PAUL SCHERRER INSTITUT

PSI

Wir schaffen Wissen – heute für morgen

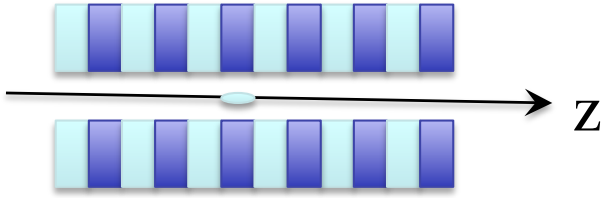
Paul Scherrer Institut

Sven Reiche

Numericla Methods for FEL Simulations (FEL Prize Talk)

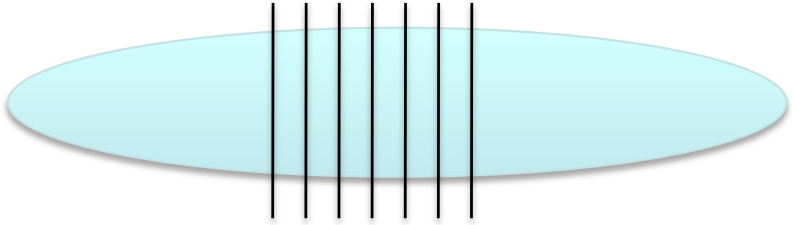
The Modeling Frame

Undulator Field (~200 m)



Longitudinal position is independent variable. Undulator field and focusing become "time-dependent"

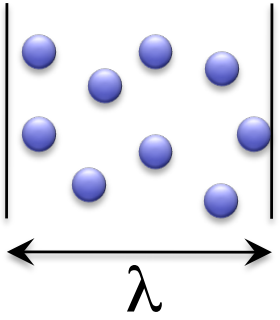
Electron Beam (~50 μm)



Co-moving frame:

$$s = z - c\beta_0 t$$

Electron Slice (~1 Å)



Slice thickness λ defines reference wavelength, which is not necessarily the resonant wavelength. Though both should be close to avoid strong drifts in slice:

$$\beta_0 = \frac{k}{k + k_u}$$

The FEL Equations (period-averaged)

Particle Motion

$$\frac{d}{dz}\theta = \left(k + k_u\right)\beta_z - k \qquad \textit{(Ponderomotive Phase)}$$

$$\frac{d}{dz}\gamma = -k\frac{f_c K}{2\gamma}\left(ue^{i\theta} + cc\right) + \frac{e}{imc^2}\left(\tilde{E}e^{i\theta} - cc\right)$$

$$\frac{d}{dz}\vec{r}_\perp = \frac{\vec{p}_\perp}{\gamma}$$

$$\frac{d}{dz}\vec{p}_\perp = \underline{M}(z)\cdot\vec{r}_\perp$$

Field Equation

Radiation field

$$\left[\nabla_\perp^2 + 2ik\left(\frac{\partial}{\partial z} + \frac{\partial}{c\,\partial t}\right)\right]u = i\frac{e^2\mu_0}{m}\sum_j \frac{f_c K}{\gamma_j}e^{-i\theta_j}$$

