Wellcome to the 2nd Day of FLASH2020+ Start to End Simulation Workshop

11:00	**10 - Introduction to Puffin**	*Dr Pardis Niknejadi*	
	DESY		11:00 - 11:30
	11 - Setting up ELEGANT lattice for S2E simulations	*Dmitrii Samoilenko*	
	DESY		11:30 - 12:00
12:00	**13 - Bridging Elegant to Genesis simulations with a pseudo-one4one approach**	*Mihai Pop*	
	DESY		12:00 - 12:30
	Short Coffee Break		
	14 - EEHG Working Points in GENESIS and ELEGANT”	*Fabian Pannek et al.*	
13:00	*DESY*		12:45 - 13:15
	15 - Summary	*Dr Pardis Niknejadi*	
	16 - Final Disscussion		
	DESY		13:30 - 14:00
14:00			

FLASH2020+

FLASH2020+ Start to End Simulation Workshop

Introduction to Puffin

Pardis Niknejadi
On behalf of FLASH2020+ team
Hamburg, 17.12.2021

About Puffin

- New code on the block
- SVES vs. PIC
- How to run

Technical comparison between Puffin and Genesis 1.3 V4

- General
- Seeding specific

Seeding simulations in Puffin

- With in the S2E framework
- Status and Prerequisites

About Puffin

New code on the block

Sources: Talk by L.T. Campbell (2012),
Physics of Plasma 19, 093119 (2012)
Puffin Manual
and IPAC2018 contribution

Puffin: Parallel Unaveraged Fel Integrator (L.T. Campbell & B.W.J. McNeil)

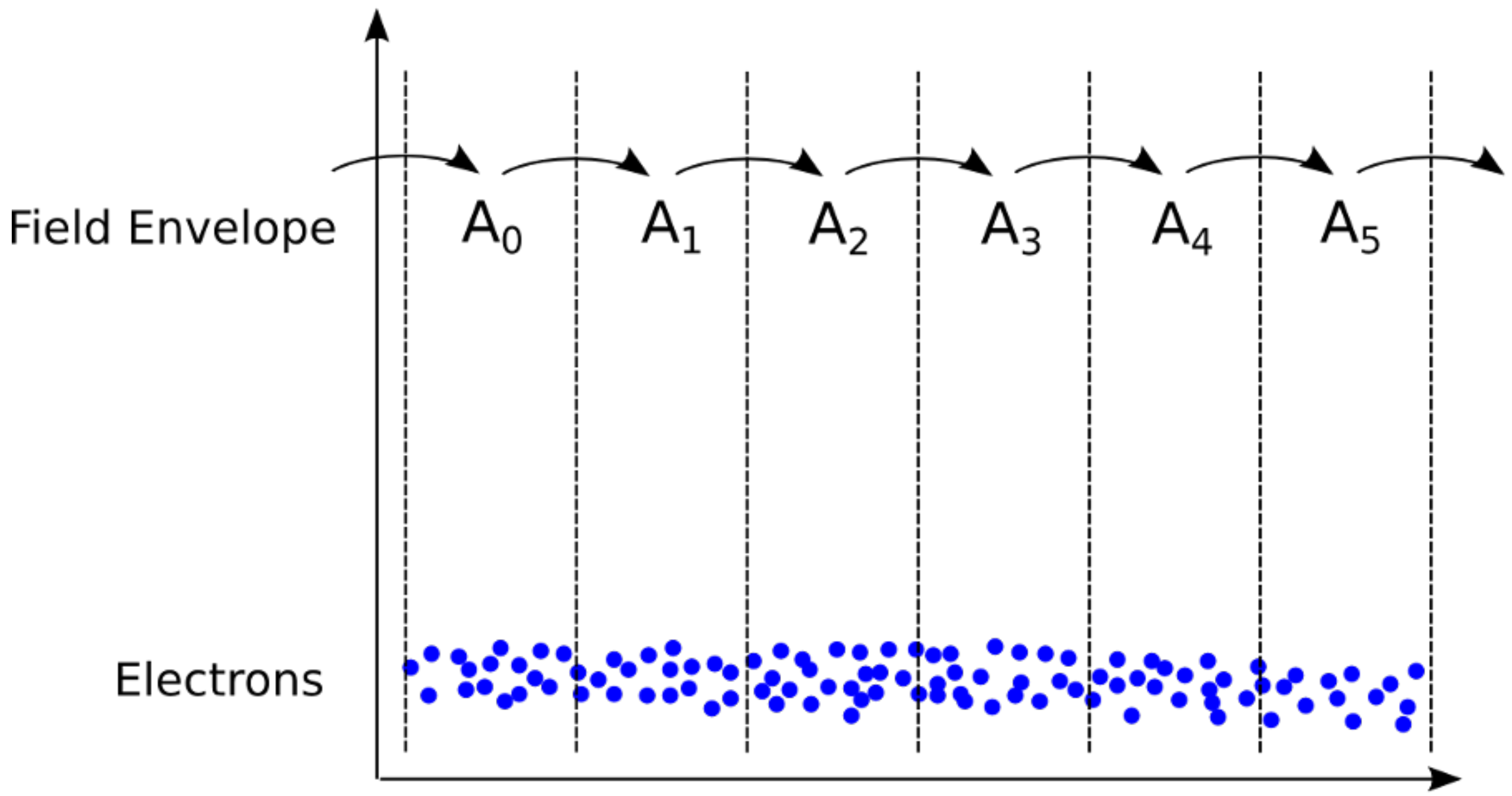
- Is the first Unaveraged Free Electron Laser (FEL) code available in 3D
    - For first 1D see C. Maroli, V. Petrillo & M. Ferrario in Review of Modern Physics 14, 7 (2011)
- Has at least two versions
- Uses a PIC approach instead of SVEA
- Written in Fortran 90 using MPI
- Models a variably polarised undulator FEL
- Intended for
    - short electron pulses which may exhibit a large degree of coherent emission (Laser Plasma Accelerator),
    - electron pulses with large changes in the relative electron positions

