Dual grid PIC solver in FASTION

L. Mether, G.Rumolo

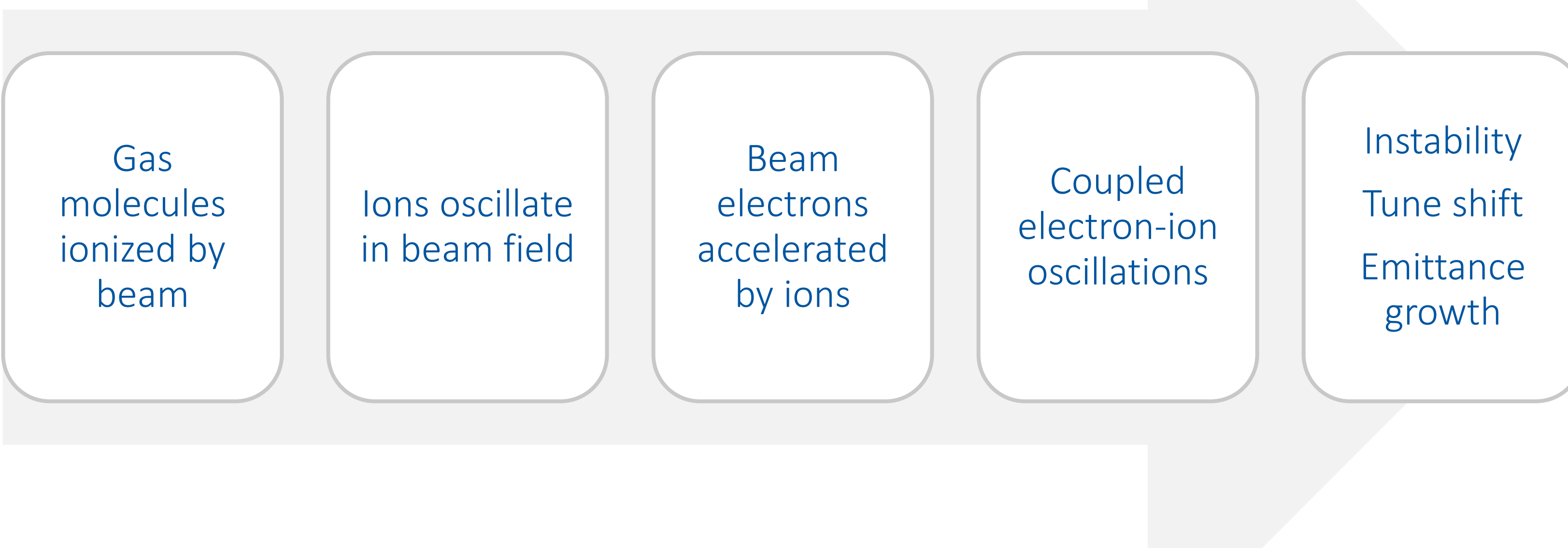
Thanks to: Kevin Li, Gianni Iadraola

12/16/2014

Outline

- **Introduction**
  - Fast beam-ion instability
- **FASTION**
  - Simulation procedure
  - FFT PIC solver in FASTION
- **Dual grid PIC solver**
  - Motivation
  - Implementation
  - Performance

Fast beam-ion instability



- **In circular accelerator without clearing gap, ion density builds up over several turns**
  - Conventional beam-ion instability

- **For bunch train followed by gap, ions build up only during one train passage**
  - Fast beam-ion instability

FASTION

- **Simulation code developed at CERN, by Giovanni, for studying FBII in CLIC structures**
  - Used for scanning the effect of vacuum pressure and composition on instability
  - Previously applied to linear structures: Main linac, transfer line, BDS
  - Optimization for rings on-going
  - Code is still in “development“ phase for rings, systematic scans with realistic vacuum pressures and compositions have not yet been done

- **2D particle-in-cell (PIC) multi-bunch tracking code**
  - Strong-strong code: both beam and ions represented by macro-particles
  - Based on old version of Headtail e-cloud
  - Written in C

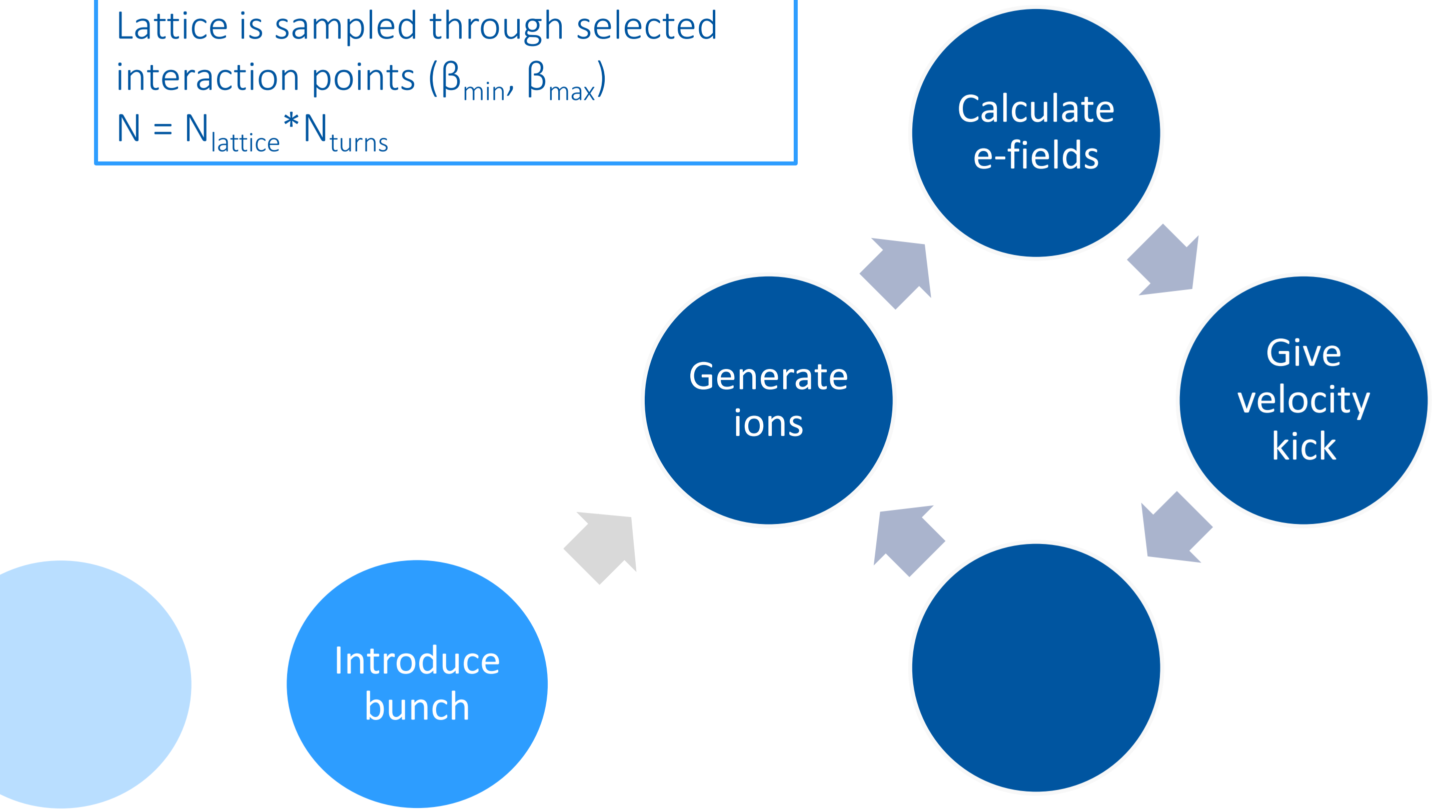
  - Available on GitHub
    https://github.com/lmether/FASTION
    - Currently only one branch: master

FASTION simulation outline

An electron bunch train is tracked through the lattice

In every interaction point, the beam is passed, bunch by bunch

Lattice is sampled through selected interaction points (β_{min}, β_{max})
$N = N_{lattice} * N_{turns}$