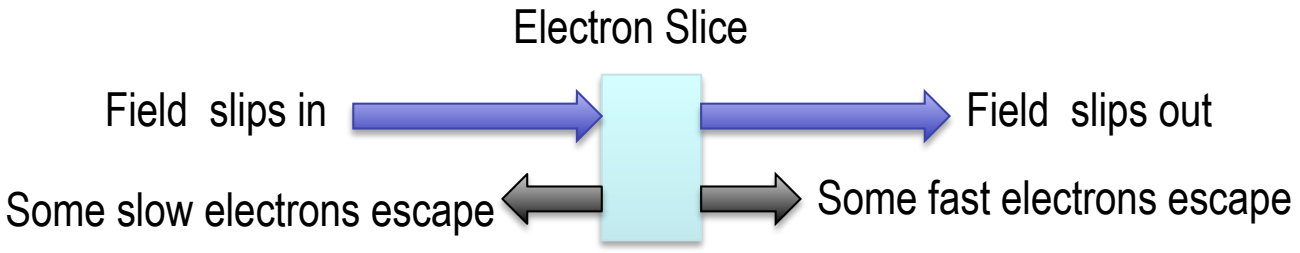
Space charge field

$$\left[\nabla_\perp^2 - \frac{k^2}{\langle\gamma_z\rangle^2}\right]\tilde{E} = i\frac{e}{\varepsilon_0}\frac{k}{\langle\gamma_z\rangle^2}\sum_j \delta\left(\vec{r} - \vec{r}_j\right)e^{-i\theta_j}$$

Core Algorithm – Steady State Simulation

Two step algorithm:

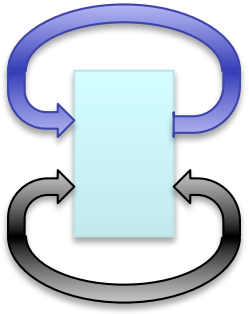
- Advance radiation field (diffraction + emission by electrons)
- Advance electrons (interaction with field and change in ponderomotive phase)



In steady-state simulations:

- Infinite long bunch with the same properties (no time-dependence)
- Zero net flow of field and electrons of any slice
- Simplification: field and particles are fed back into the same slice

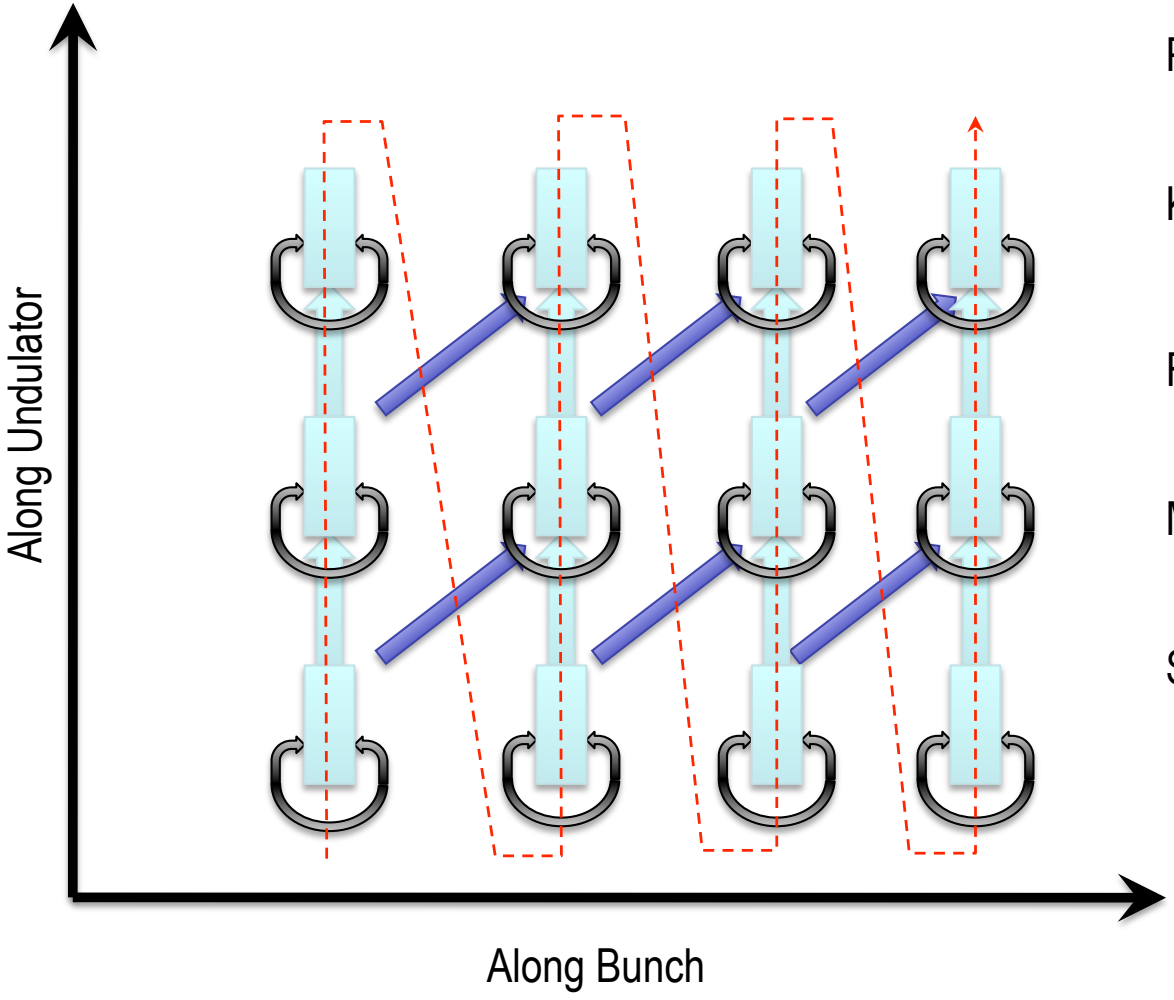
Tracking of only on radiation field and one electron slice