About Puffin

SVES vs. PIC

Slowly Varying Envelope Approximation (SVEA), and use averaged quantities to drive field

To simulate relative electron-radiation 'slippage,' pass field on to next slice

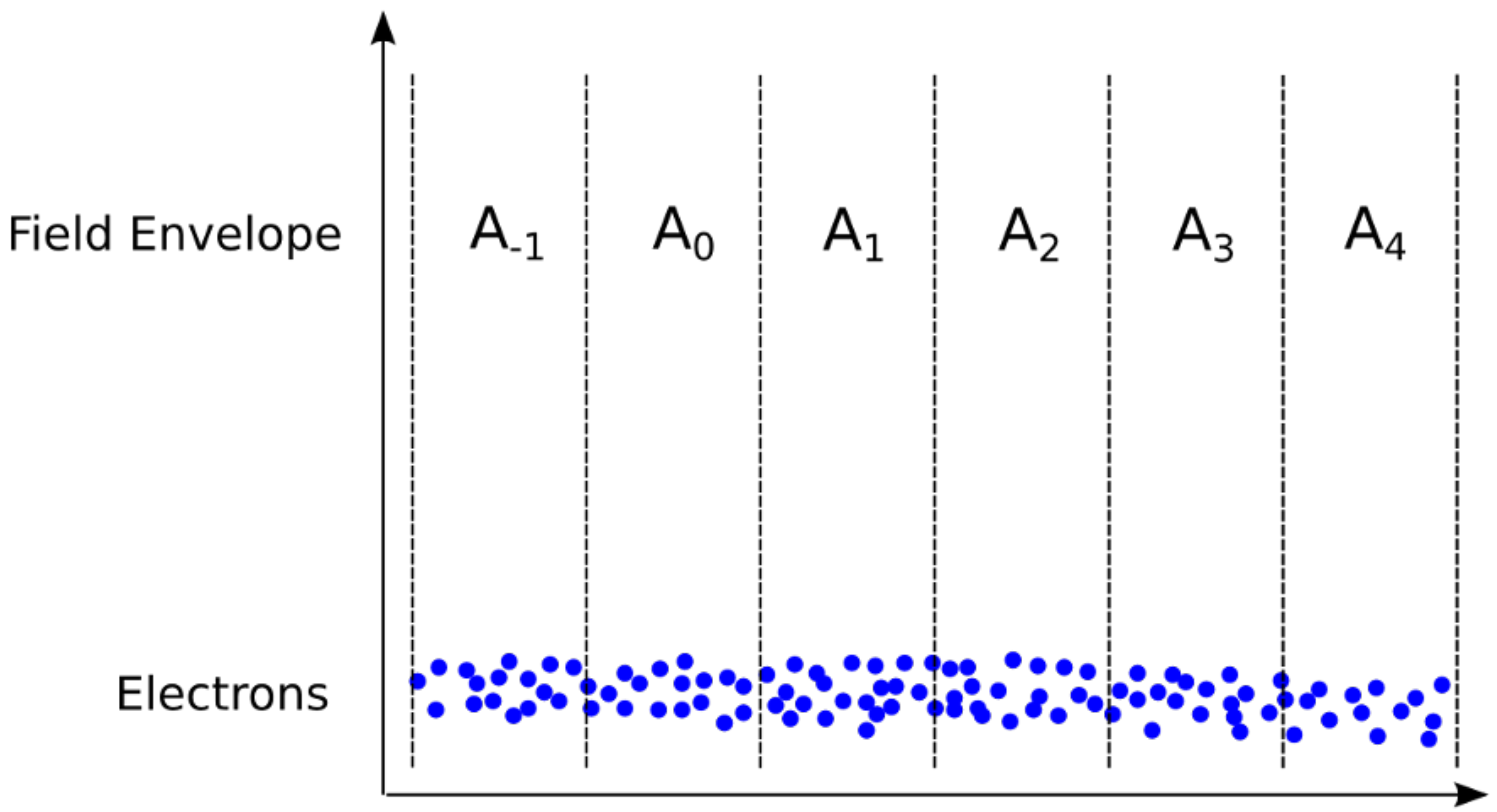


About Puffin

SVES vs. PIC

Slowly Varying Envelope Approximation (SVEA), and use averaged quantities to drive field