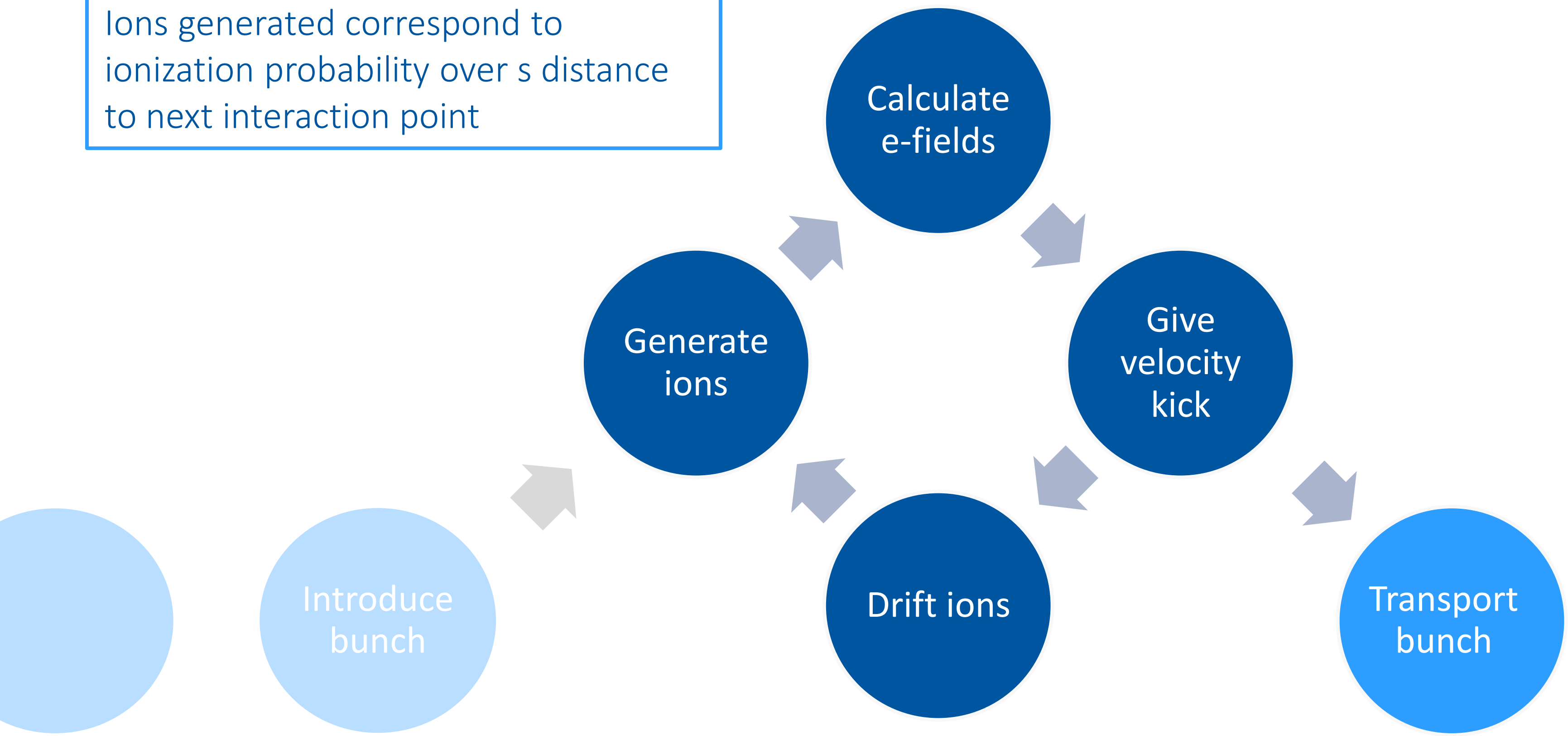


FASTION simulation outline

An electron bunch train is tracked through the lattice

In every interaction point, the beam is passed, bunch by bunch

Ions generated correspond to ionization probability over s distance to next interaction point

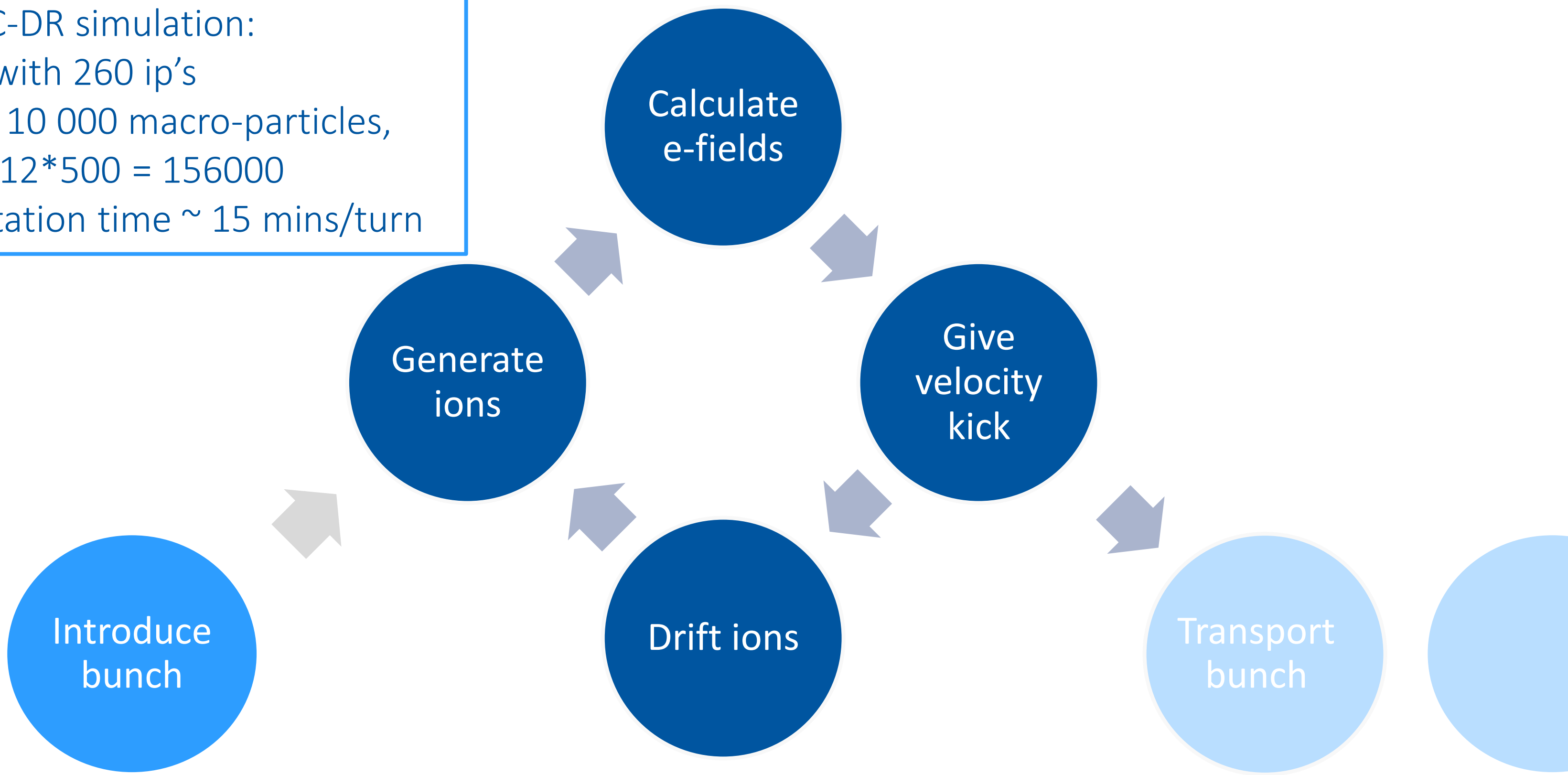


FASTION simulation outline

An electron bunch train is tracked through the lattice

In every interaction point, the beam is passed, bunch by bunch

Typical CLIC-DR simulation:
- Lattice with 260 ip's
- Beam = 10 000 macro-particles, ions = 312*500 = 156000
- Computation time ~ 15 mins/turn



FFT PIC solvers

- There are (at least) two types of FFT PIC solvers

- Direct matrix solver
  - Discretize the Poisson equation and rewrite as a matrix equation M*ϕ + ϕ*M = P
  - M constant valued matrix, can be factorized M = Q Λ Q^{-1} with Λ diagonal

    -> Λ (Q^{-1} ϕ Q) + (Q^{-1} ϕ Q) Λ = (Q^{-1} P Q), which can be solved element-wise for (Q^{-1} ϕ Q)
  - Q = Q^{-1} related to FFT -> matrix multiplications Q^{-1} P Q etc can be done by FFT

- Green's function solver
  - Discretize the solution of the Poisson equation in terms of the Green's function

$$\varphi_{i,j} = \Delta_x \Delta_y \Sigma_i' \Sigma_j' \rho_{i',j'} G_{i-i',j-j'}$$

  - Calculate convolution using FFT

$$\varphi_{i,j} = \Delta_x \Delta_y F^{-1}_{i,j} \left(F(\rho) F(G) \right)$$

  - This is the one used in FASTION

Green's function PIC solver

- Free space Green's function in 2D:

$$G(x,y) = \ln(x^2 + y^2)/4\pi\epsilon_0$$

- Use integrated Green's function to approximate contribution from entire cell

$$\begin{aligned} Fi,j = {} & F(x_i + \Delta x, y_j + \Delta y) - F(x_i + \Delta x, y_j - \Delta y) \\ & - F(x_i - \Delta x, y_j + \Delta y) + F(x_i + \Delta x, y_j + \Delta y) \end{aligned}$$

$$F(x,y) = \frac{1}{4\pi\epsilon_0}\left[xy\ln(x^2+y^2) + x^2\arctan(y/x) + y^2\arctan(x/y) - 3xy\right]$$