Self-fed Amplifier

Quasi-Time-Dependent Simulations

Requirement: Tracking N_s slices over N_z steps along undulator.
Minimum required slices: $N_{slp} = \lambda/\lambda_u N_z$ (Slippage)
Most Simulations: $N_s > N_{slp}$



Propagate Field to Next Slice

Keep Slice in position

Feed electrons back into slice

Model of chained amplifiers

Simulation can crawl through bunch:

- Inner loop: undulator
- Outer loop: bunch

Pros & Cons of Quasi-Time-dependent Simulations

Pros:

- Very efficient memory demand (1 Electron Slice and N_{slp} Radiation Slices)
- Easy implementation on parallel computers (N_{core} Electron Slices, N_{slp} Radiation Slices)
- Fixed number of macro particles per slice (charge scaling) yields highly efficient performance on parallel computer

Cons:

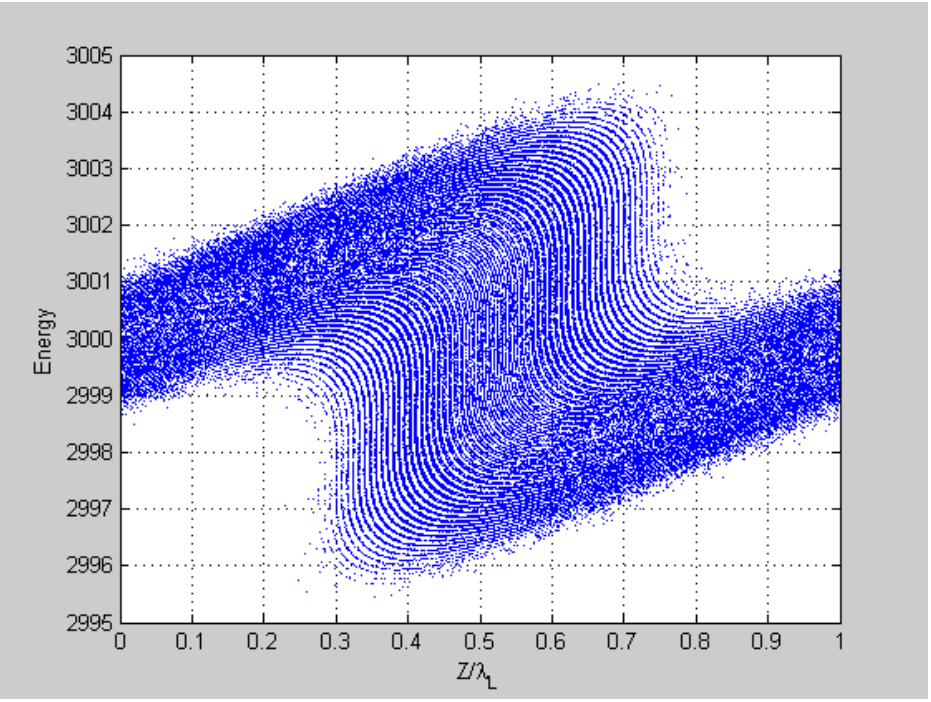
- No exchange between electron slices (Rigid current profile)
- Pre-calculated collective effects (undulator wakefields) instead of self-consistent models
- Crude approximation of chirps (no chirp within a slice)
- Calculation wasted in low current slices

