ICALEPCS '23 – TH2BCO01

Synchronized Nonlinear Motion Trajectories at MAX IV Beamlines

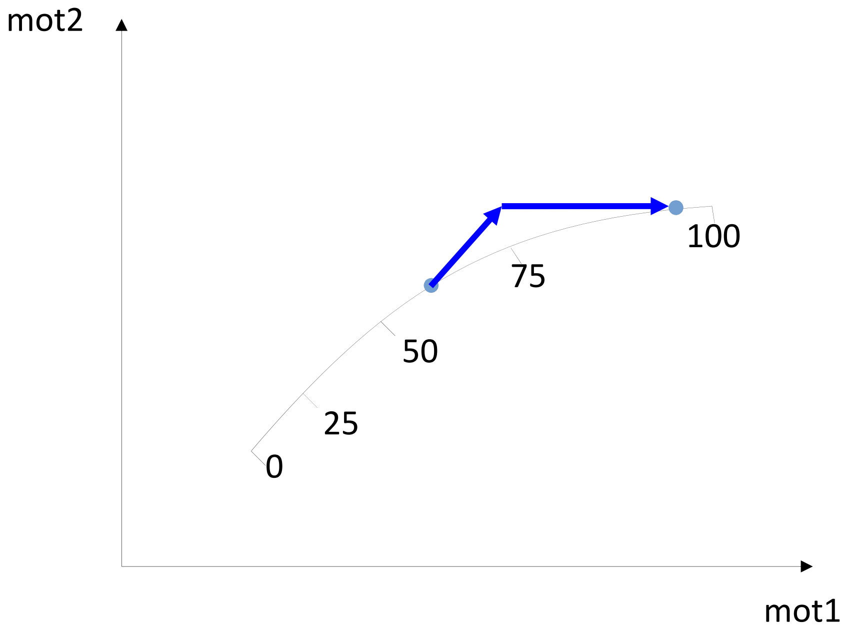
Peter Sjöblom, Henrik Enquist, Aureo Freitas, Julio Lidón-Simón, Miriam Lindberg, Suleyman Malki and many many more

Standard PseudoMotor

Two motors started simultaneously but finish independently.

Makes the motion deviate from path.

Typical step scan.

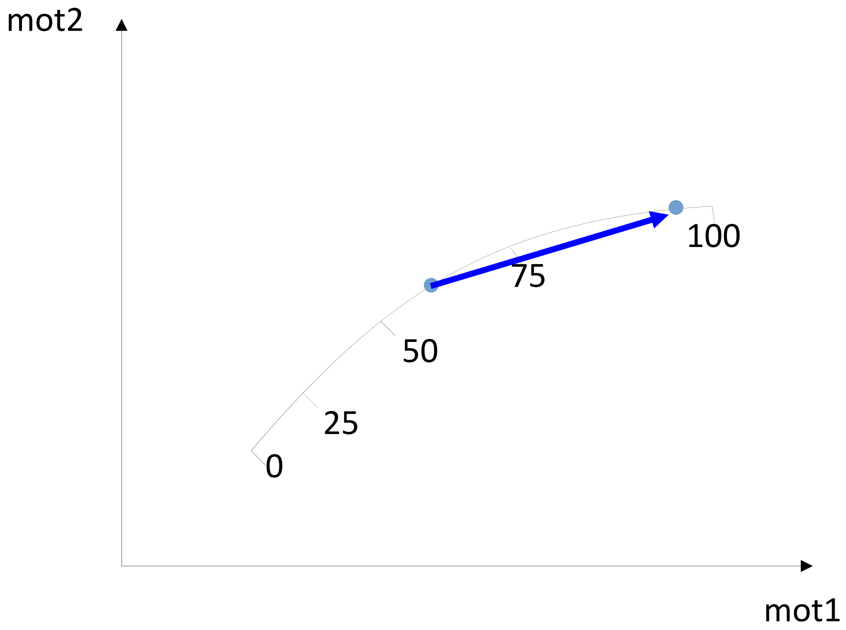


Standard PseudoMotor

With synchronized linear speed, motors start and finish simultaneously.

Cut corners in the trajectory.

Common in monochromators.