  - Calculating Green's function costs 3 trigonometric functions + 1 FFT
  - Note: $F_{i,j}$ is not Green's function for cell i,j but distance i*Δx, j*Δy

FASTION PIC routine

- **FASTION PIC solver**
  - Green's function solver with open boundary (free space)
  - Evaluate Green's function for all grid-spacings
  - For every grid point, calculate potential by summing over charge distribution*Green's function of every other grid point, using FFT

- **Separate treatment of ions and beam**
  - Ion and beam charges are stored separately but using same grid (as real and imaginary parts of arrays for FFT)
  - Potential for ions is calculated using only beam distribution and vice versa
    - No ion-ion, or electron-electron interactions
  - Velocity kick is applied according to electric field induced by opposite particles

Outline

- Introduction
  - Fast beam-ion instability
- FASTION
  - Simulation procedure
  - FFT PIC solver in FASTION
- **Dual grid PIC solver**
  - Motivation
  - Implementation
  - Performance

Resolving beam & tracking ions

- **Simultaneously resolving the beam and tracking ions is computationally challenging**
  - CLIC beam is very small, σ = 2-4 μm in vertical direction
  - Ions can oscillate with very large amplitudes > 1mm

    -> extent of ion distribution up to 1000 * beam size

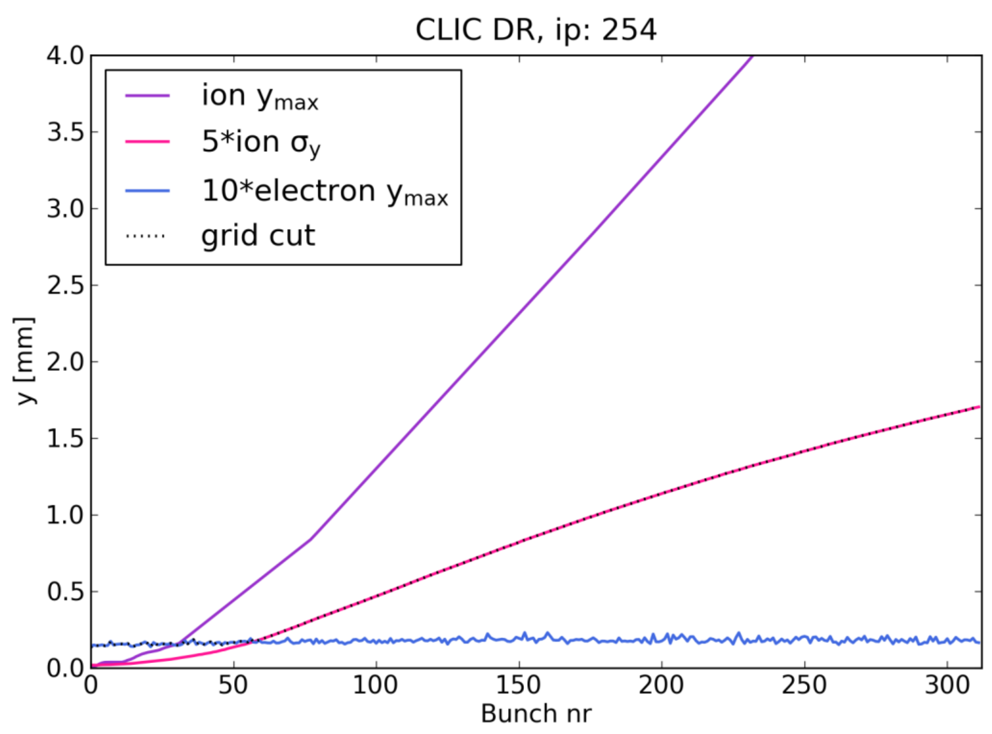
  - Good beam resolution necessary, but also want to track at least most of the ions

- **Current approach: rescale grid during simulations**
  - Check physical dimensions: max electron coordinate, max ion coordinate, etc.
  - Rescale physical size of grid-cells to match physical dimensions at **every bunch passage**
    - Recalculate Green's functions
  - Not possible to track all ions
    - Cut grid at whichever is larger: 10*electron r_{max} or $5*\sigma_{ion}$
  - Ions outside of grid
    - No longer kicked, but drift with constant velocity
    - Not taken into account when calculating kick on beam

Resolving beam & tracking ions

- Simultaneously resolving the beam and tracking ions is computationally challenging

Limits for cutting grid



- Pre

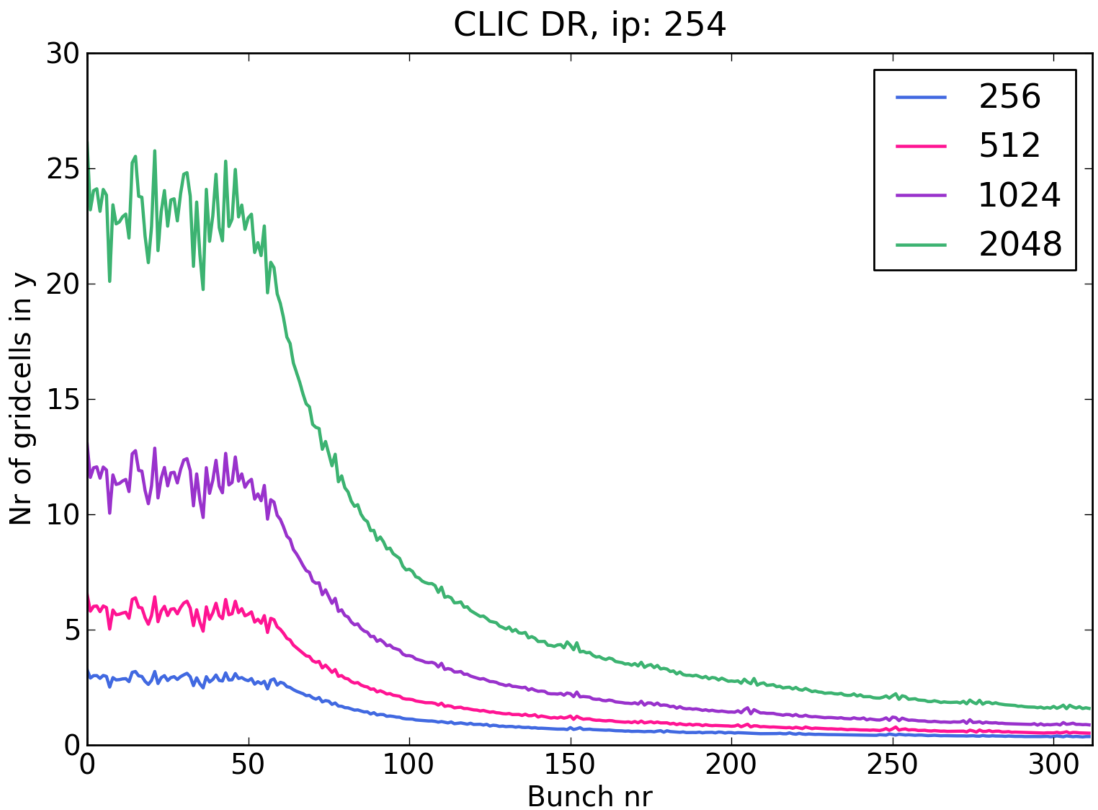
Problems with current approach

- **As σ_{ion} grows beam resolution becomes very poor**
  - Large grid required for convergence, but even then resolution eventually poor
    - In fact, no simulations made for the CLIC-DR until this point were properly converged, and even less for the CLIC Main Linac, where past simulations completely missed the instability

Problems with current approach

- As σ_{ion} grows beam resolution becomes very poor
  - ○ bility

Beam resolution with varying grid size



Problems with current approach

- **As σ_{ion} grows beam resolution becomes very poor**
  - Large grid required for convergence, but even then resolution eventually poor
    - In fact, no simulations made for the CLIC-DR until this point were properly converged, and even less for the CLIC Main Linac, where past simulations completely missed the instability

- **Cutting out ions leads to incorrect trajectories**
  - Ions outside grid give unphysical contribution to σ_{ion}, making resolution even worse