Where it breaks…

Quasi-time-dependent simulations works fell for single-pass SASE FELs

However self-seeding und EEHG seeding have a very strong impact on phase space distribution:

EEHG Simulation for SwissFEL Athos Beamline at 1 nm



Configuration example:
A1 = 5, A2 = 5
R56_1 = -8.636mm
R56_2 = -0.0714mm

→ Electrons are shifted up to 5 microns

→ Mixing of up to 10000 slices

Import/Export with other Programs very problematic

No mixing with quasi time-dependent code alone

Fully time-dependent code would solve the problem

Bunching & Harmonics

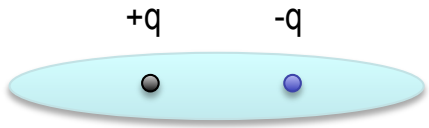
To use less macro particles than electron to be simulated, the preapration of phase space needs special care:

- Remove any random noise from the macro particles (quiet noise)
- Apply a well define error in the longitudinal position of macro particles

Macro particle with internal degree of freedom

Typical Implementation

Beamlets (*Fawley*)
Position varies

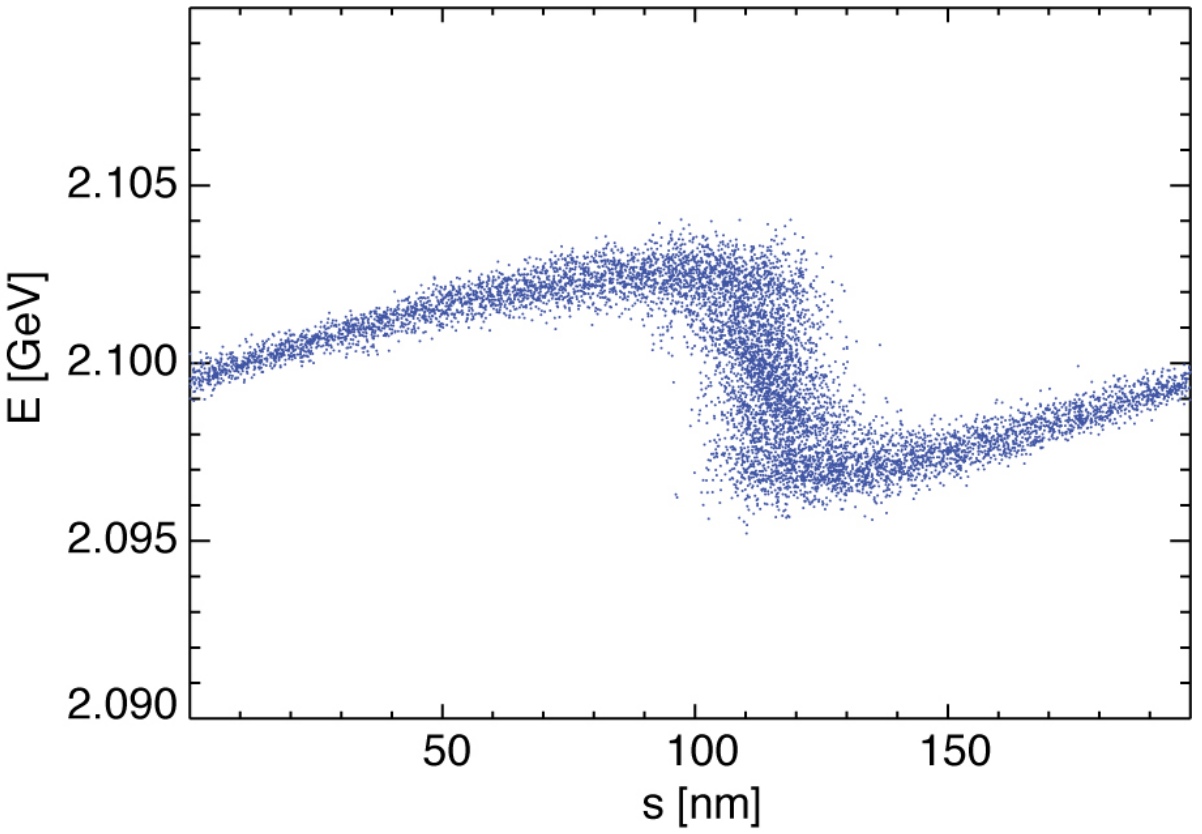


Dipole (*Litvinienko*)
Charge and separation varies

With higher harmonics the number of internal degrees needs to be increase

High Harmonic Conversions

High Gain Harmonic Generation is a strong redistribution of phase space and then “sampled” with a new wavelength.



- Rich harmonic content requires many particles per beamlet
- 90 degree rotation in phase space converts energy distribution into longitudinal position
