

# Reproducing LhARA stage 1 BDSIM results

WP6 LhARA meeting

16/02/2026

Marie Mannie-Corbisier

Supervisor: William Shields

# Proton source definition (from CDR draft)

	Previous	Current
rpmax (sin( $\alpha$ )) (Angle cutoff)	0.005 ( $\neq \sin(2.75^\circ)=0.05$ )	0.0287( $=\sin(1.64^\circ)$ )
E_laser	10J	70J
E_proton_ref	15 MeV	15 MeV
Kmax (max K energy emitted)	None	25 MeV
Thickness	5e-6 m	4e-7 m
Duration	4e-14s	2.8e-14s
Laser Power (P)	2.5e14 W	2.5e15 W
Laser Intensity (I)	4e20 W/cm <sup>2</sup>	4e20 W/cm <sup>2</sup>
Laser spot radius (sqrt(P/(I* $\pi$ )))	1.5e-6 m	5.6e-6 m

Current values: CDR =>[80] N. Dover et al. LhARA linear optics documentation. Report LhARA-TN-07.

<https://ccap.hep.ph.ic.ac.uk/trac/raw-attachment/wiki/Research/LhARA/Documents/LhARA-TN-2024-07.pdf>. (CDR p.13)

# Code issue resolved in 2BDSIM.py

```
opts, args = getopt.getopt(argv,"hd:i:o:l:n:",["ifile=", "ofile=", "loc", "nEvts"])
```

└ Location ("loc") of the element in the beamline is an argument

```
iPrtcl.writeParticleBDSIM(ibmIOw.getdataFILE(), 1, True)
```

└ Previously not used

```
iPrtcl.writeParticleBDSIM(ibmIOw.getdataFILE(), iLoc, True)
```

└ Changed

# Code issue resolved in writeParticleBDSIM.py

```
try
    if self.getDebug():
        with np.printoptions(linewidth=500,precision=5,suppress=True):
            print("      ----> iAddr, iLoc, name:", iAddr, iLoc, \
                BLE.BeamLineElement.getinstances()[iLoc].getName())
            print("      ----> Trace space:", self.getTraceSpace()[iAddr])

    """
    reportString = "distrFileFormat = " +
        "x[m]:y[m]:z[m]:xp[rad]:yp[rad]:E[MeV]\n";
    """

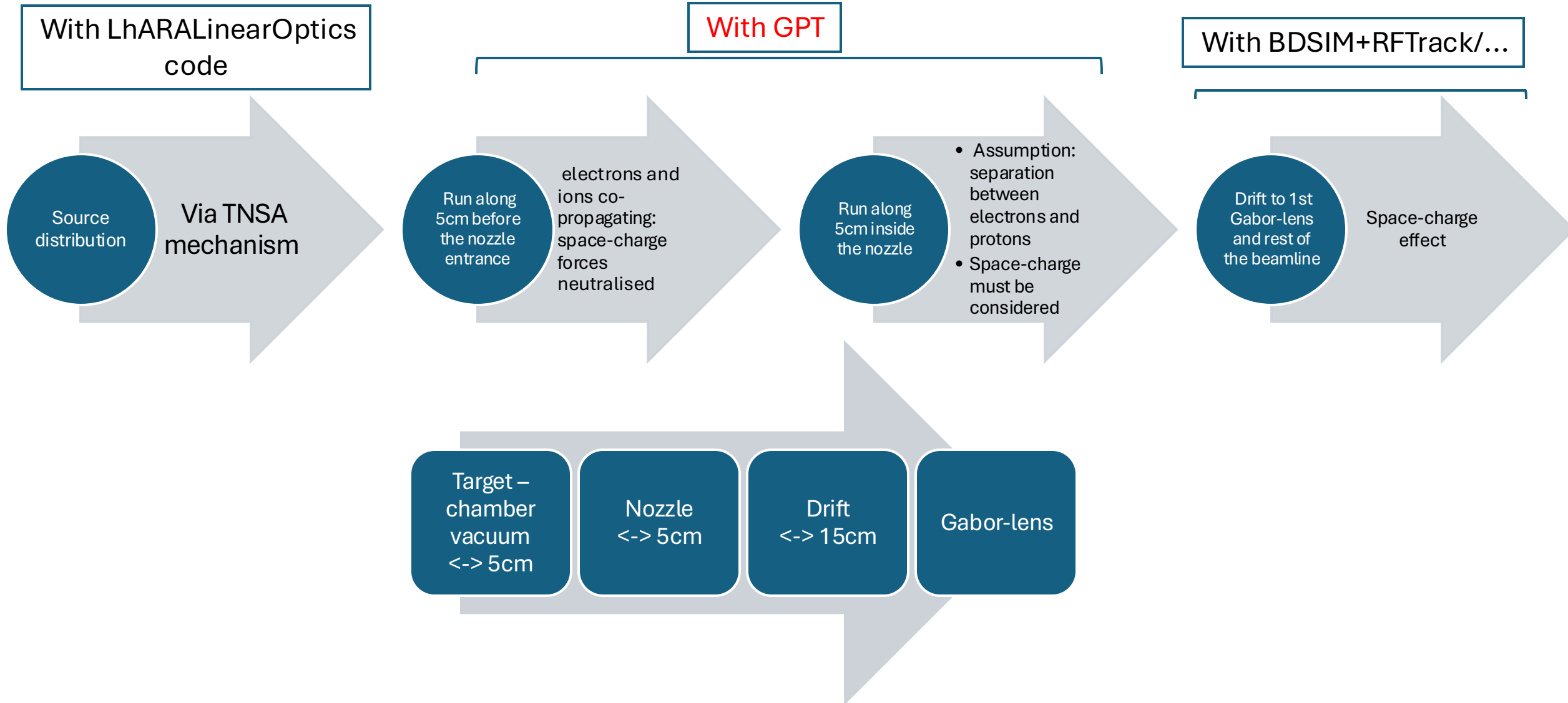
    Line = str(self.getTraceSpace()[iAddr][0]) + ' ' + \
        str(self.getTraceSpace()[iAddr][2]) + ' ' + \
        str(z) + ' ' + \
        str(self.getTraceSpace()[iAddr][1]) + ' ' + \
        str(self.getTraceSpace()[iAddr][3]) + ' ' + \
        str(E) + "\n"

    ParticleFILE.write(Line)
    if self.getDebug():
        print("      ----> Line:", Line)

    if CleanAfterWrite:
        Cleaned = self.cleanParticles()
except IndexError:
    pass
```