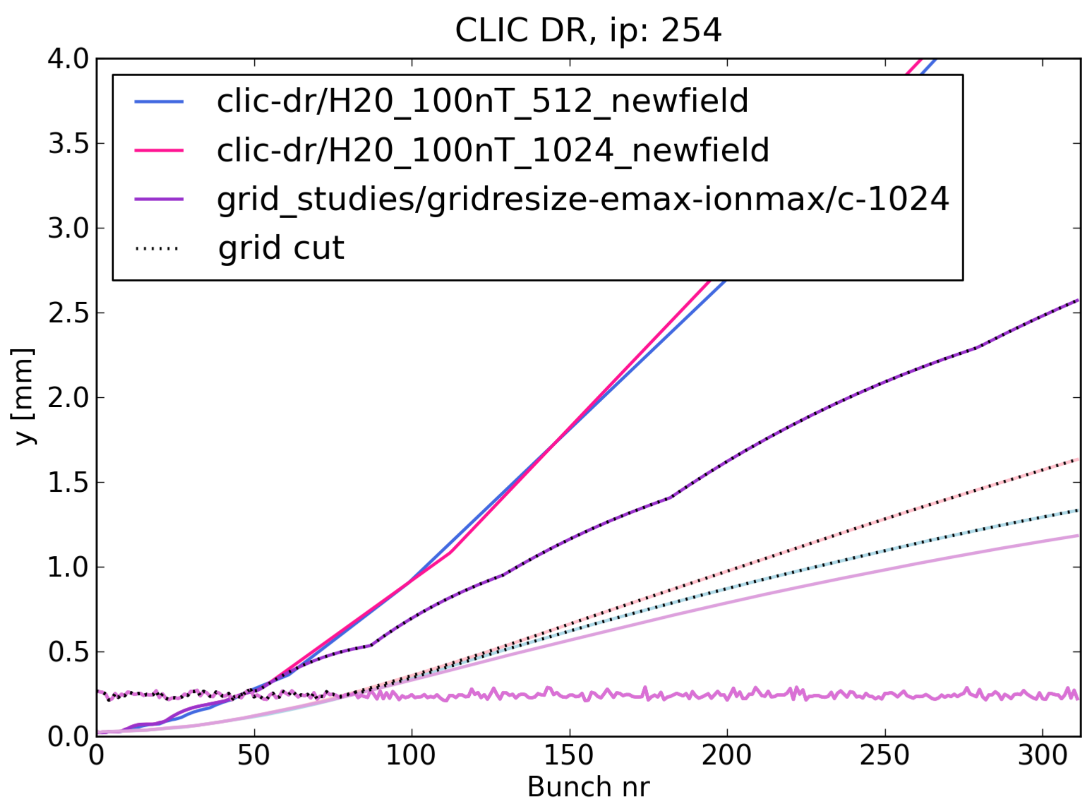
… but tracking all ions is also unpractical

  - Ion trajectories "correct", but grid domain grows even faster, giving ever worse resolution
  - Or requires an unfeasibly large grid

Problems with current approach

- As σ_{ion} grows beam resolution becomes very poor
  - ○ … bility
- Cu…
  - ○
  - …
  - ○ … on
  - ○

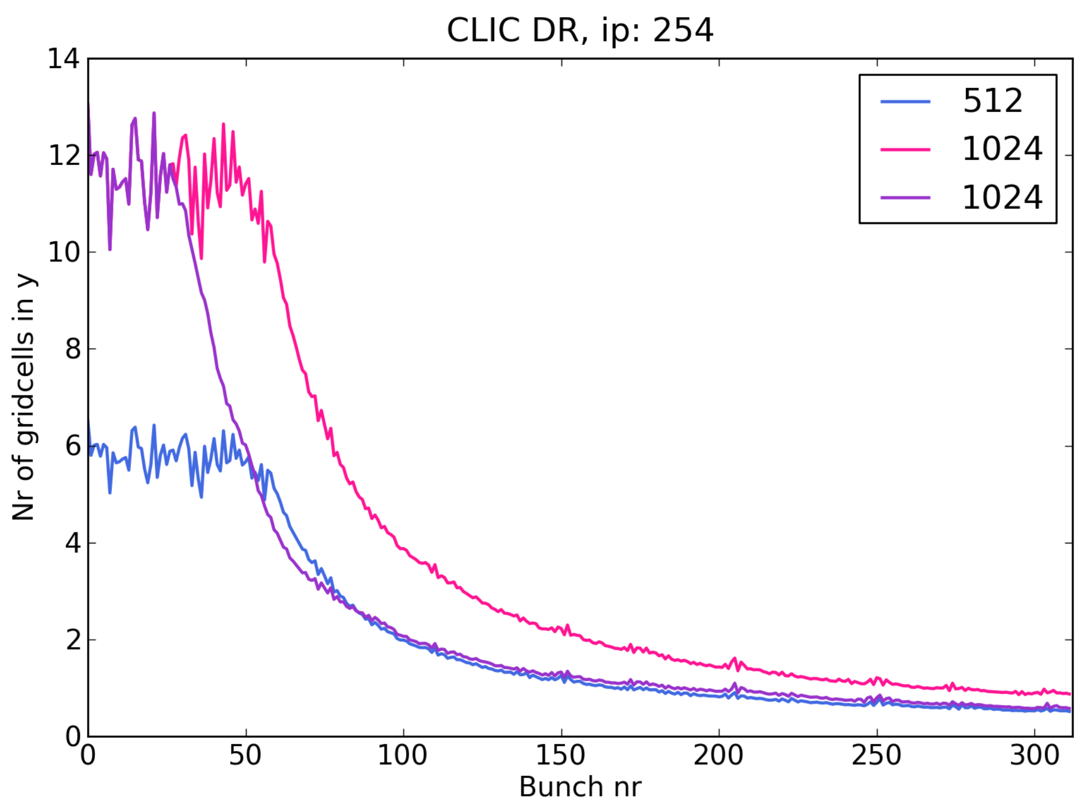
Grid cut limits



Problems with current approach

- As σ_{ion} grows beam resolution becomes very poor
  - ○ … bility
- Cu…
  - ○
  - …
  - ○ … on
  - ○

Grid cut limits



Problems with current approach

- **As σ_{ion} grows beam resolution becomes very poor**
  - Large grid required for convergence, but even then resolution eventually poor
    - In fact, no simulations made for the CLIC-DR until this point were properly converged, and even less for the CLIC Main Linac, where past simulations completely missed the instability

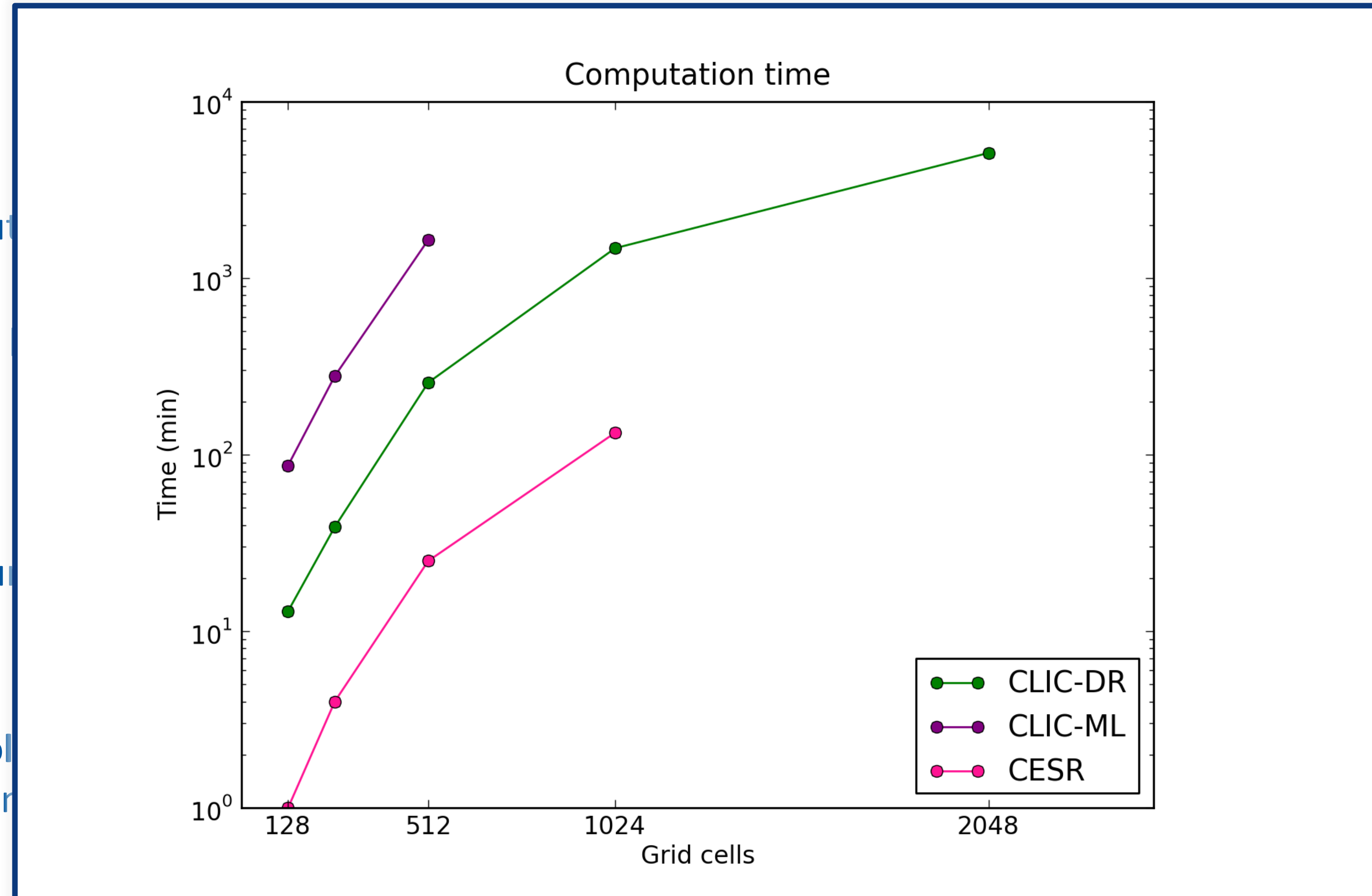
- **Cutting out ions leads to incorrect trajectories**
  - Ions outside grid give unphysical contribution to σ_{ion}, making resolution even worse