- Shot noise not correct anymore
- Higher harmonics are probing the form factor of the initial energy distribution. Strong model dependence

Needs large numbers of beamlets and particles per beamlets

Where it might break…

Harmonic conversion:

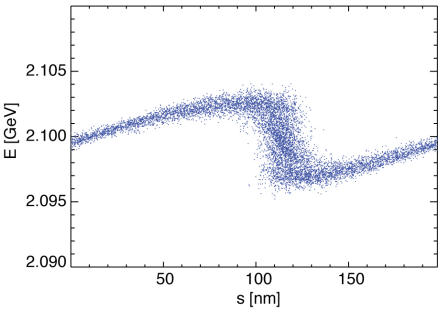
$$\theta \rightarrow n\theta$$

However sampling volume keeps the same because it is almost impossible to break apart beamlets.

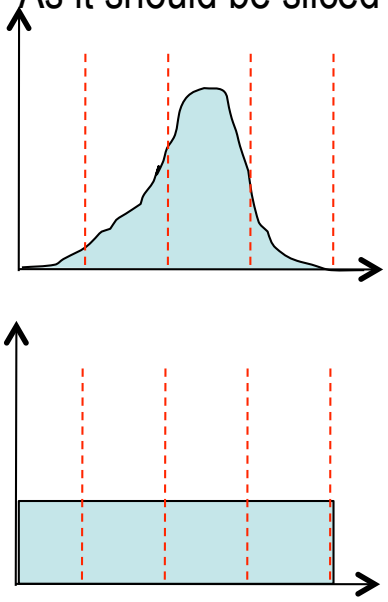
- Sampling rate is reduced by the harmonic number
- Reduced bandwidth of the simulation

Also harmonic conversion implementation averages over multiple wavelength of the new harmoncis

Current profile



As it should be sliced



As it is treated

Difference between sliced and averaged approach:

- Slices with low current
- Reduced gain guiding
- Diffraction and Guoy phase shift
- Non-linear dependence on current

Straight forward slicing would be very welcome.

Are These Problems Relevant ?

First generation of high gain FELs were single-pass SASE FELs, where quasi time-dependent codes worked well.

Current focus on the next generation of FELs is seeding, which includes either:

- Strong manipulation of phase space
- High harmonic conversions

Quasi time-dependent codes can give a (good) approximation but it reminds more of a patchwork calculation than a satisfying, self-consistent model.

The multiple change between codes makes optimization difficult.

Can these problems be solved?