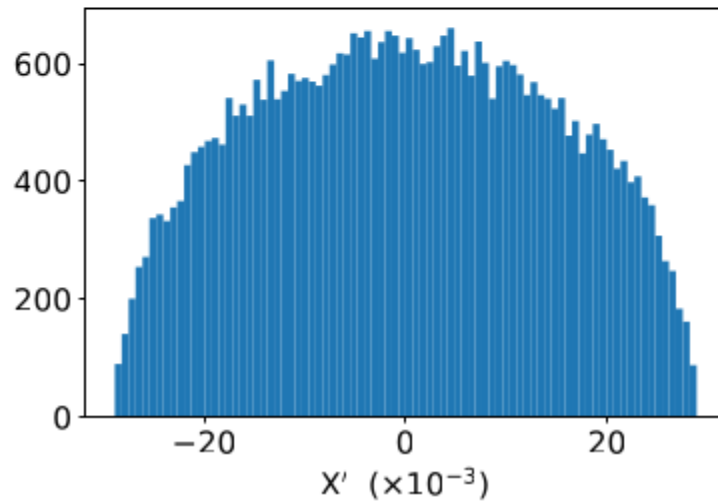
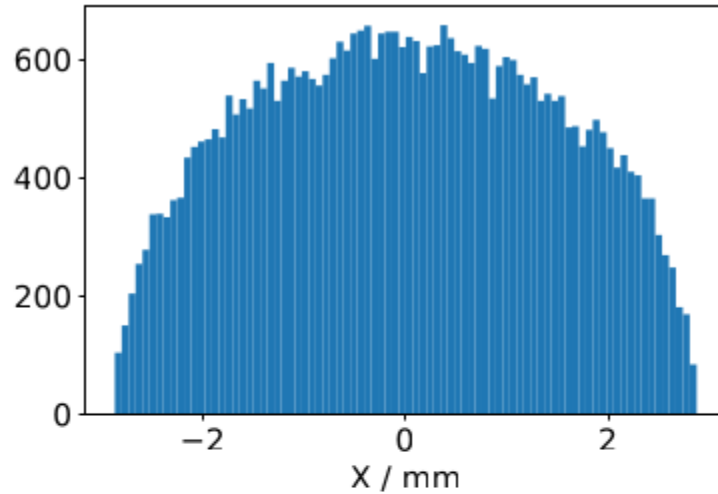
**'Try-except' added to handle cases where the particle did not reach the element of interest**

