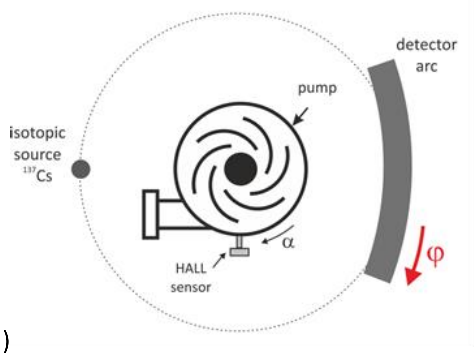
# Parallelisation of an algorithm for the projection data reassignment

**Background**:

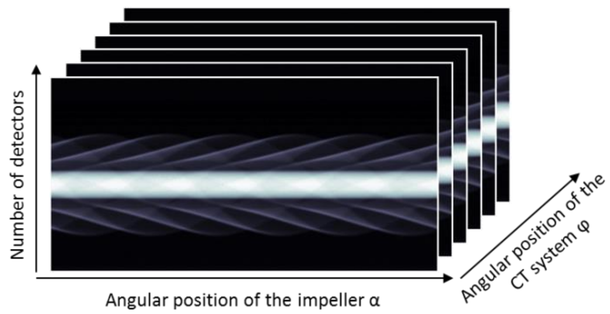
With the spatial high resolution gamma-ray computed tomography measurement system for high energy gamma radiation (662 keV) dense objects with up to a diameter of 700 mm can be non-destructively analysed with an in-plane resolution of approximately 2 mm. For a CT scan so called projections are acquired from various angular positons $p$ in an interval of $0 < \varphi \leq 2\pi$ with a radiation detector containing $d$ = [1 … $N_{det}$] single detectors. With the help of standard reconstruction algorithms the non-superimposed cross-sectional image can be obtained. For the sharp visualisation of fast rotating objects $f_{object}$, the detector is sampled with $f_{detector}$. The number of projections $N_{proj}$ per rotation is than defined as:

$$N_{proj} = \frac{f_{detector}}{f_{object}}$$

Because for one rotation of the object the signal to noise ratio is worse, over a certain number of rotations $N$ must be measured. Subsequently, projections for the same angular position of the object $\alpha$ are averaged. Therefore, a so called zero-crossing signal from the rotating object $S_{object}$ is used to synchronise the projection data set. If such time-averaged and rotation-synchronised CT data sets are acquired from different gamma-ray CT system positions $\varphi$ (synchronisation by another positioning signal device $S_{CTSystem}$), new projection data set can be resorted in which the static as well as rotating parts can be sharply mapped in the reconstructed image.



a)



b)

**Figure 1:** a) sketch of the measuring principle and b) obtained data sets of the rotation synchronized GammaCT.

**Task**:

The available single core C++ program (32-bit compiler) has to be analysed and optimised concerning parallelisation, i.e. multi core CPU and/or many core GPU application. Subsequently, the following data processing steps must be performance-optimised executed:

- Data conversion (from serial data stream to a „Byte“ data stream)
- Projection data subtraction
- Projection data averaging
- Searching for trigger signals $S_{object}$ and $S_{CTSystem}$ within the projection data stream
- Projection data interpolation to a constant projection number $N_{proj}$ per rotation (due to fluctuations)
- Generation of averaged projection data sets depended on the CT system positon
- Saving the data

**Contact**:

André Bieberle (HZDR) | a.bieberle@hzdr.de | +49 351 260 2913