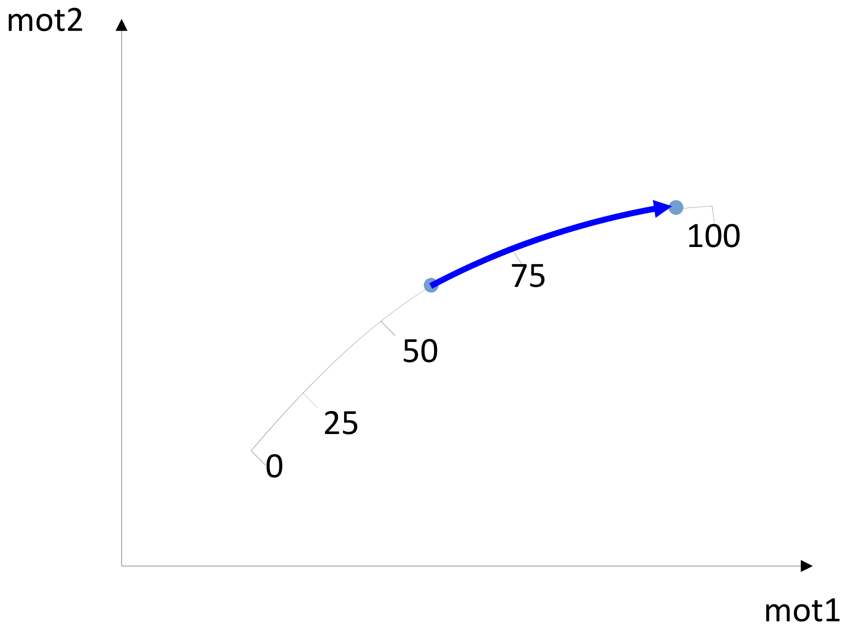


Following trajectory

A trajectory is followed when start, stop, speed and acceleration is adjusted along the path.

Possible to scan e.g. energy in constant speed.

Trajectories is now a function available in the IcePAP motion system through its collaboration.



