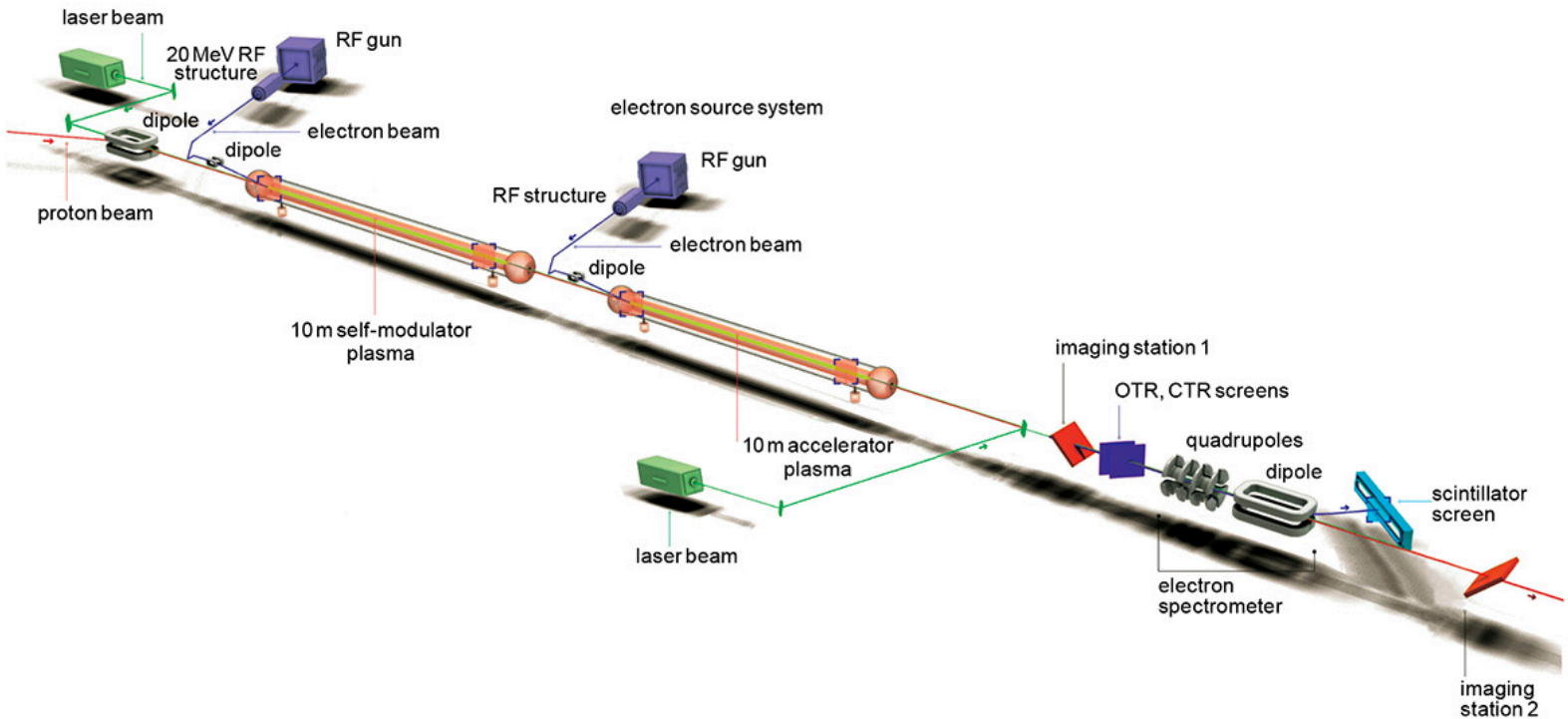
Finding tolerances for the future diagnostics of AWAKE Run 2c