Yes

What has to be done

Avoid “beamlets”, most straight forward by “one-one” simulation (though other algorithm exists with even less)

Full bunch simulation at 1 Å are already close to the electron number

Utilize parallel computer architecture

Already used in code and more commonly accessible

Change loop order: inner loop along bunch, outer loop along undulator

Here works needs to be done, mostly solving the memory problem

Allow for sorting and rebinning of the particles

Should be possible to benefit from other parallel codes

Wakefields and long range space charge

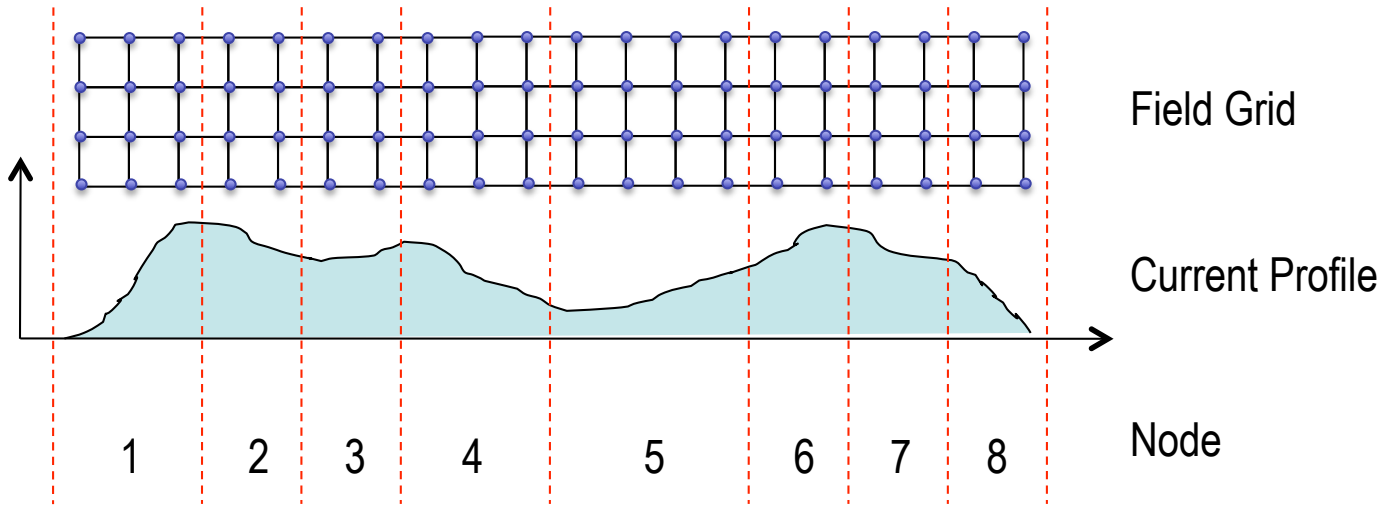
Number Crunching Example

Beam: 200 pC, 20 micron length at 1 Å:

- $1.25 \cdot 10^9$ macro particles → 60 GByte Memory
- $100 \times 100 \times 200000 = 2 \cdot 10^9$ field grid points → 30 Gbyte Memory

Balancing:

- Slice beam to keep CPU operations per integration step the same



- Sorting/Rebinning/Slicing only required at key positions

- Preferable larger Clusters (>100 nodes) to avoid caching/swapping of information to hard disk

Advanced Algorithms I – Non-Period Average Model

Superradiant regime and HHG seed signal can violate the slow varying amplitude assumption:

$$\partial_z u(z) << k\, u(z)$$

Period averaged codes (Paraxial & Slowly Varying Envelope Approximation):

$$\left[\nabla_\perp^2 + 2ik\left(\frac{\partial}{\partial z} + \frac{\partial}{c\,\partial t}\right)\right]u = i\frac{e^2\mu_0}{m}\sum_j \frac{f_c K}{\gamma_j} e^{-i\theta_j}$$

Larger integration steps
Simpler equations

Harmonics in separate frequency bands
Coupling has to be calculated analytically

Non period average codes (Paraxial approximation only):

$$\left[\nabla_\perp^2 + 2ik\left(\frac{\partial}{\partial z} + \frac{\partial}{c\,\partial t}\right)\right]u = i\frac{e^2\mu_0}{m}\sum_j q_j \delta\left(\vec{r}_\perp - \vec{r}_{\perp,j}\right)\delta\left(t - t_j\right)\frac{K}{\gamma_j} e^{-i\frac{k_u}{1-\beta_z}(ct-z)}$$