Rail system

Vacuum pipe

Grating chamber

Mirror Chamber

Sample manipulator

Detector wagon

Bellow and support system

Beam from exit slit

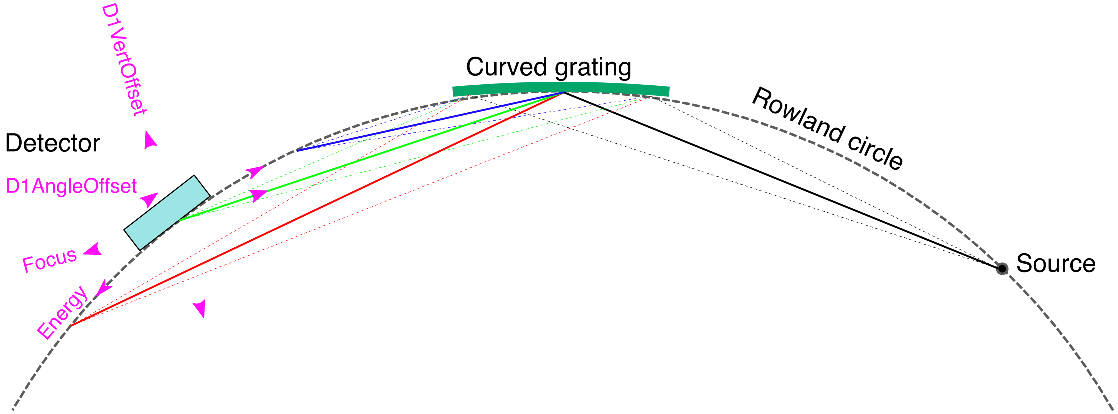
Veritas Rowland Spectrometer

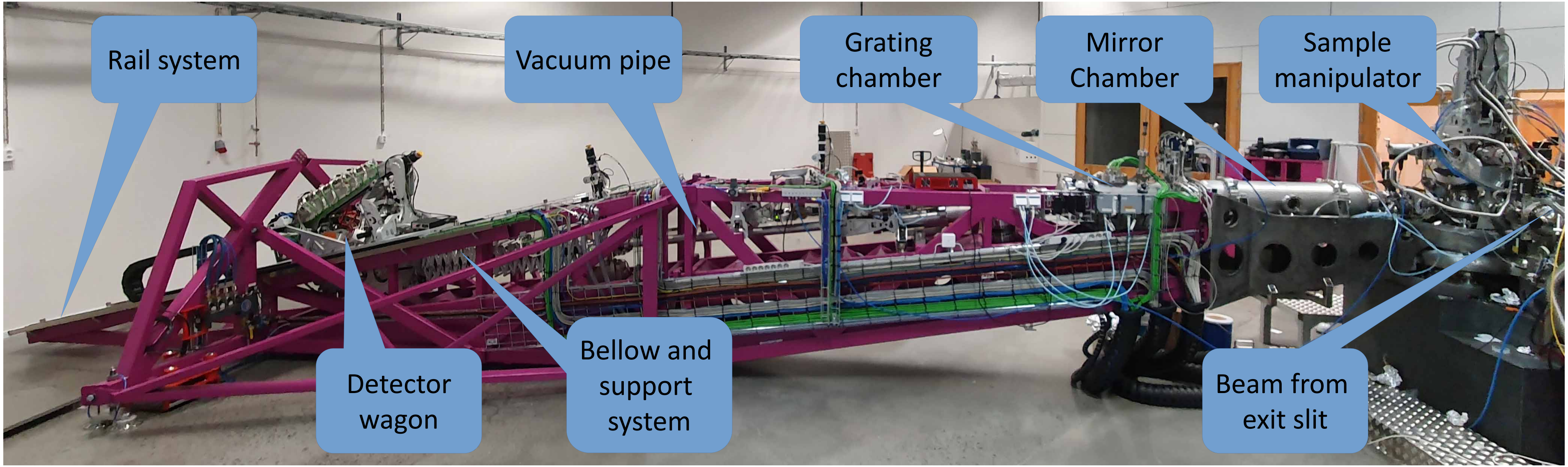
MAX IV

Rowland Spectrometer

With a curved grating with radius 2R, the detector will be in focus if the source, grating center and detector is on the same circle with radius R.

Great example where trajectories are useful.

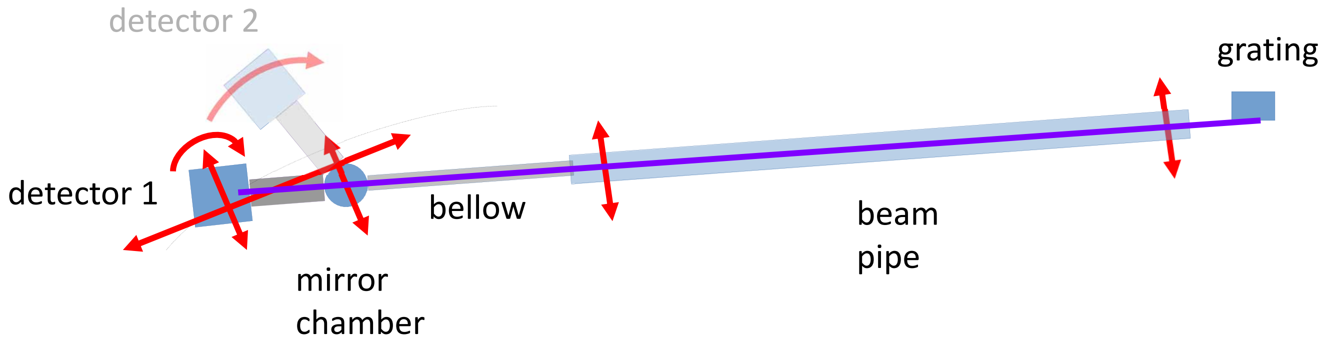




Veritas Rowland Spectrometer

MAX IV

Veritas Rowland Spectrometer
What to move?



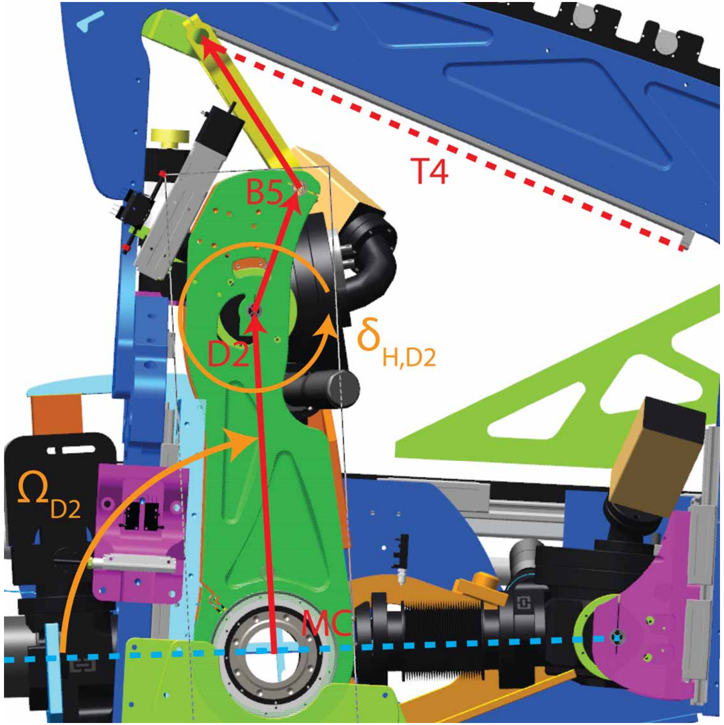
Parameters
- Energy
- Detector 1 angle adjust
- Detector 1 distance adjust
- Detector 1 height adjust
- Grating line density
- Detector 2 angle