To simulate relative electron-radiation 'slippage,' pass field on to next slice

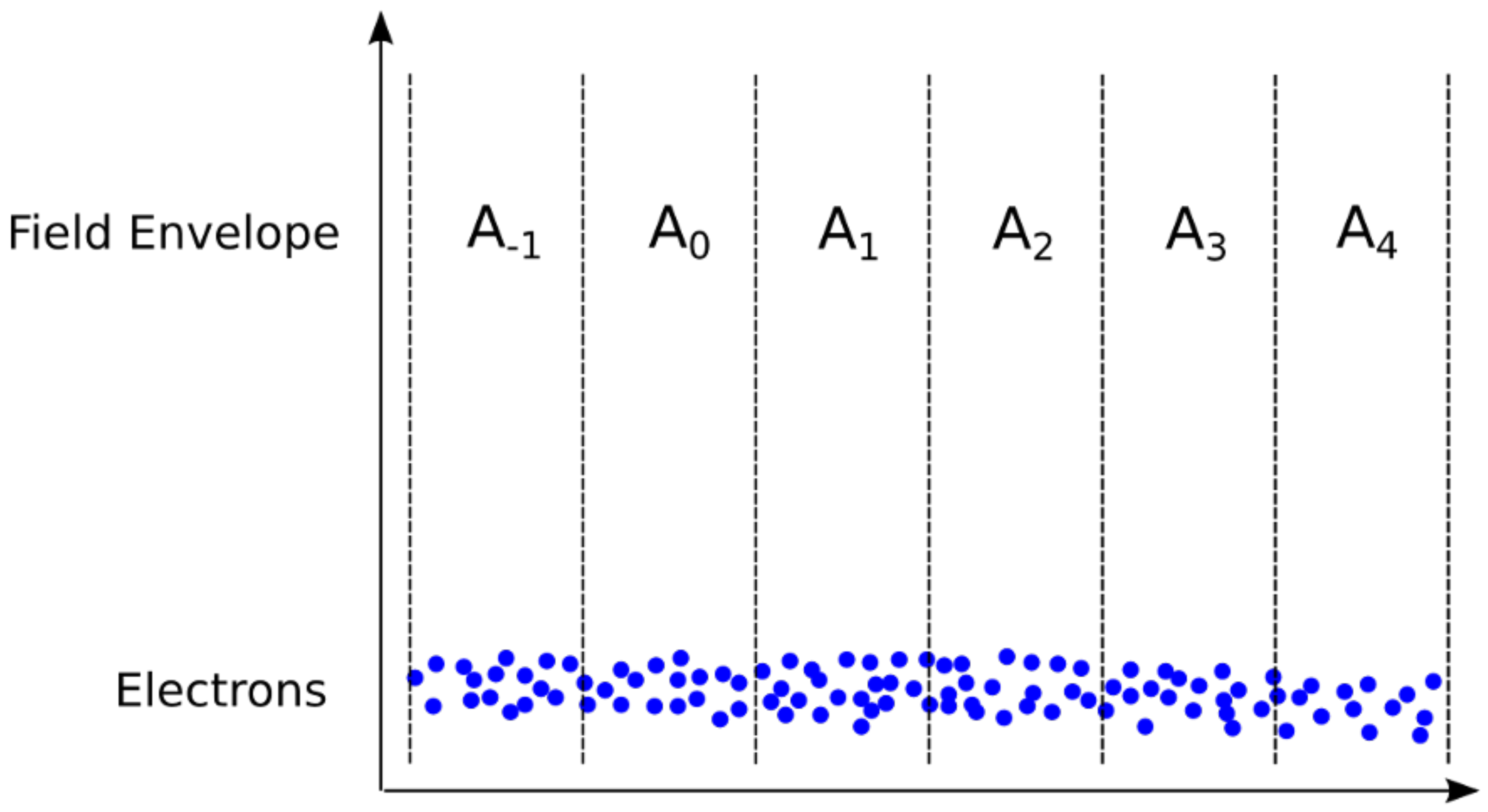


About Puffin

SVES vs. PIC

Slowly Varying Envelope Approximation (SVEA), and use averaged quantities to drive field

To simulate relative electron-radiation 'slippage,' pass field on to next slice



In an unaveraged (PIC) code: the radiation field envelope can vary quickly and the full radiation spectrum is modeled self consistently (limited by the Nyquist frequency)

Electrons are not confined to slices:

