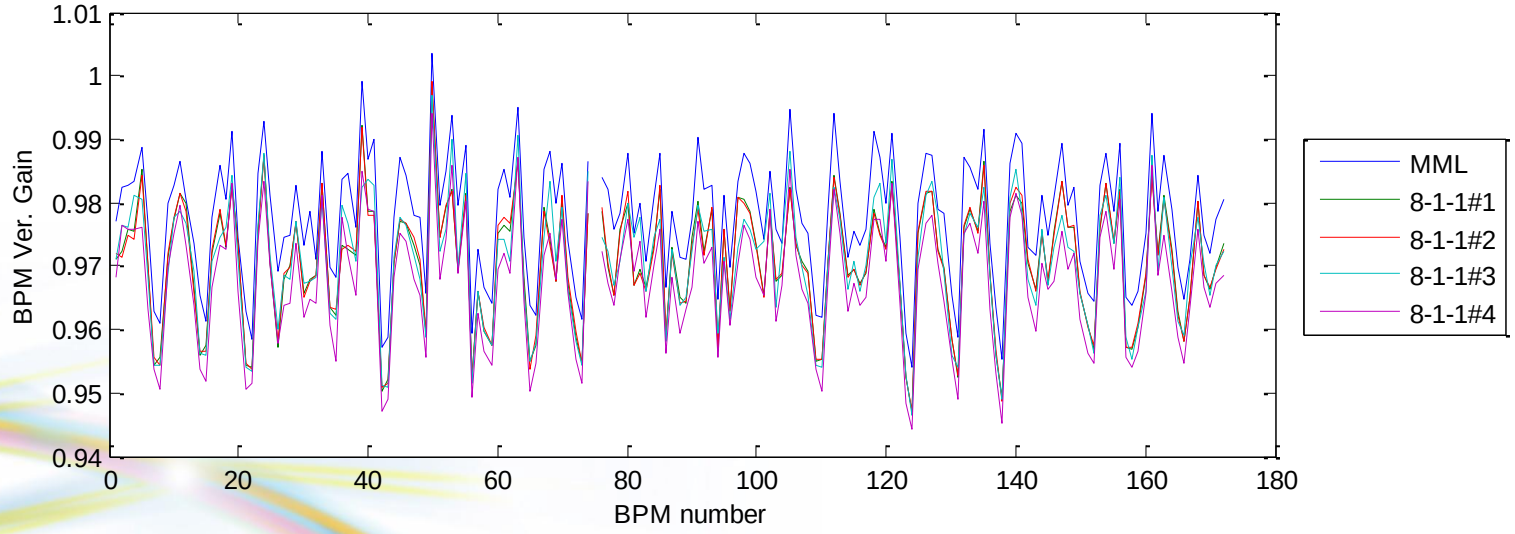






LOCO results 3





Summary / Issues

- Will save us about 14 minutes per ORM measurement
- This will not only facilitate more frequent optic corrections, but also allow using ORM for many tasks that previously took too long due to many repetitions required
- New ORM probably even more reproducible than old method, both due to less noise and shorter time between runs
- What's the best compromise between speed and precision? We've pushed it to a full ORM in 10 seconds!