Each motion needs to have its equation.

Trajectories makes updates easy.

Blue beam path travel through mirror camber to two detectors while motion and joints keep them aligned.

Detector wagon detail
Objects are interlinked



MAX IV

How to make a trajectory move

Function $f(energy)$ calculates positions for mot1 and mot2

Function $units2steps()$ converts from units to motor steps

Make the tables:

energytable = [ ]

mot1table = [ ]

mot2table = [ ]

for energy in range(0,100):

- mot1, mot2 = f(energy)
- energytable.append(energy)
- mot1table.append(units2steps(mot1))
- mot2table.append(units2steps(mot2))

The structure of a active trajectory uploaded in IcePAP


```
"ActiveTrajectory": {
    "_description": "Normal trajectory, 240.0-960.0, 9.0-9.0, 0.0-0.0, 0.0-0.0, 1350.0-1350.0, 67.0-67.0, 0.00262-0.00262, 0.03054-0.03054, ",
    "_table": [
[
    240.0,
240.72072072072072,
    ...
    ...
    ...
    ...
959.2792792792792,
    960.0
    ],
[
166.95959306707357,
166.42171860494807,
    ...
    ...
    ...
    ...
7.352565954100715,
7.414482558850775
    ],
[
-164.33317679657114,
-163.6535240867587,
    ...
    ...
    ...
    ...
55.33610679205606,
55.41263456218098
    ],
[
8.860391518052928,
9.249130018203317,
    ...
    ...
    ...
    ...
-71.4607670496774,
-71.5626280589275
    ],
[
2657.4049499999996,
2645.9161999999997,
    ...
    ...
    ...
    ...
-1062.1475500000001,
-1063.4650500000002
    ],
],
    "_time": "2023-09-27T01:14:24.841193+02:00"
  }
```

How to make a trajectory move

1. Upload table to each motor using pyIcePAP:
   ipap[mot1axis].clear_parametric_table()
   ipap[mot2axis].clear_parametric_table()
   ipap[mot1axis].set_parametric_table(energytable, mot1table, mode='LINEAR')
   ipap[mot2axis].set_parametric_table(energytable, mot1table, mode='LINEAR')

   Or 'SPLINE'

2. Move onto trajectory:
   ipap.movep(energytable[0], [mot1axis, mot2axis], group=True)

3. Set velocity:
   ipap[mot1axis].parvel = vel
   ipap[mot2axis].parvel = vel

   Can be different!

4. Move along trajectory:
   ipap.pmove(75.3, [mot1axis, mot2axis], group=True)

   Important!

Various notes

- Each motor can have a different parametric velocity.

- Without group mode: If one motor stops due to some error the others continue.