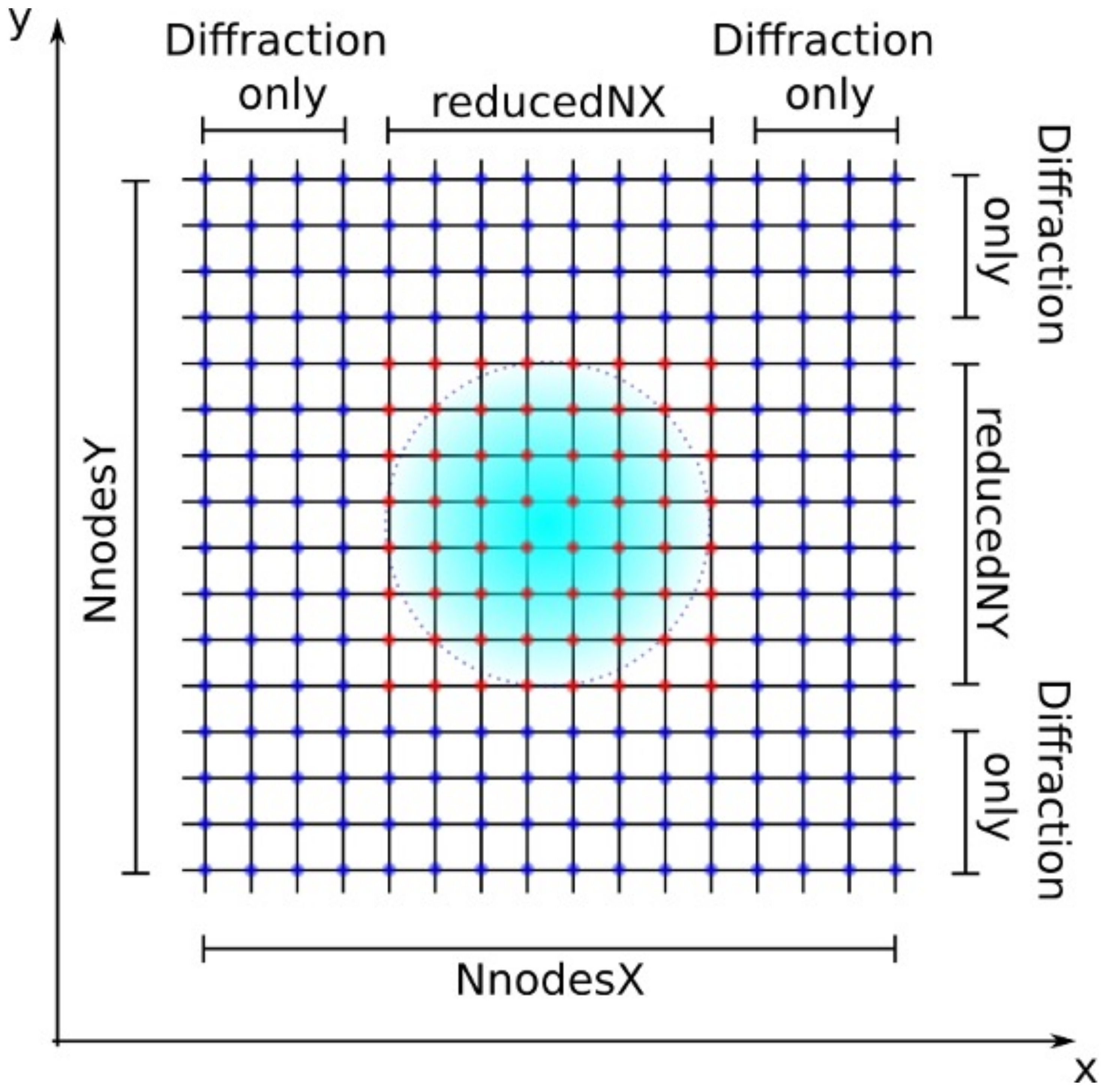
- An electron beam is Sampled with a gaussian current distribution
- Then a grid is created
- Into each grid element a macroparticle with charge weight determined by the current distribution is assigned
- Random Poisson deviation is assigned to each macroparticle charge weight & position to simulate noise
- Then a Fourier field solver is used