M. Moreira[1], J. Farmer[1, 2], E. Gschwendtner[1], P. Muggli[1, 2], and M. Turner[1]

[1] CERN, Geneva, Switzerland

[2] Max Planck Institute for Physics, Garching, Germany

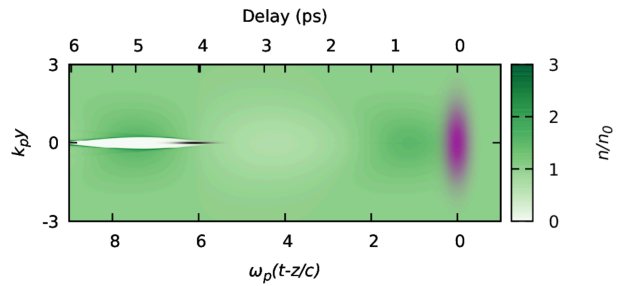
AWAKE Run 2c is currently being designed and planned



The mission for Run 2c is bunch quality

→ The main goal is to **accelerate an injected electron bunch** while **preserving its quality**[1]

Witness bunch drives its own blowout



Target specifications of accelerated bunch

Electron bunch	Goal
Energy	4–10 GeV
Energy spread	5–8%
Emittance	2–32 mm mrad
Charge	100–400 pC (100% charge capture)

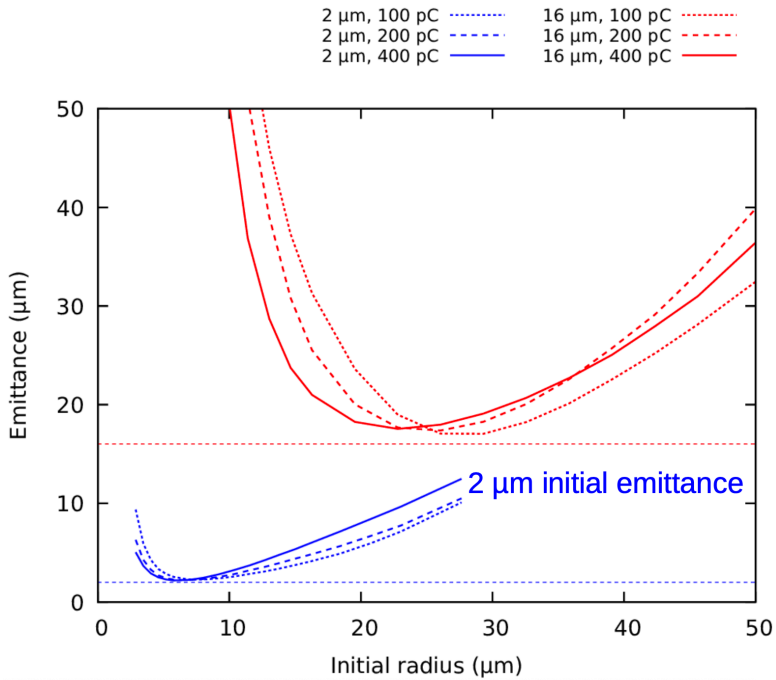
→ matching condition for the bunch radius:

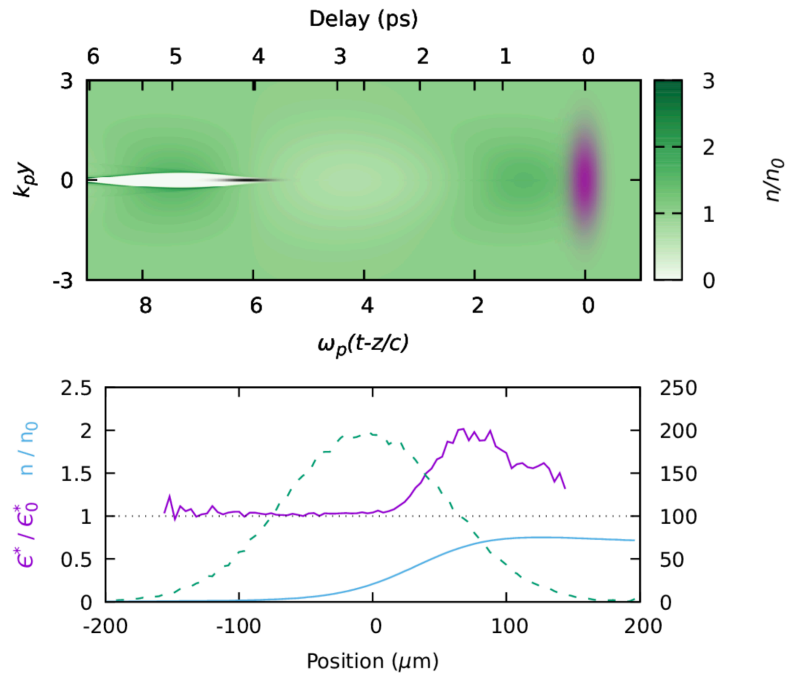
$$\sigma_{\mathrm{ic}} = \left(\frac{2c^2 \varepsilon_{N,x}'^2}{\gamma \omega_p^2} \right)^{1/4}$$