# Simulation workflow



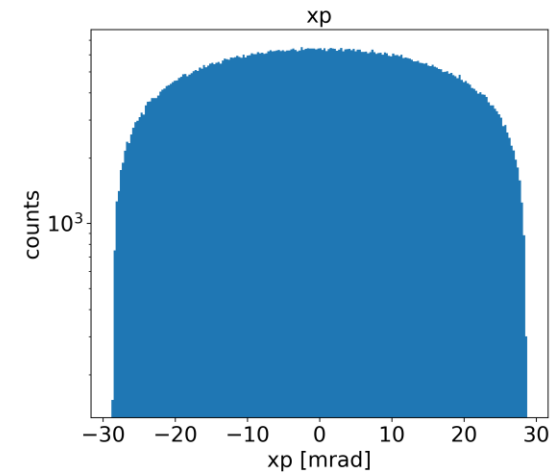
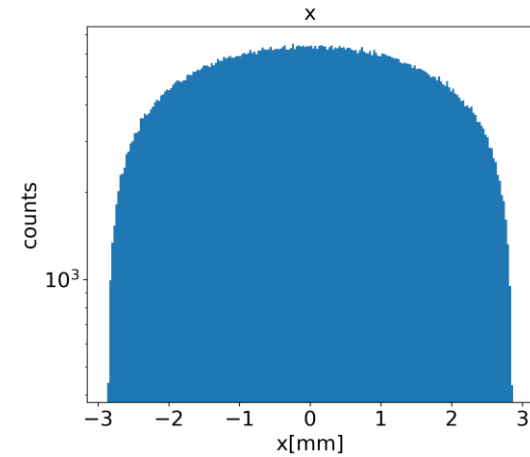
# Source distribution at the exit of the nozzle **without GPT**

## CDR

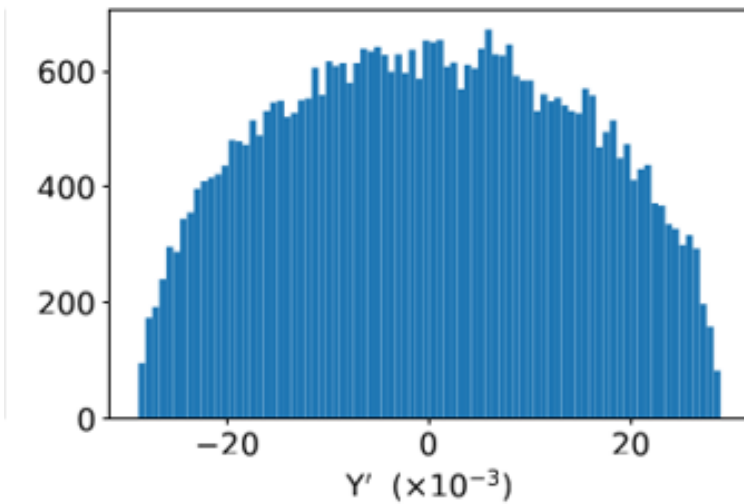
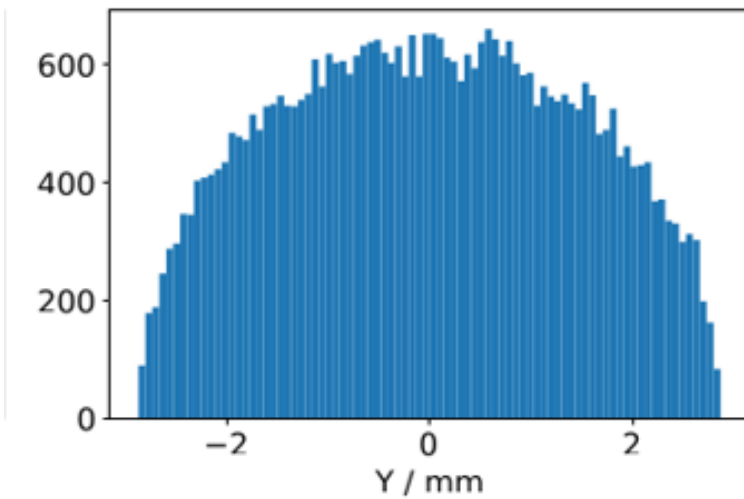