  … but tracking all ions is also unpractical
  - Ion trajectories "correct", but grid domain grows even faster, giving ever worse resolution
  - Or requires an unfeasibly large grid

- **Large grids require long runtimes**
  - Also rescaling grid increases runtime, as Green's function has to be recalculated and FFT'd

Problems with current approach

- As σ_{ion} grows beam resolution becomes very poor
  - ○ … bility
- Cu…
  - ○
  - …
  - ○ … on
  - ○
- Ru…
- **Sol**… m co…



Problems with current approach

- As σ_{ion} grows beam resolution becomes very poor
  - Large grid required for convergence, but even then resolution eventually poor
    - In fact, no simulations made for the CLIC-DR until this point were properly converged, and even less for the CLIC Main Linac, where past simulations completely missed the instability

- Cutting out ions leads to incorrect trajectories
  - Ions outside grid give unphysical contribution to σ_{ion}, making resolution even worse

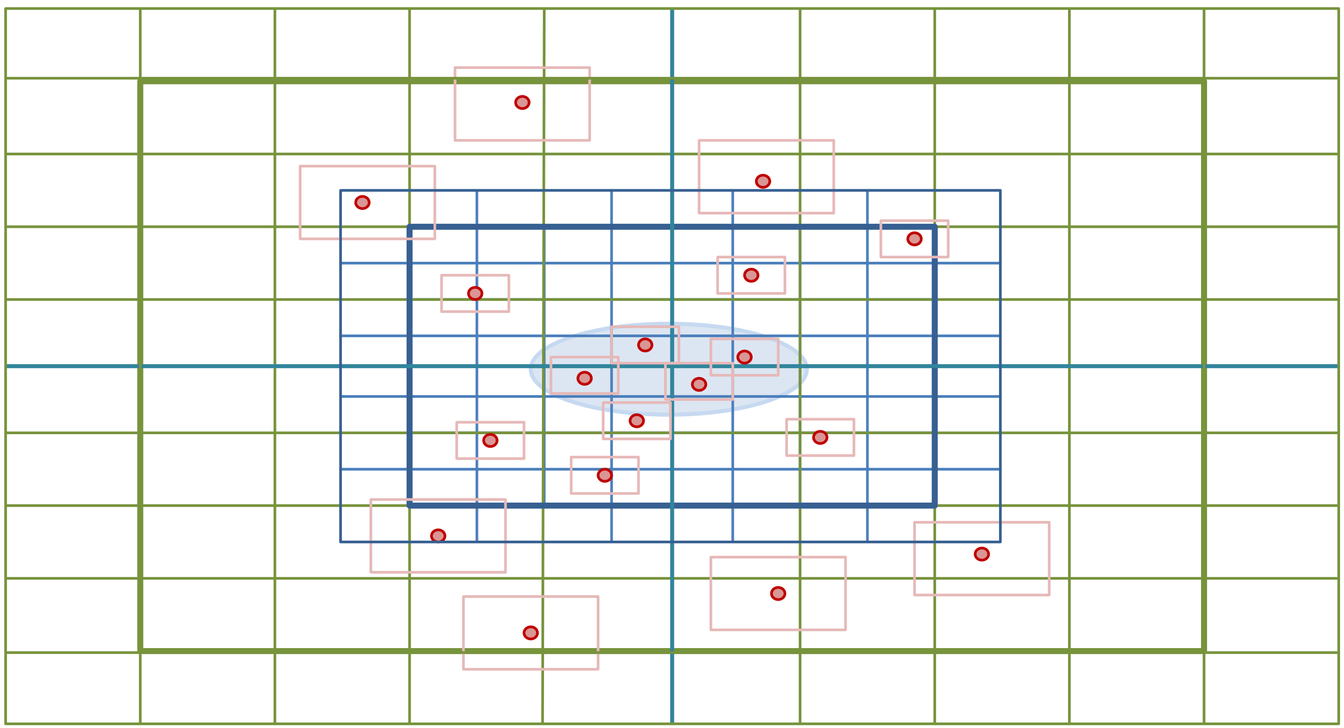
  … but tracking all ions is also unpractical
  - Ion trajectories "correct", but grid domain grows even faster, giving ever worse resolution
  - Or requires an unfeasibly large grid

- Large grids require long runtimes
  - Also rescaling grid increases runtime, as Green's function has to be recalculated and FFT'd

- **Solution**: track all ions to avoid artificial trajectories, but keep resolution around beam constant -> **dual-grid model** (or N-grid)

Dual-grid solver

- **Basic idea: use two overlapping grids**
  - A large, coarse grid
    - Scaled around maximum ion trajectory, to keep track of all ions
  - Inside the coarse grid, a smaller, fine grid centred around beam
    - Kept fixed, to keep beam resolution approximately constant



Implementation

- **Distribution**
  - Particles on inner grid are distributed onto inner grid and separately on outer grid
  - Particles on outer grid are distributed on outer grid only

- **Potential calculated by convoluting charge distributions and Green's functions as normal**
  - Independently for fine and coarse grid
  - Potential on coarse grid contains contribution from all particles
  - Potential on fine grid contains contribution from fine grid particles only

- **To calculate electric field kick**
  - For particle on outer grid, calculate as normal with outer grid potential
  - For particles on inner grid
    - Calculate separately e-field and kick due to fine grid particles, using fine grid
    - Calculate separately e-field and kick due to coarse grid particles on coarse grid
    - Sum the two contributions

Implementation details

- **In principle two coarse grids needed**
  - To separate fine and coarse particle contribution
  - Fine grid is designed to contain all electrons - > no electrons on coarse grid
  - In FASTION no ion-ion interaction –> not necessary to do distribute fine ions on coarse grid

    -> Only one coarse grid, containing coarse ions and fine electrons needed

Implementation details

- **In principle two coarse grids needed**
  - To separate fine and coarse particle contribution
  - Fine grid is designed to contain all electrons - > no electrons on coarse grid
  - In FASTION no ion-ion interaction –> not necessary to do distribute fine ions on coarse grid

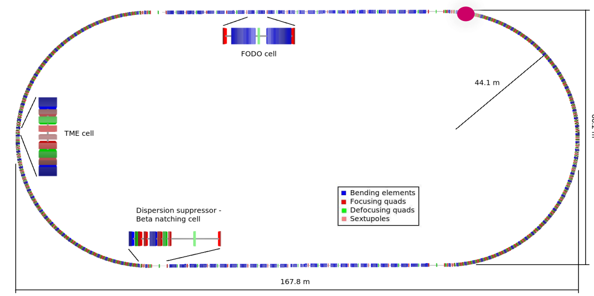
    -> Only one coarse grid, containing coarse ions and fine electrons needed