[1] V.K.B. Olsen, E. Adli, and P. Muggli, Phys. Rev. Accel. Beams 21, 011301 (2018)

credit to J. Farmer

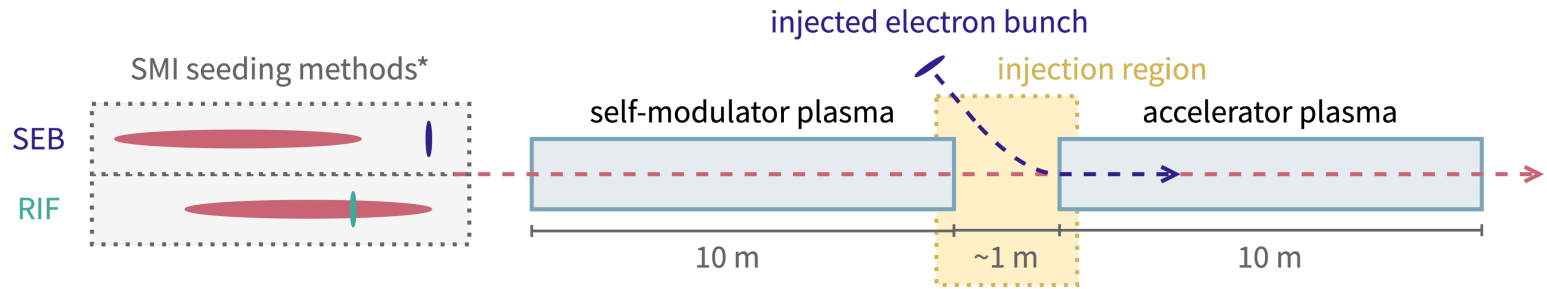
Injection studies have been conducted using a toy model*





* "Injection tolerances and self-matching in a quasilinear wakefield accelerator", J. Farmer *et al.* (2022)

Realistic simulations of Run 2c are challenging



many components

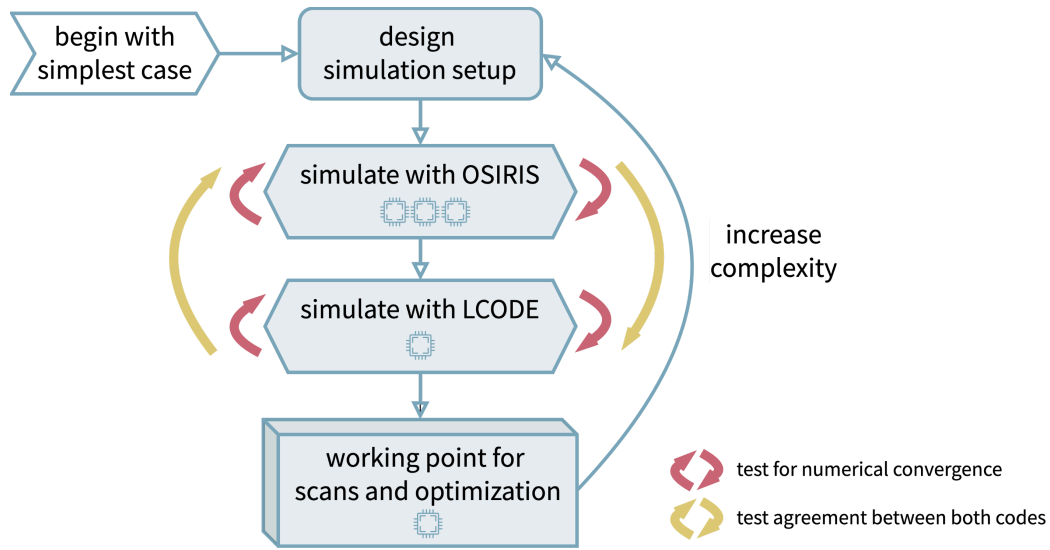
↓

very large parameter space

- two seeding methods
- dynamics of SMI
- plasma density step (self-modulator plasma)
- exit plasma density ramp (accelerator plasma)
- beam loading of witness bunch
- vacuum/vapor gap