Correct

## Myself

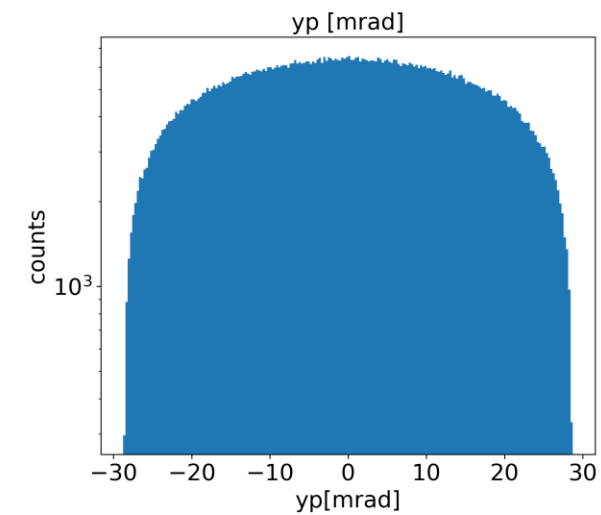
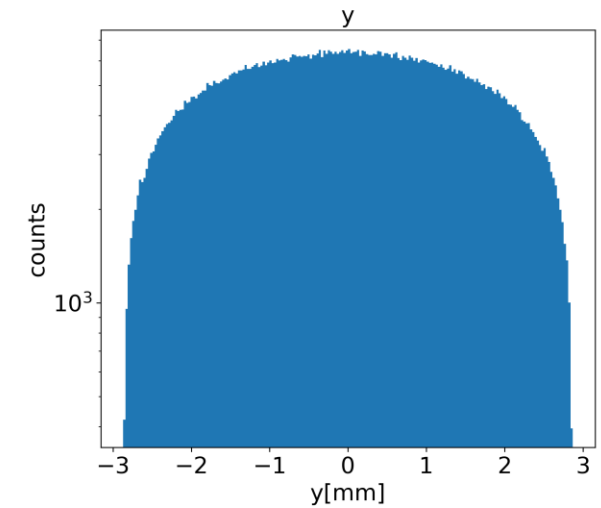


# CDR

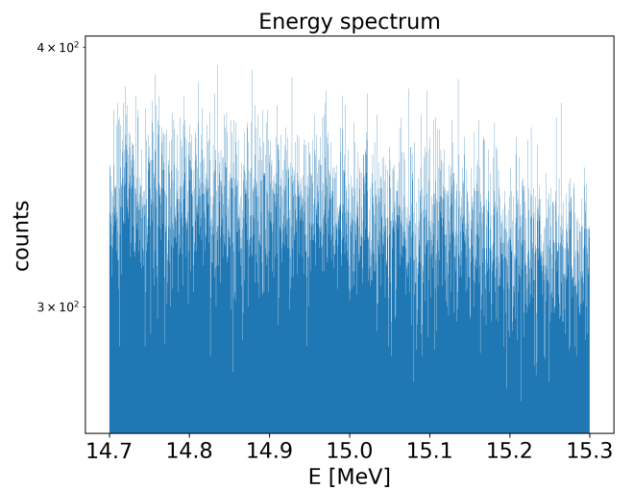
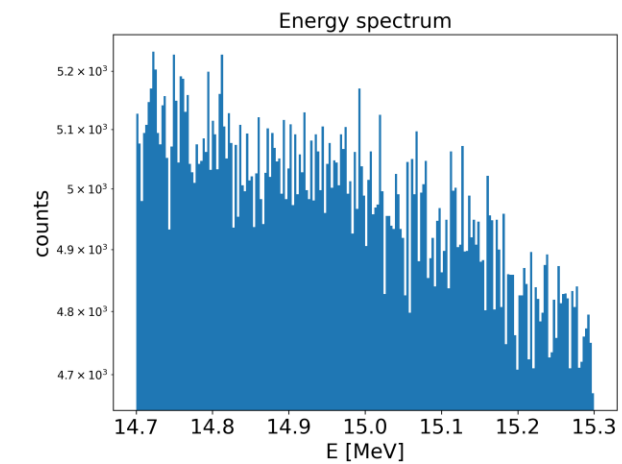