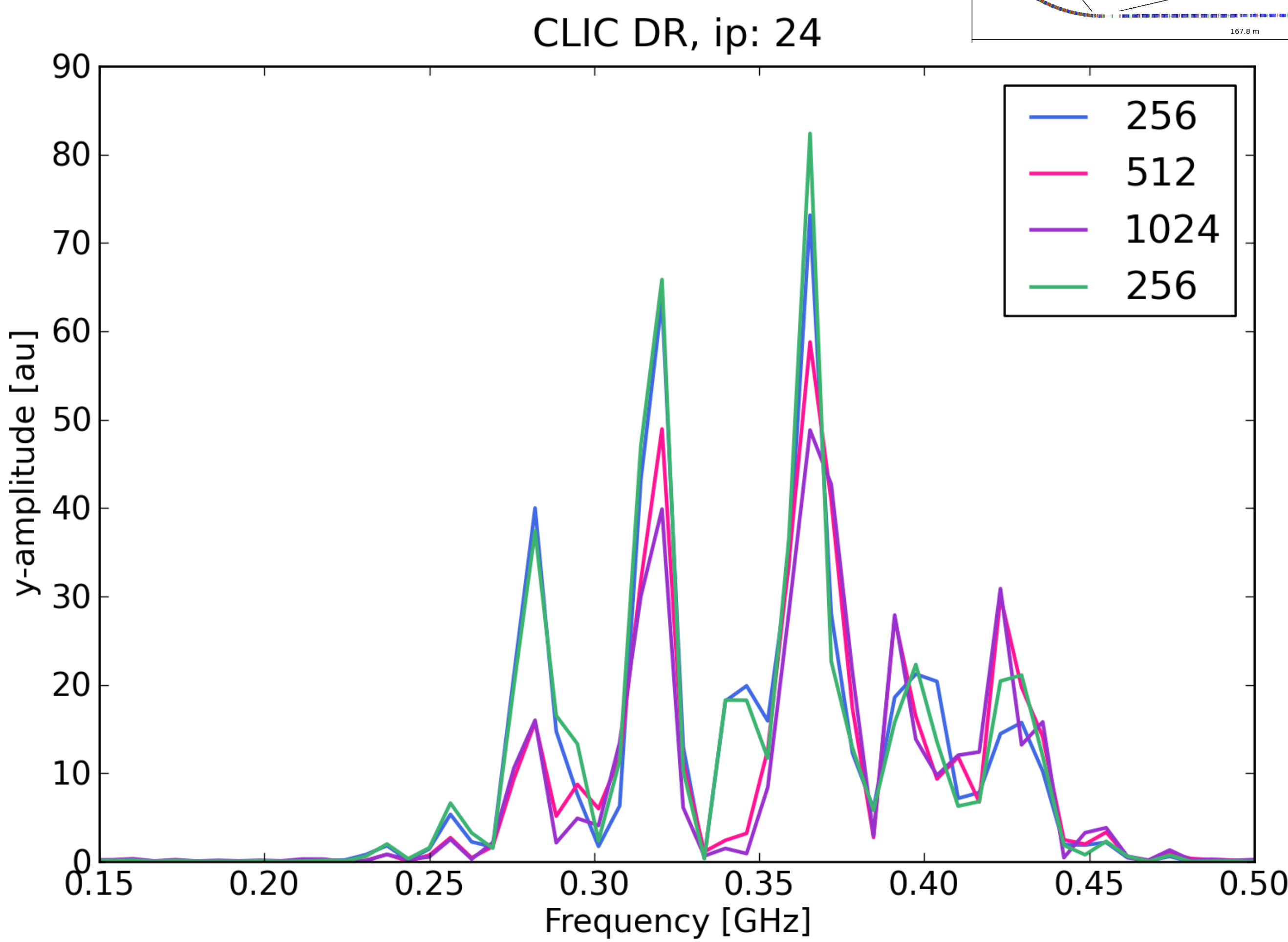
- **Grids must be matched at border**
  - So that the outer edge of the inner grid coincides with a cell edge in the larger grid
  - Start with $\Delta x_{out} = X_{in}/N_{out}$, i.e. the two domains the same
  - When ion reaches beyond inner grid domain, loop over i, scaling $\Delta x_{out} = X_{in}/(N_{out}-i)$ until reach of outer grid, $N_{out}*\Delta x_{out}$, includes ion coordinate
  - In consequence outer grid can be scaled with finer steps in the beginning, getting coarser as grid grows
  - Maximum achievable outer grid size given by $N_{out}*X_{in}$

Let's look at some results!

Oscillation frequency of bunch train

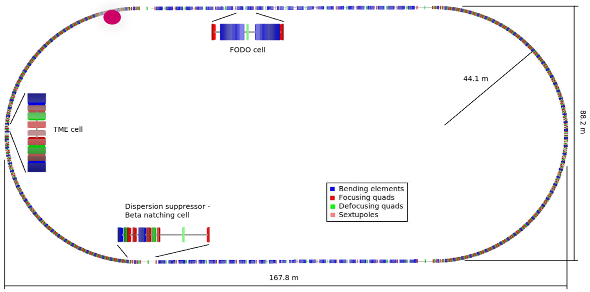
- Various grid sizes compared to 256x256 dual-grid solver

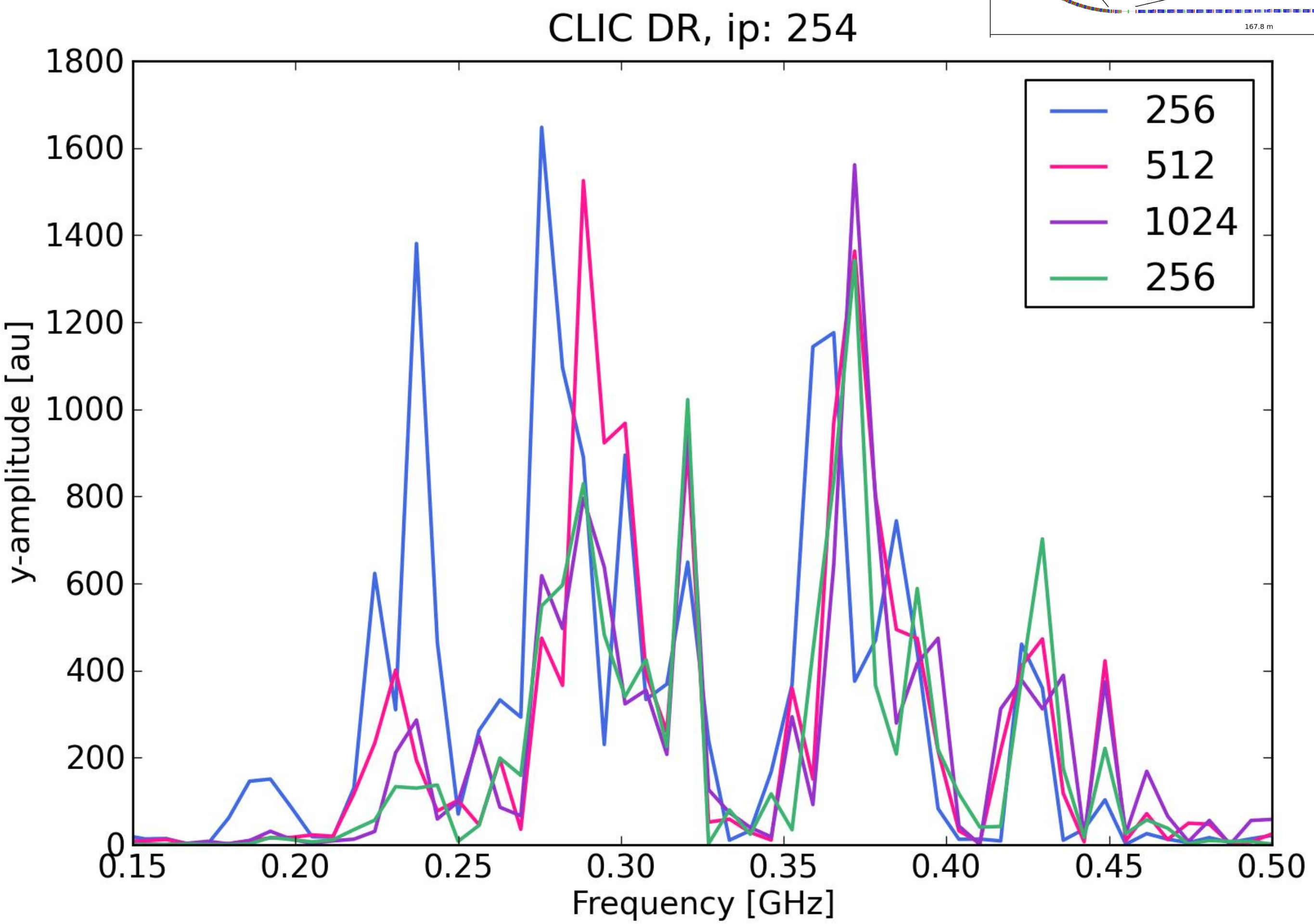




Oscillation frequency of bunch train

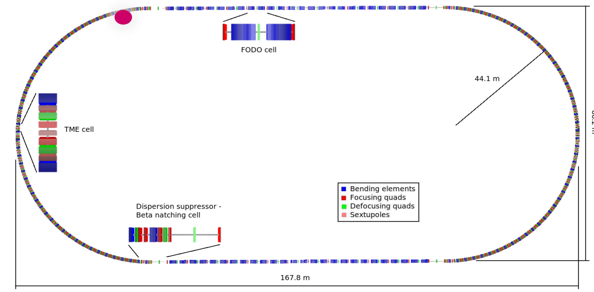
- Various grid sizes compared to 256x256 dual-grid solver