Simulation workflow relies on OSIRIS and LCODE

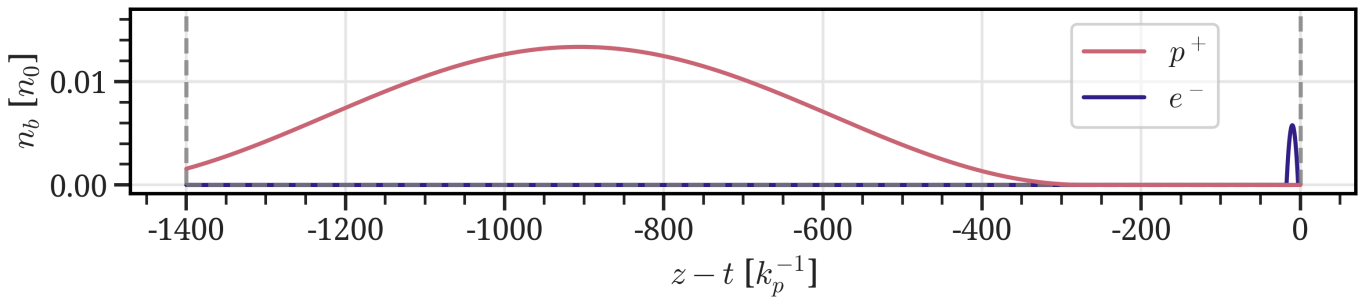
- goal is to use **quasistatic** code LCODE[1] whenever possible
- **benchmark** iteratively against 2D cyl. OSIRIS[2]



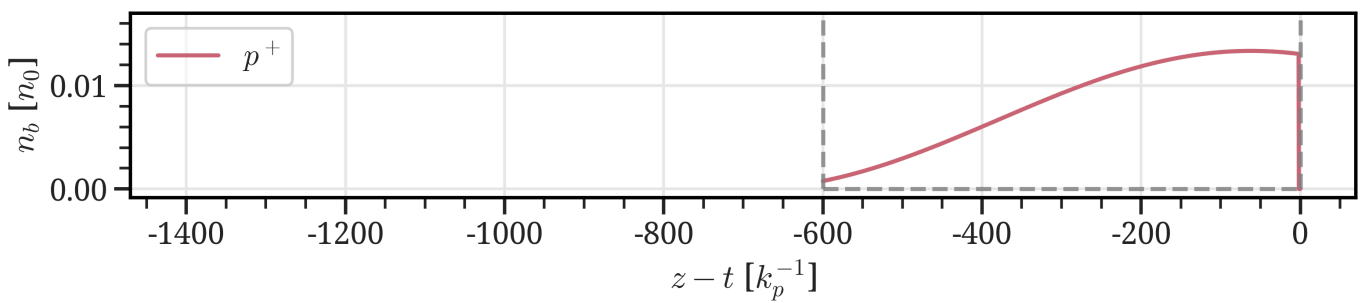
[1] A.P. Sosedkin and K.V. Lotov, Nucl. Instrum. Methods Phys. Res. A, 829, 350-352 (2016)
[2] R.A. Fonseca et al., *Computational Science — ICCS 2002*, vol. 2331, Springer (2002)