Correct

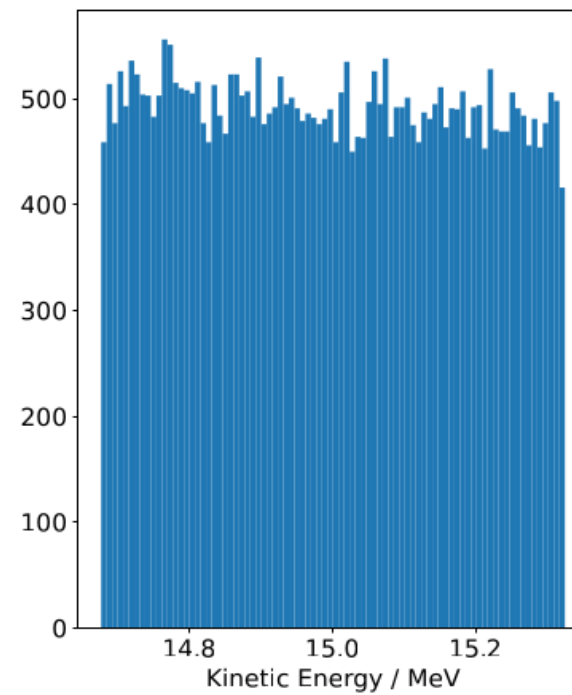
# Myself



# Myself

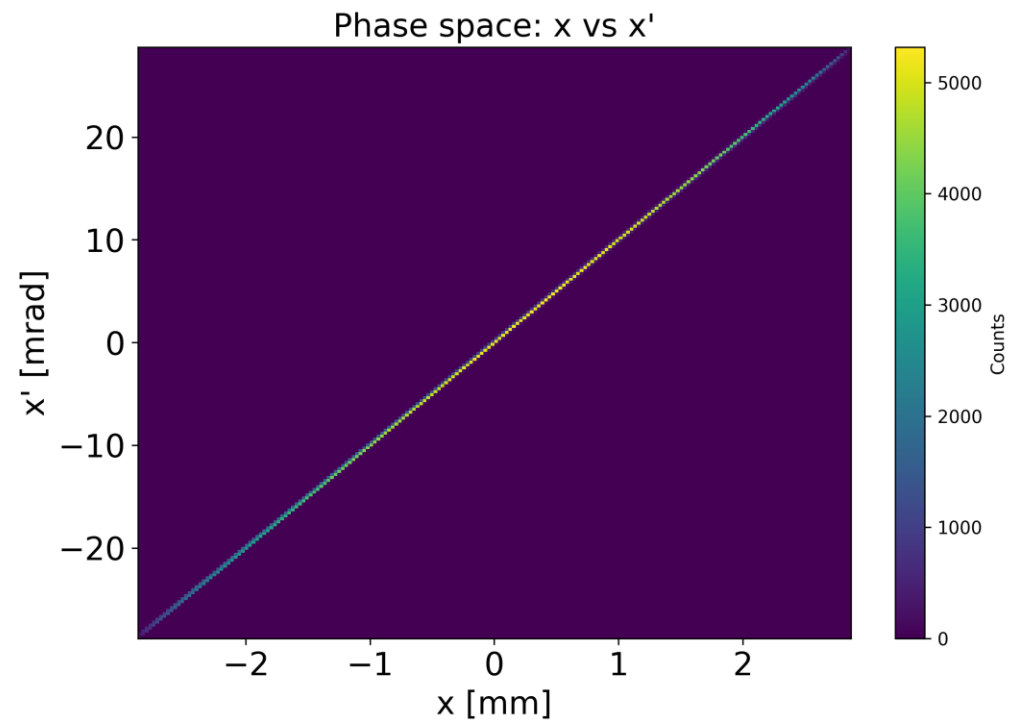


# CDR

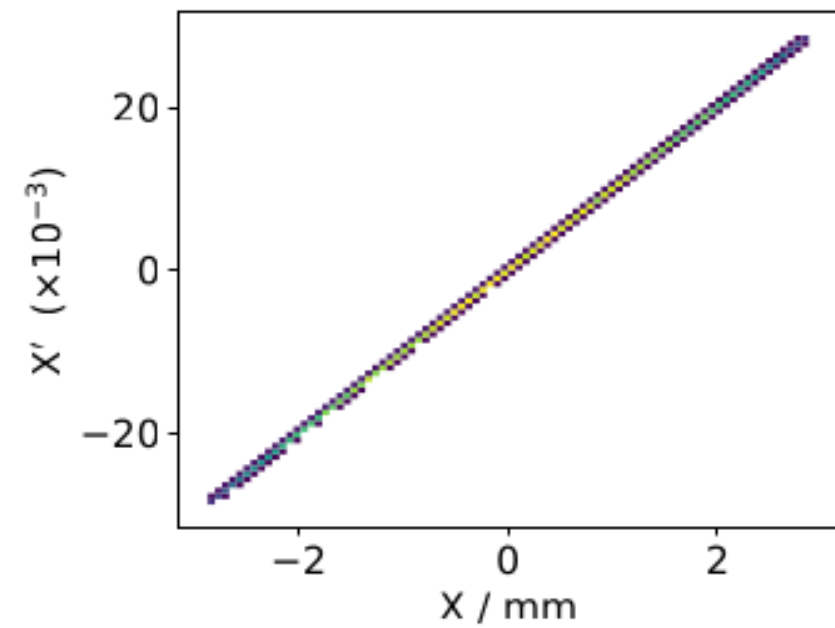




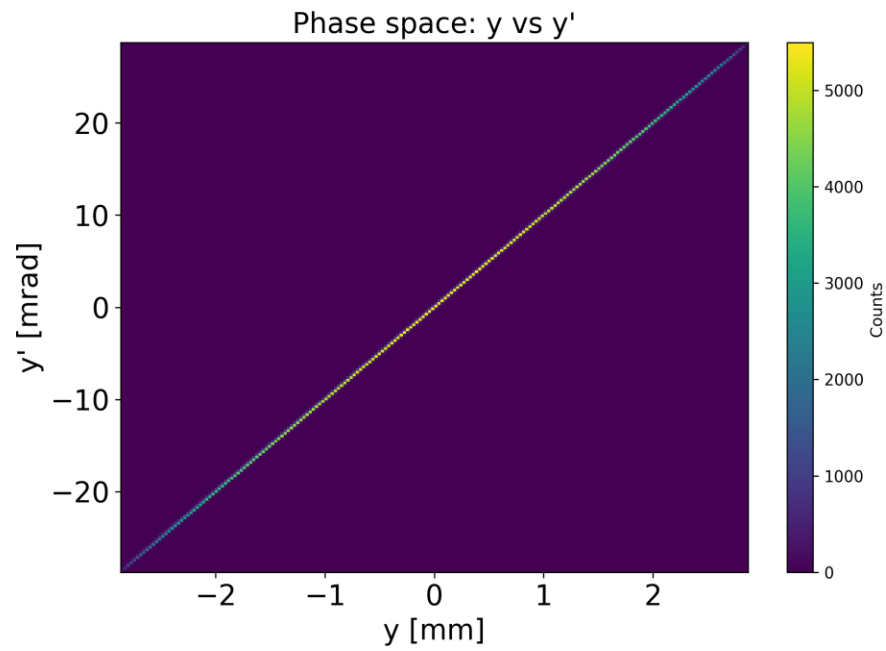