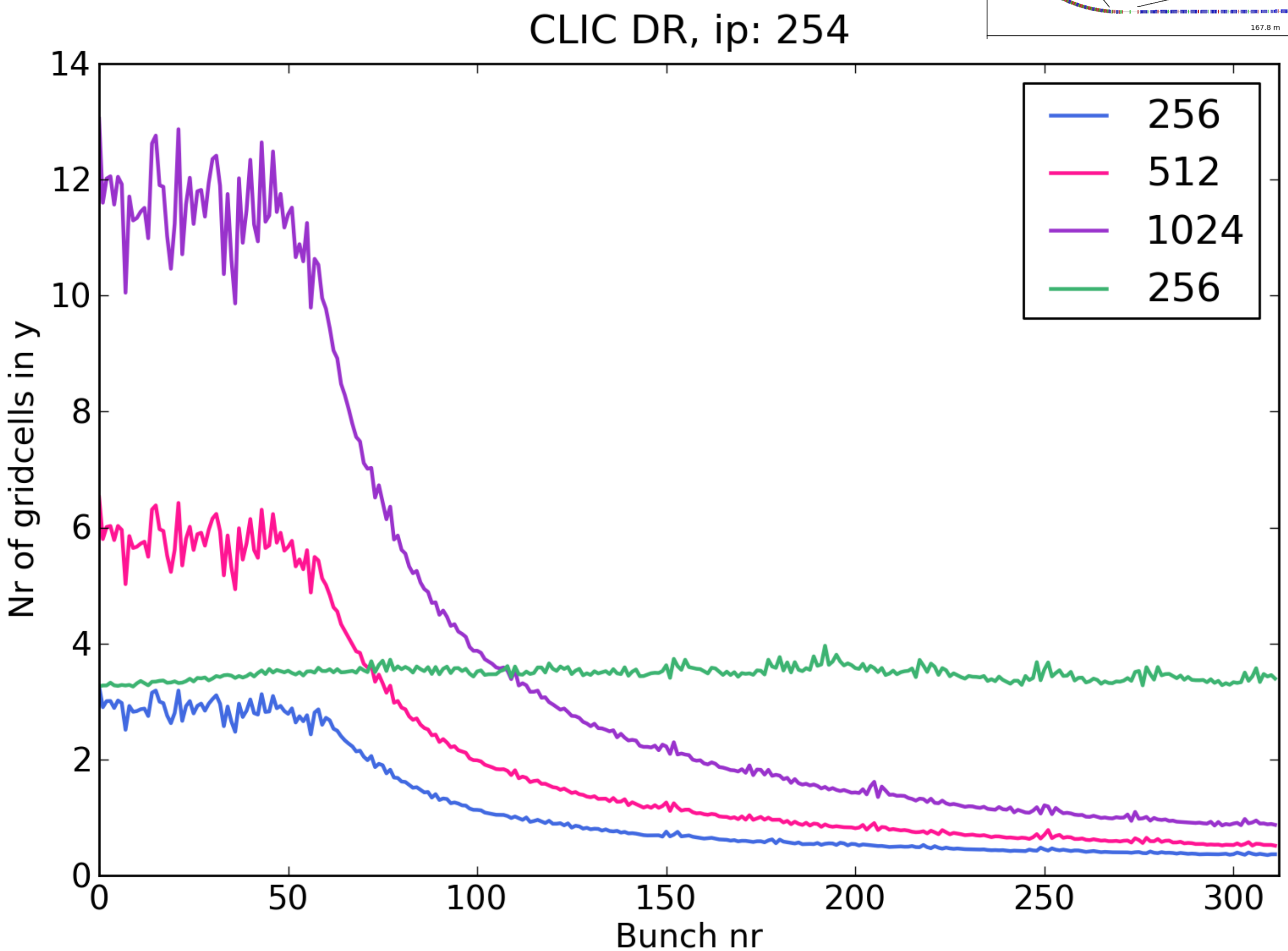




Beam resolution

- Various grid sizes compared to 256x256 dual-grid solver





Accuracy

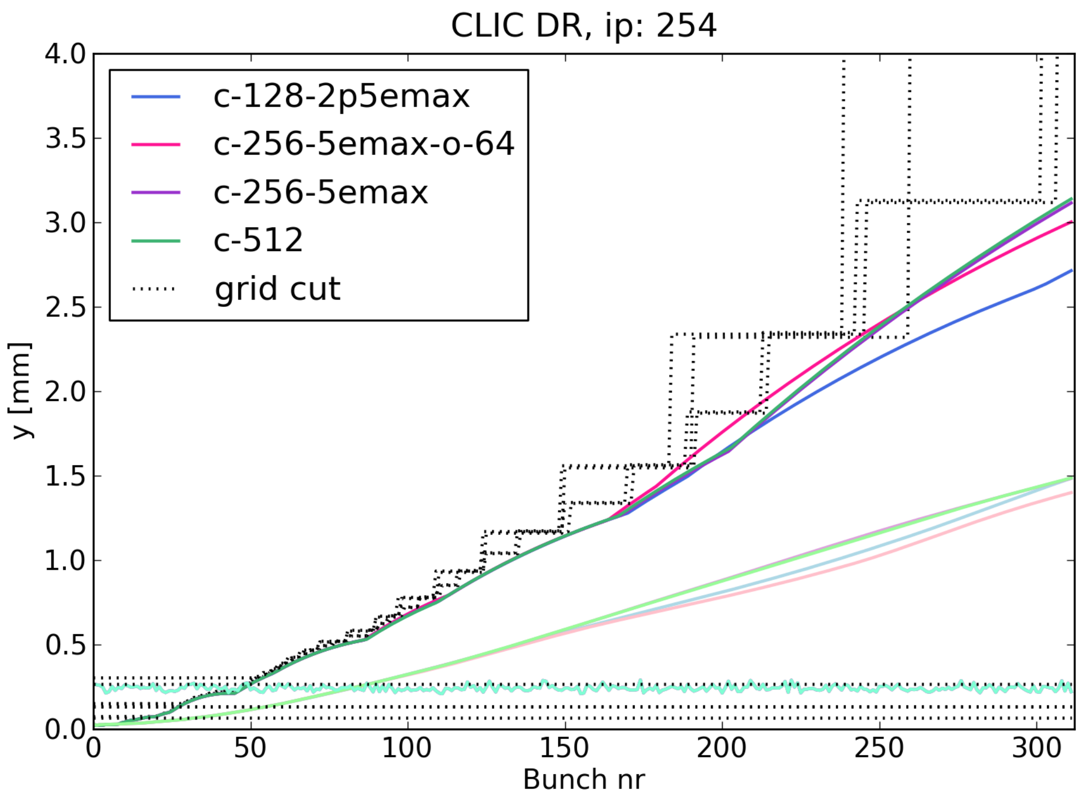
- Summing contributions from grids with different Δx introduces an error
  - For beam, effect of outer ions can be considered a small correction
  - Do not see large effect from this in simulations
  - For ion-ion interactions, especially close to grid boundary, effect should be investigated more carefully

- Compared to single grid
  - A smaller is grid sufficient for convergence compared to single grid, since resolution barely changes during bunch train passage
  - In addition, inner grid can safely be cut closer to beam, since outer grid tracks ions

Accuracy

- Summing contributions from grids with different Δx introduces an error
  - ○
  - ○
- Co
  - ○ ely
  - ○

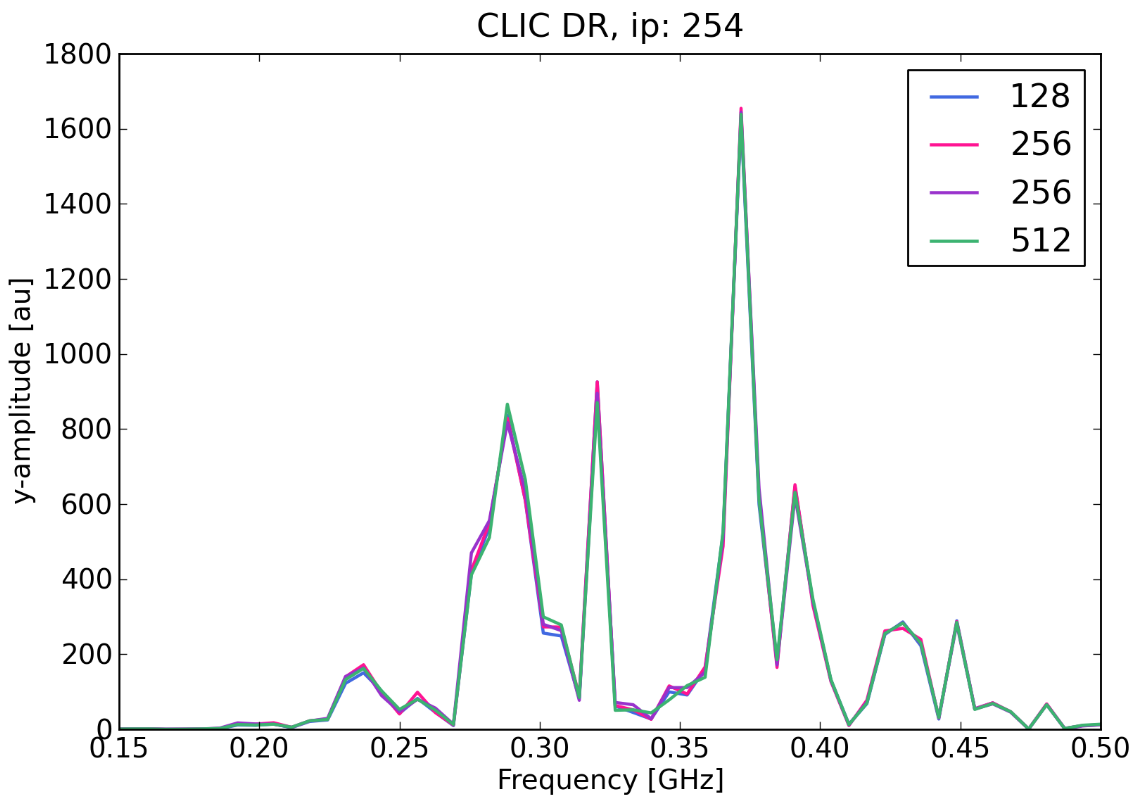
Limits for grids, with same beam resolution, but different outer grid