Only single, broadband field
Coupling comes naturally

Requires sub-period integration steps
More terms in FEL equation to be calculated

Recursive Field Solver (Basic Principle)

Common solver for partial differential equations on parallel computers due to “local” calculation of the field.
Example of 1D free space paraxial equation:

$$\left[\partial_{xx} + 2ik\partial_z\right]u = 0$$

Field is sampled with a grid spacing Δx and an integration step Δz.
The field points can be represented as a vector and the differential operation as a matrix operation:

$$\begin{pmatrix} -2 & 1 & 0 & \dots \\ 1 & -2 & 1 & \dots \\ 0 & 1 & -2 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix} \vec{u}(z+\Delta z) + 2ik\Delta x^2 \frac{\left(\vec{u}(z+\Delta z) - \vec{u}(z)\right)}{\Delta z} = 0$$

Known and treated as constant

Rearranged for the *l*th element of u(z+Δz)

$$u_i = \frac{u_{i+1} + u_{i-1}}{2(1 - ik\Delta x^2/\Delta z)} + c$$

Start with arbitrary solution and use equation as a recursive algorithm

Convergence is given due to “diagonal dominance” of the system

Error will average out to zero

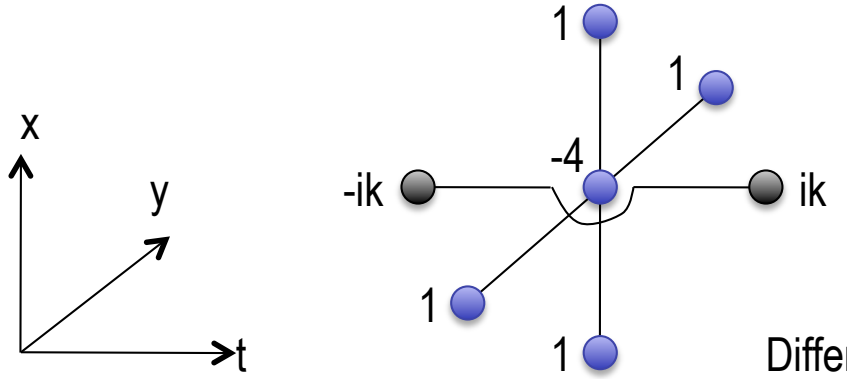
Ideal for parallel computers because only neighbor points are used

Advanced Algorithm II – Fully 3D Field solver

Solving the u(x,y,t,z) on a 3D grid with a single solver (instead of N_t 2D grid solver in quasi-time-dependent solver).

Example: Stanza of finite-difference approach

$$\left[\nabla_{\perp}^{2} + 2ik\left(\frac{\partial}{\partial z} + \frac{\partial}{c\,\partial t}\right)\right]u = i\frac{e^{2}\mu_{0}}{m}\sum_{j}\frac{f_{c}K}{\gamma_{j}}e^{-i\theta_{j}}$$



Differential operators are represented by a matrix operation

Very good initial guess is previous solution, shifted in time by the slippage length over the integration step

Allows for very powerful solver (e.g. Adaptive Multigrid Solver to avoid “empty grid” calculation)

Only one of many algorithms to solve the problem

Challenges beyond the FEL physics

Some FEL research is based on physics outside of the FEL:

1) High Harmonic Generation → Ionization and recombination of atoms

2) Self-seeding with crystals → dynamic diffraction theory

Physics beyond the scope of FEL codes.
Needs stepwise calculations

However the free space transport of the radiation can be improved to include basic optical elements (mirrors, apertures, monochromators etc).

Support for FEL Oscillator configurations

PAUL SCHERRER INSTITUT

PSI

FEL Codes have been very sucessful with single pass SASE FEL, but

- **New schemes are difficult for existing codes to model self-consistently**

Some codes could adapt to the new challenges (e.g. changing loop order)

However, fully dedicated code for parallel computers can improve on the model.

- **Are the missing physics important for existing and planned facilities?**