Size of simulation window depends on seeding method

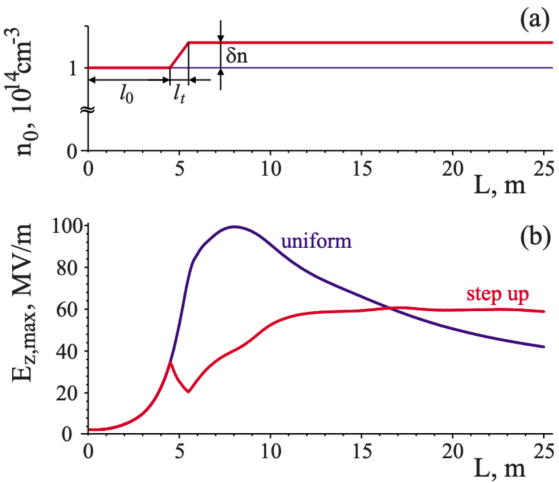
e^--seeded

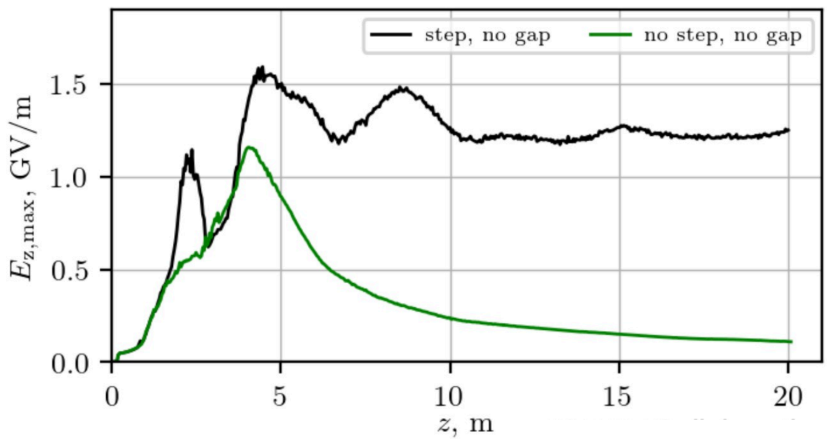


RIF-seeded



Using a density step to optimize the amplitude



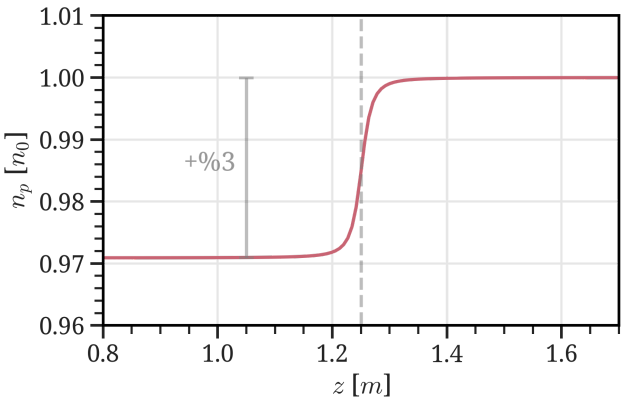


- early step in plasma density profile stabilizes the field after saturation[1,2]

[1] K.V. Lotov, Phys. Plasmas **18**, 024501 (2011)

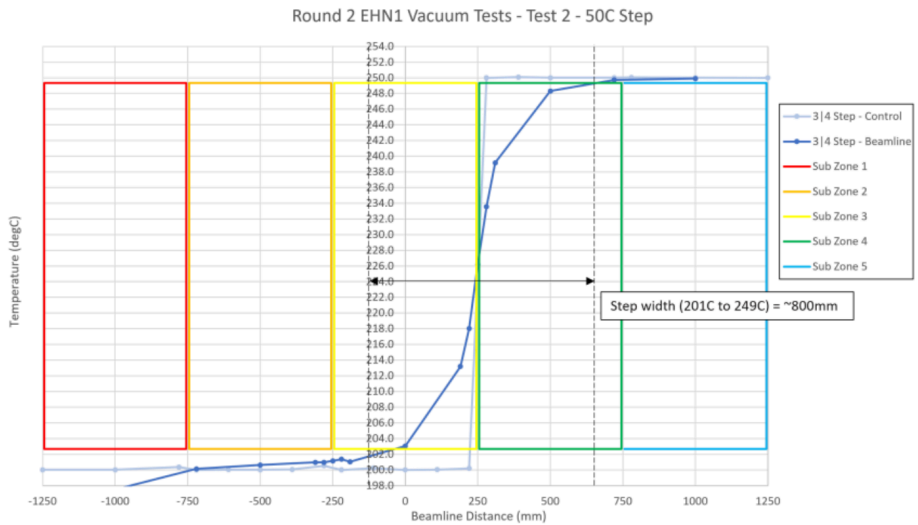
[2] E. Gschwendtner, et al. (AWAKE Collaboration), Symmetry **2022**, 14(8), 1680 (2022)