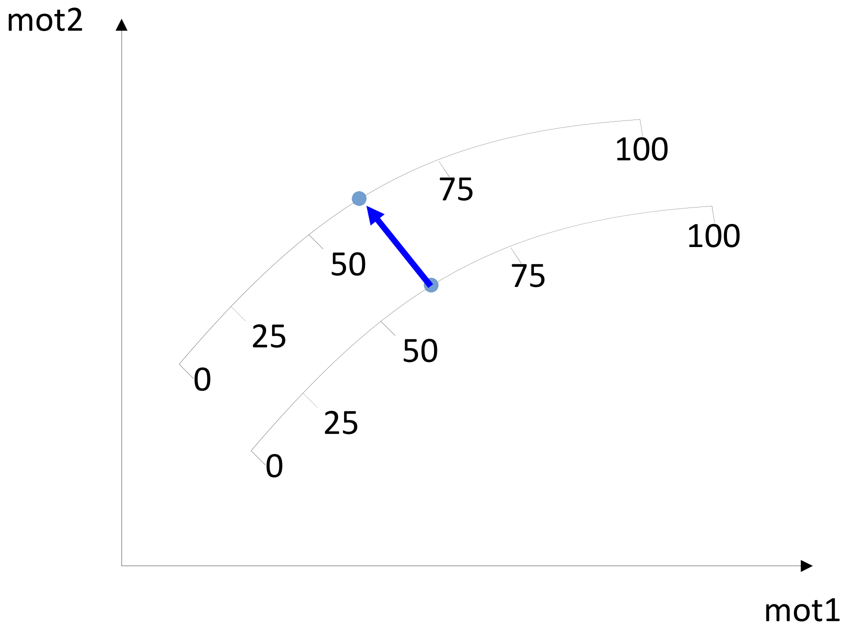
- It's still possible to move each motor individually, doing so breaks the trajectory "magic" and it's necessary to do a MOVEP to get them back in sync.

- Trajectory coordinate can be read from any axis:
  Pos = ipap[mot1axis].parpos
- If a motor is not on the trajectory this will raise an exception.

Changing other parameters
Method 1: keep trajectory coordinate

Most requests result in small changes in trajectory.

The new setpoint will be the same value on the new trajectory.

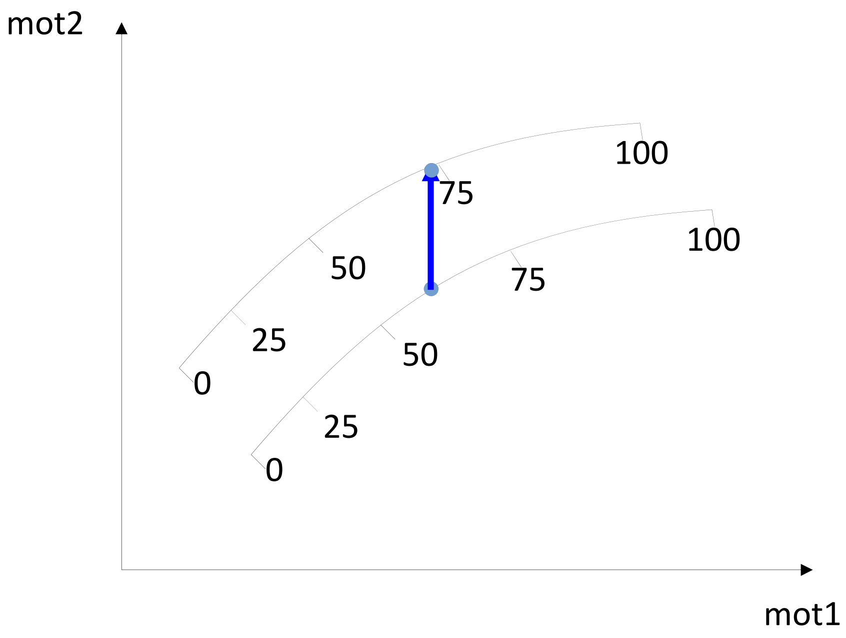


Changing other parameters
Method 2: keep master position

With a new trajectory resulting in a large time consuming motion, a new setpoint is defined as the physically closest point.

The way between them are traveled through a temporary trajectory

Automatically uploaded, removed and replaced with the final trajectory when the change is completed.