# Myself



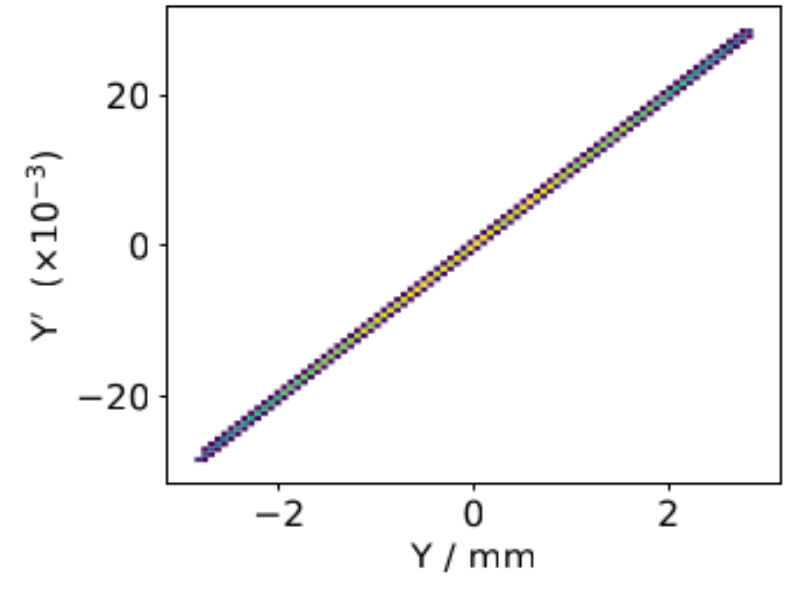
# CDR



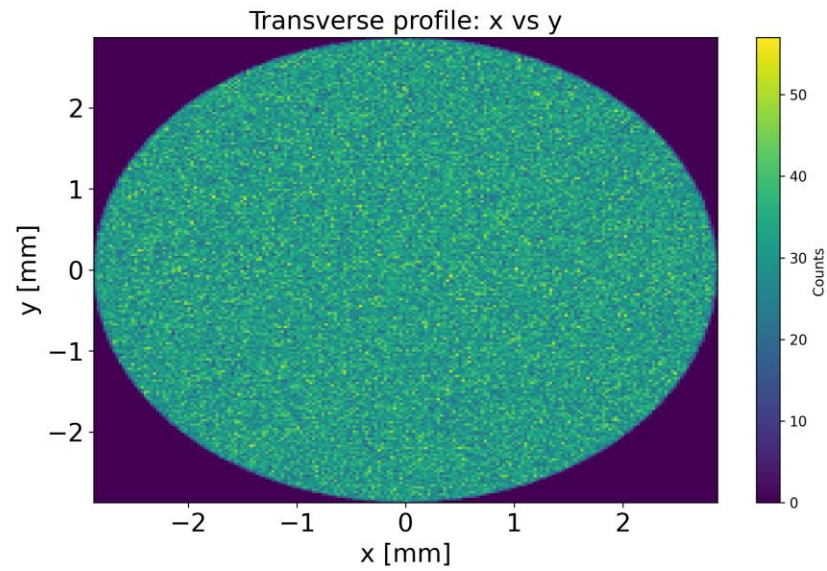
# Myself



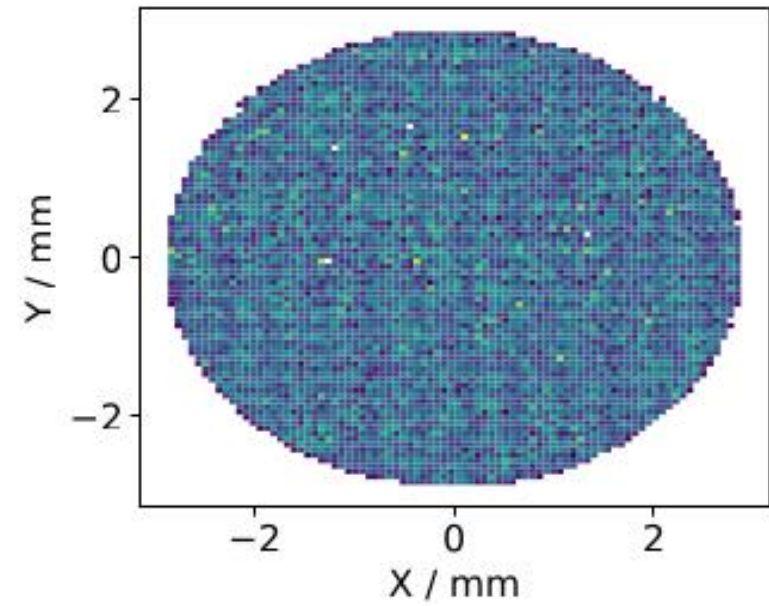
# CDR



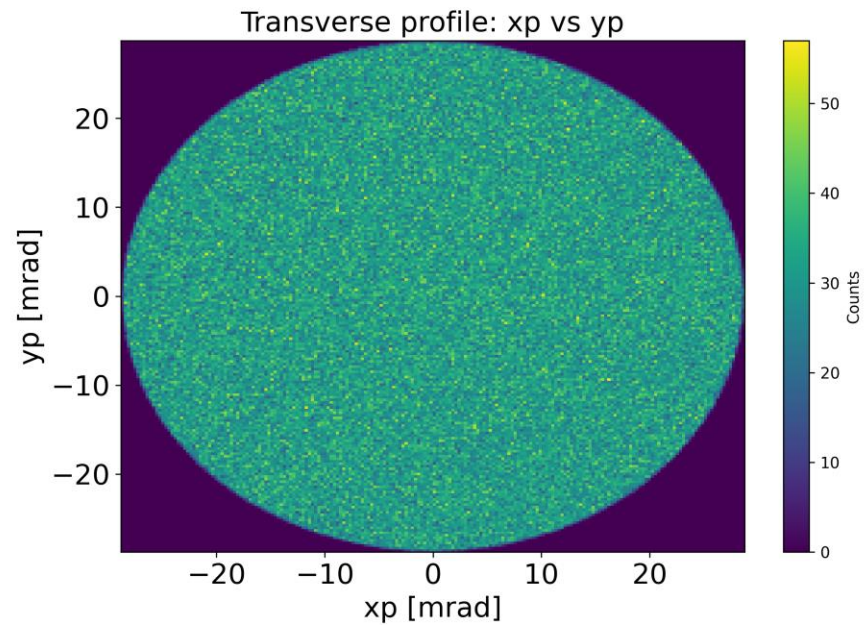
# Myself



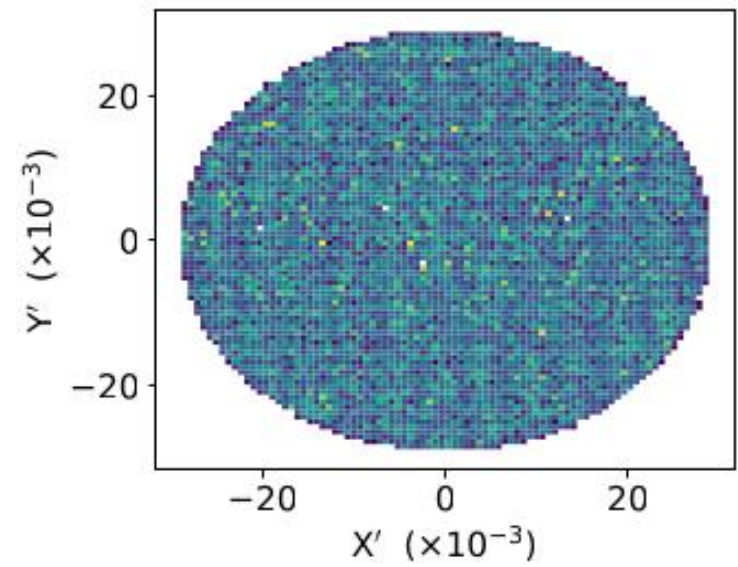