Using a realistic plasma density profile



Highlight of density step

→ width of step profile*: 25.56 cm**

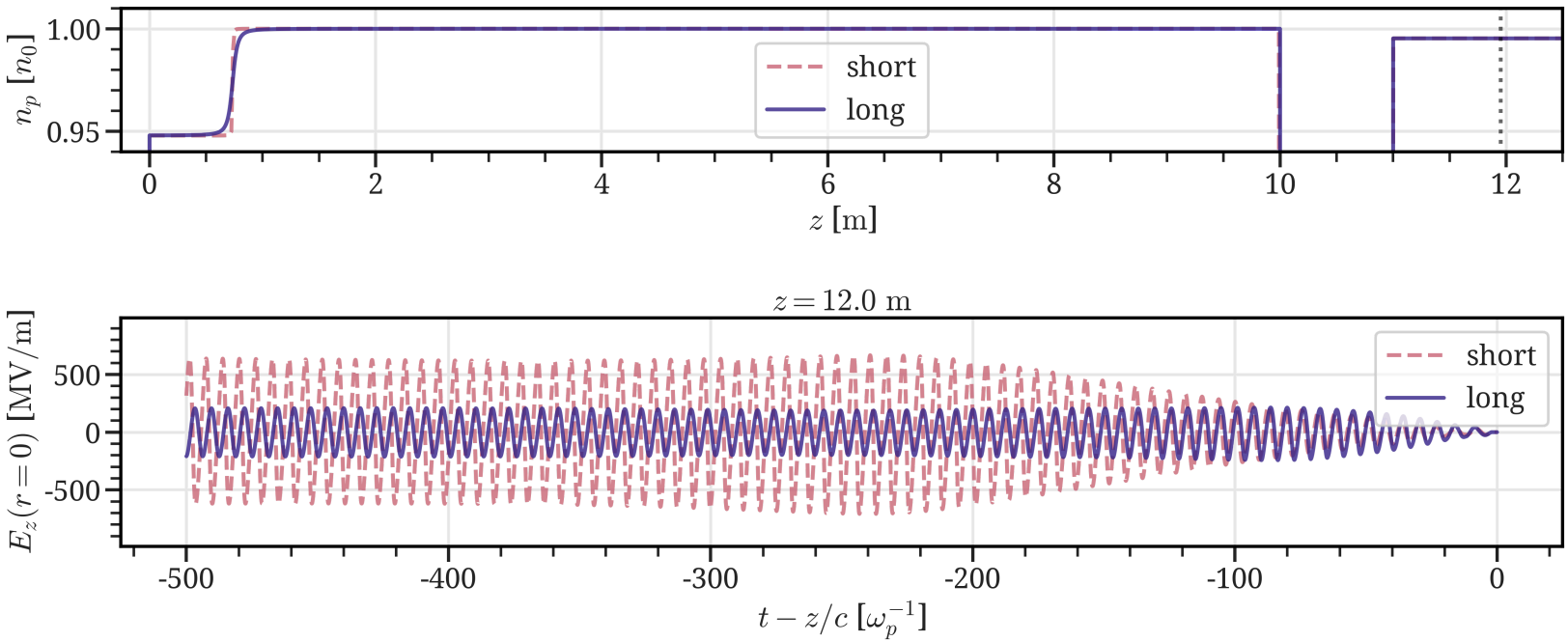


*Measurement of density step***

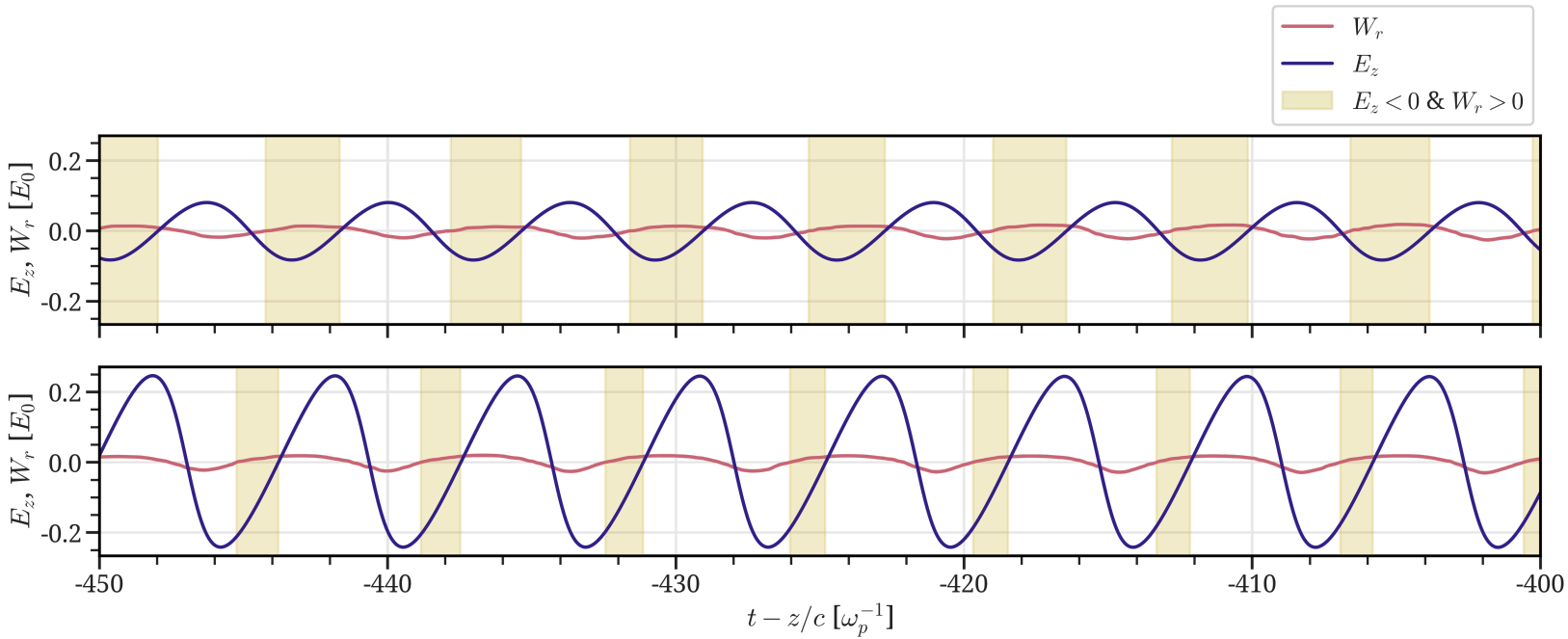
* G. Plyushchev, et al., J. Phys. D: Appl. Phys. **51**, 025203 (2018)

** Interpolated between 10 cm for +1% and 80 cm for +10%, as suggested by measurements

Length of the density step impacts the wakefields



Suitable injection is more difficult at higher amplitudes



- focusing and accelerating regions for electrons shrink as the wakefields become more non-linear

ozzy: a Python package ~~no one asked for~~

The dream:

- developed to **simplify data handling** for particle-in-cell (PIC) simulations
- enables easy analysis and visualization of **large datasets** across **multiple codes**
- built on `xarray`, but is **flexible**: use your existing worklow (e.g. with `numpy`, `pandas`) whenever convenient
- wants to be a healthy **open-source** project when it grows up

Start here!

- Documentation
- GitHub repository

tips? feedback? collaboration?

→ mariana.moreira@cern.ch

Conclusion

- **realistic and complete** simulations of the AWAKE Run 2c setup are challenging, e.g. due to:
  - several components with their own physical effects
  - large range of spatiotemporal scales
  - large parameter space to cover
- current approach is based on iterative benchmarking between slower, "assumption-less" codes and faster, quasistatic codes to perform most of the work
- work is ongoing to tackle the new challenges revealed by more and more realistic simulations

Conclusion

- **realistic and complete** simulations of the AWAKE Run 2c setup are challenging, e.g. due to:
  - several components with their own physical effects
  - large range of spatiotemporal scales
  - large parameter space to cover
- current approach is based on iterative benchmarking between slower, "assumption-less" codes and faster, quasistatic codes to perform most of the work
- work is ongoing to tackle the new challenges revealed by more and more realistic simulations

Thank you!