Accuracy

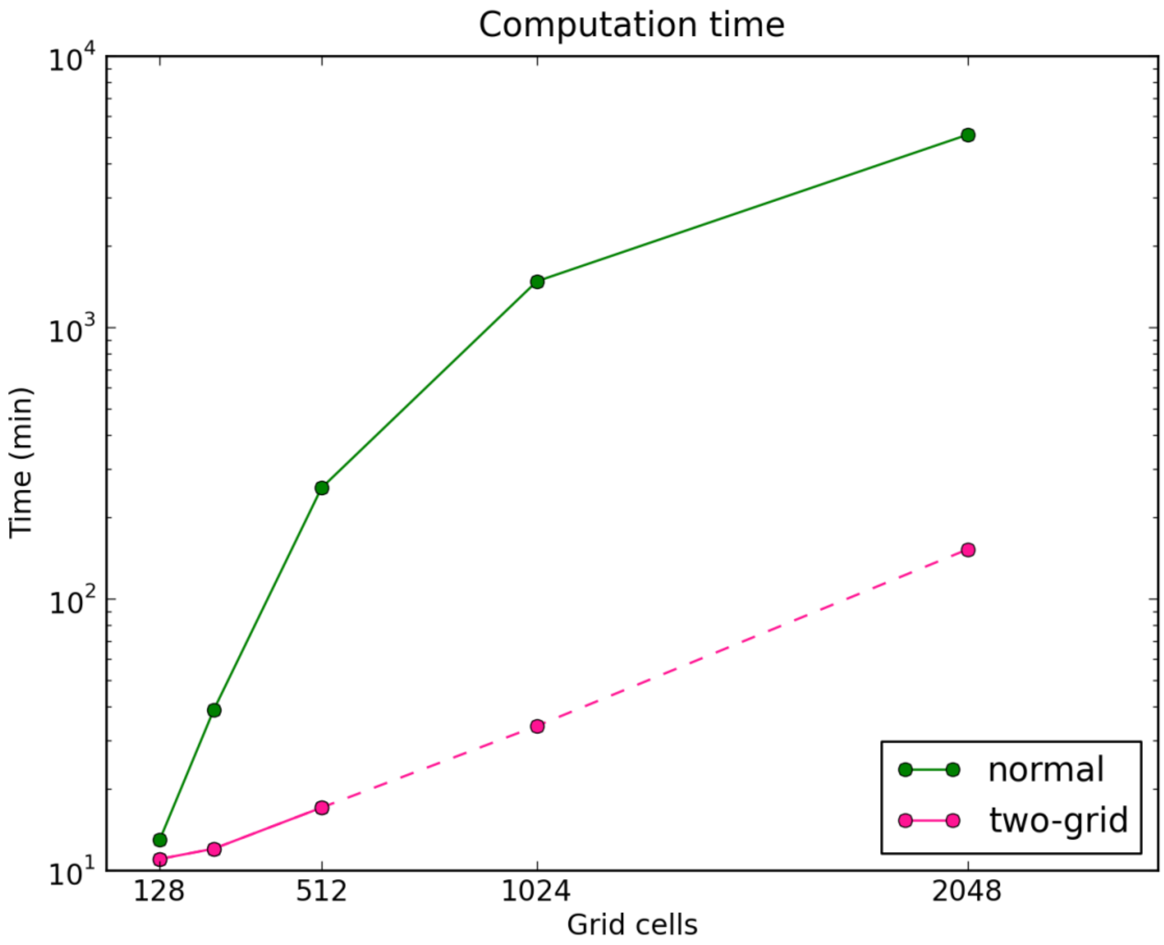
- Summing contributions from grids with different Δx introduces an error
  -
  -
- Co
  - ely
  -

Oscillation frequency for same beam resolution, but different outer grid



Computation time

- Two-grid solver much faster than single grid
  - Comparing the smallest possible converged grid that gives similar resolution as single grid



Summary & Outlook

- **Dual grid was "quick and dirty fix" for FASTION issues**
  - Implementation is very explicit and quite rigid
  - But fast to implement and compatible with existing PIC routine

- **Might be useful also in other cases where the main charge density is concentrated in a small area, with spread out distribution around**
  - In particular, when FFT occupies majority of computation time
  - However, computation time gain dependent also on number of macro-particles
    - Here I used quite low numbers
  - Including interactions between all particles will also decrease computation time gain

Summary & Outlook

- **Dual grid was "quick and dirty fix" for FASTION issues**
  - Implementation is very explicit and quite rigid
  - But fast to implement and compatible with existing PIC routine

- **Might be useful also in other cases where the main charge density is concentrated in a small area, with spread out distribution around**
  - In particular, when FFT occupies majority of computation time
  - However, computation time gain dependent also on number of macro-particles
    - Here I used quite low numbers
  - Including interactions between all particles will also decrease computation time gain

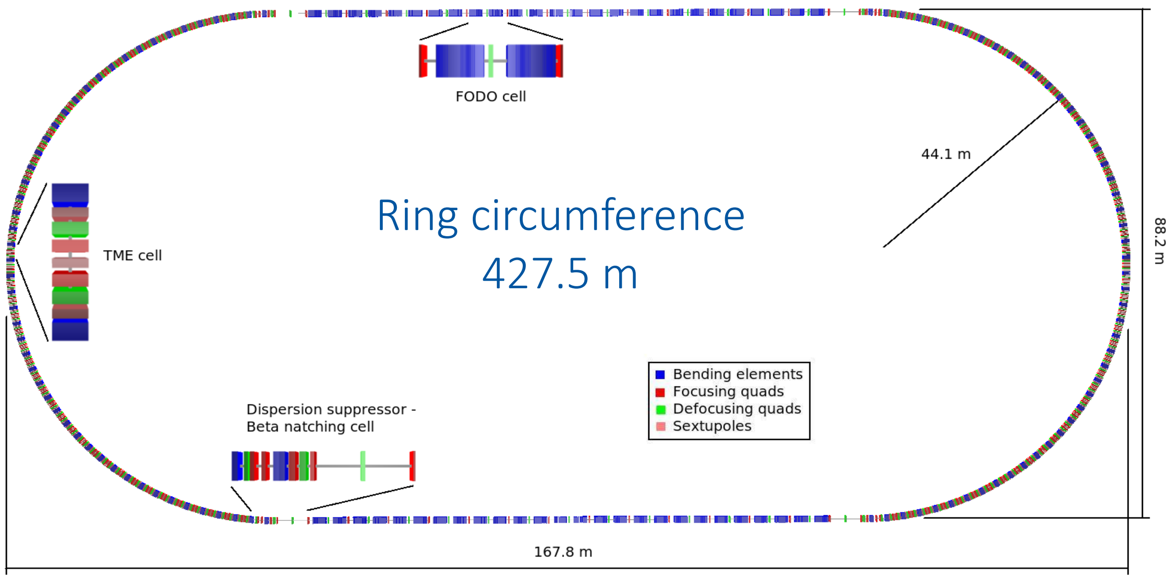
- **Open questions**
  - Effect of boundary crossing, especially for nearby charges
  - Compatibility with boundary conditions

  - Eventually explicit N-grid solver could be replaced by adaptive grid?
  - Would be nice to look also at other grid methods before deciding which one(s) to develop further

Thank You!

www.cern.ch

CLIC e- Damping Ring



Parameters (extraction)	1 GHZ	2 GHz
Beam energy, E_0 [GeV]	2.86	
Emittances, $\varepsilon_{n,x,y}$ [nm]	500, 5	
Transverse tunes, Q_x, Q_y	48.35, 10.40	
Bunch population, N_b	4.1×10^9	
Bunches per train, n_b	156	312
Bunch spacing , T_b [ns]	1	0.5
Bunch length (rms), σ_z [mm]	1.8	1.6

Dual-grid solver

Schematically: $$E_i(p) = -h_{i,in}(p)\frac{\Delta\varphi_{in}(p)}{dx_{i,in}} - h_{i,out}(p)\frac{\Delta\varphi_{out}(p)}{dx_{i,out}}$$