# CDR



# Myself



# CDR

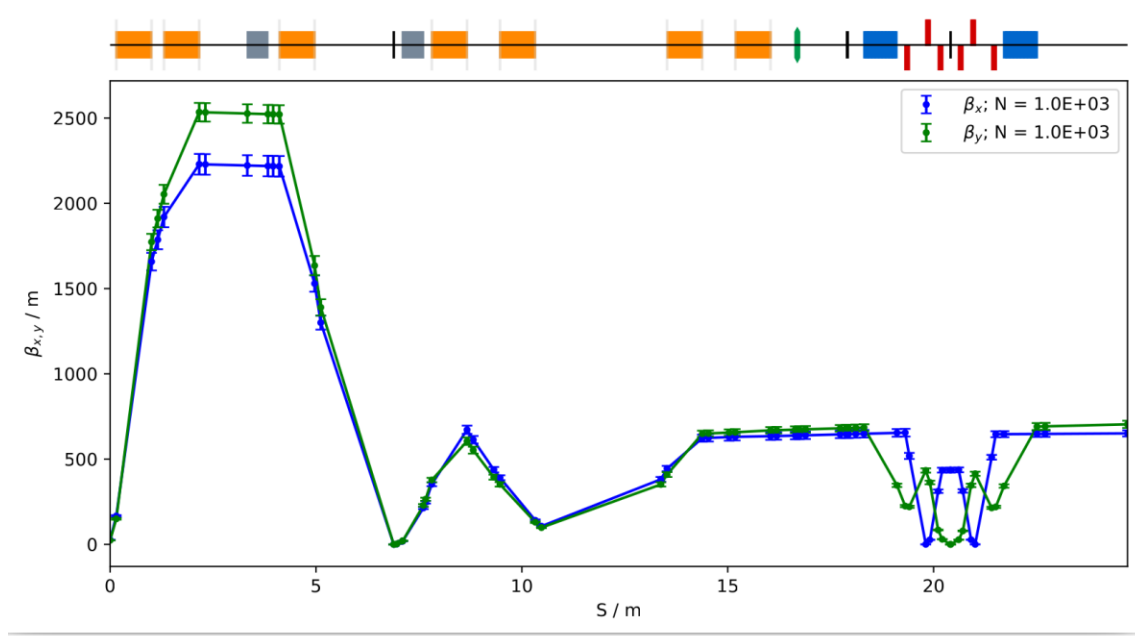


# TWISS parameters from the source for BDSIM optics simulations

	CDR	Myself
betax	35.35 m	25.289 m
betay	35.51 m	24.807 m
alphax	-355	-252.883
alphay	-356	-252.304
Dx	?	7.918e-06 m
Dy	?	1.857e-05 m
Dxp	?	7.897e-05 rad
Dyp	?	1.844e-04 rad
$\epsilon_x$	5.91e-8 m rad	8.105e-08 m rad
$\epsilon_y$	5.87e-8 m rad	8.366e-08 m rad

# BDSIM optics results

## Myself



## CDR