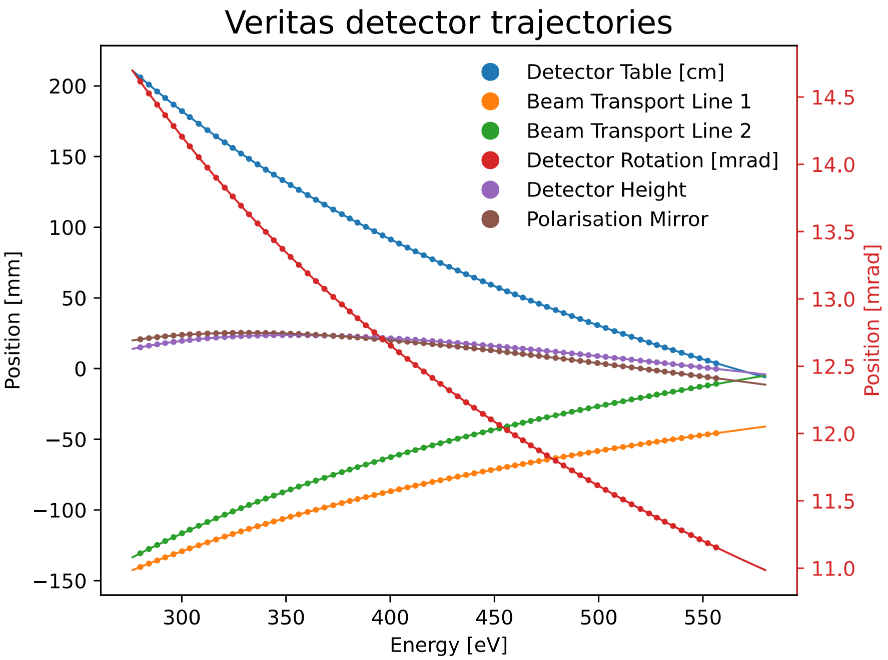


Prior motion, the trajectory is calculated and uploaded and here plotted as a solid line.

During motion, the tango attribute position is plotted as dots and laid in the same graph.

The two datasets matches well with each other.

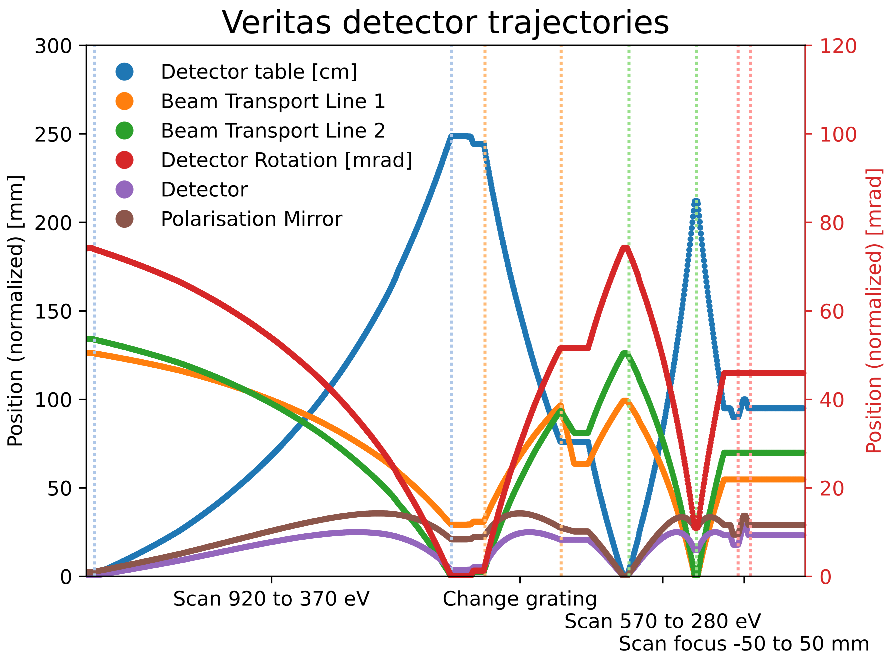
Measured position and uploaded trajectory



Trajectories at work

During operation, the trajectories need to be followed at all time.

1. Energy scan
2. Grating exchange
3. Energy scan
4. Focus scan



Fitting routine

User Parameters

- Detector height offset
- Detector rotation offset
- Detector focus offset
- Grating line density
- Grating curvature

- The exact trajectory of the Rowland geometry is very sensitive to parameters such as grating radius and line density.
- By finding a few known points along the energy scale, by using samples with well-known emission, it's possible to fit the parameters.
- Given the recorded detector positions at these known points, the routine optimizes the Rowland parameters to generate a trajectory that gets as close as possible to these positions, at the correct energies.
- The output is a set of adjusted parameters that can be entered in the Sardana motor controller.

Summary

- Trajectories successfully brings new capabilities to MAX IV motion systems in terms of motion along non-trivial and non-linear paths.
- Parameter to follow can be arbitrarily chosen as e.g. energy or a curvature in space.
- A large set of motors can be treated as one unit and the complex motion is solved without the need of interference from the user.
- They are deployed in production at several instruments including
  - SCANIA spectrometer at Balder beamline
  - flight tube at CoSAXS beamline
  - monochromators at FlexPES and FinEstBeAMS beamlines
  - Veritas Rowland Spectrometer