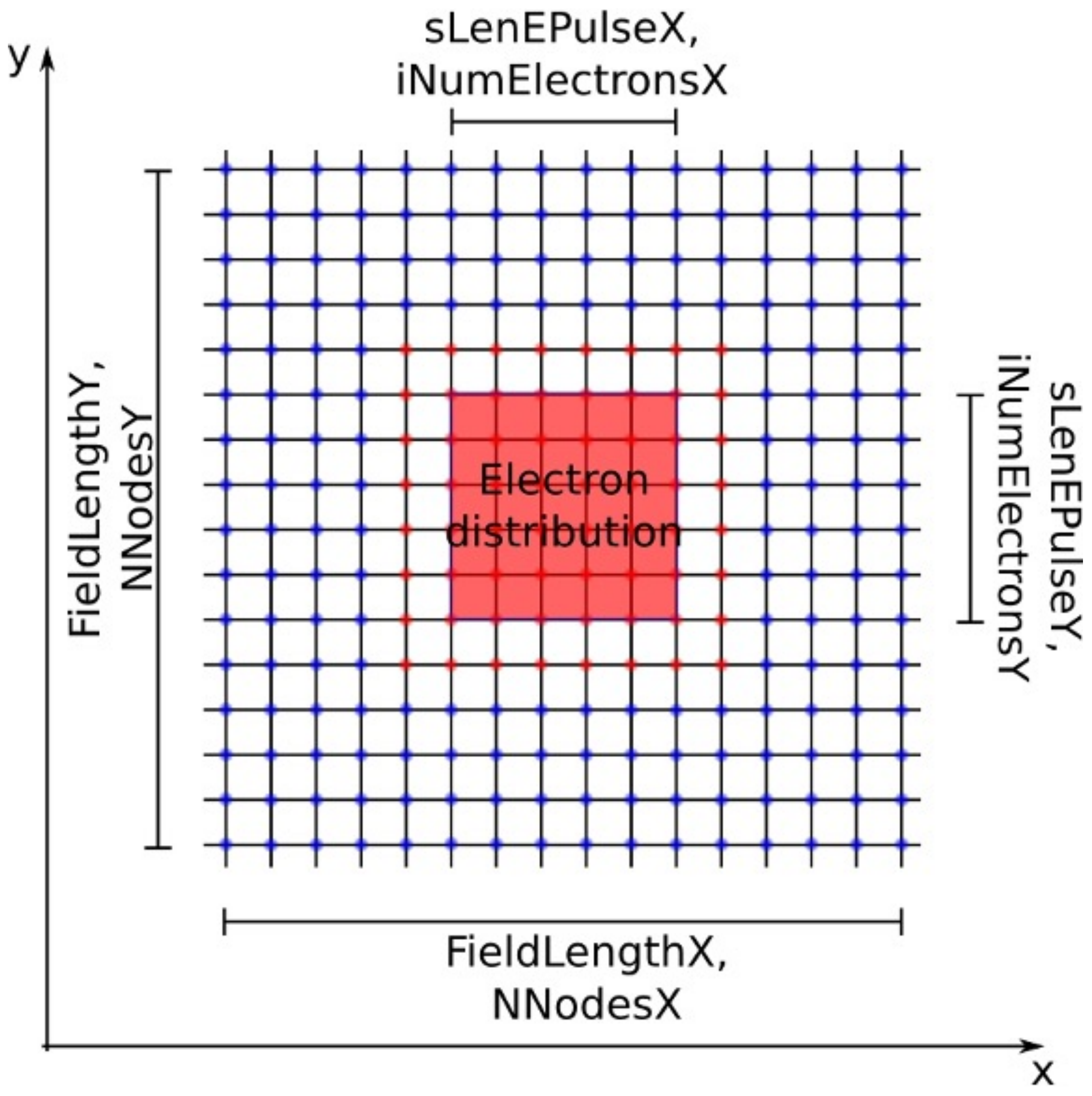
About Puffin

Field solver

The Field is solved with split step Fourier method:

- **1st half step: Field diffraction solved with Fourier Transforms**
- **2nd half step: Field source (amplification and absorption by electron macroparticles) and electron propagation**





- Electron beam has to be matched to undulator and is confined to the center nodes
- The outer nodes are only use for diffraction (if these cause unrealistic effects they can be filtered)
- electron macroparticles are interpolated directly onto the field mesh (no external solver is used in the latest version)

About Puffin

How to Run

Main file (min require):
If there is a simple beam and one undulator, every thing can be defined in this file.

```
&MDATA
 qScaled                  = .false.
 qOneD                    = .false.
 qFieldEvolve             = .true.
 qElectronsEvolve         = .true.
 qElectronFieldCoupling   = .true.
 qFocussing               = .false.
 qDiffraction             = .true.
 qUndEnds = .true.
 qFMesh_G = .false.
 beam_file                = 'xseedechobeam.in'
 sElectronThreshold       = 0.05
 iNumNodesX               = 100
 iNumNodesY               = 100
 nodesPerLambdar          = 30
 sFModelLengthX           = 0.0033
 sFModelLengthY           = 0.0033
 sFModelLengthZ2          = 106.43E-14
 iRedNodesX               = 40
 iRedNodesY               = 40
 sFiltFrac                = 0.3
 sDiffFrac                = 1.0
 saw                      = 7.049746
 sgamma_r                 = 2641.884
 lambda_w                 = 0.0826
 zundType                 = 'planepole'
 seed_file                = 'seed.in'
 lattFile                 = 'eehg4nm.latt'
 stepsPerPeriod           = 30
 nPeriods                 = 30
 sZ0                      = 0.0
 iWriteNthSteps           = 450
 iWriteIntNthSteps        = 30
 iOutInfo                 = 3
```

About Puffin

How to Run

Lattice file

```
! *********************************************
! Undulator - use 'UN'
! zundtype_arr, nw, mf, tapers, nperlam, ux_arr, uy_arr, kbnx_arr, kbny_arr
!
! ...being...
!
! undulator type,
! number of periods (integer),
! initial tuning alpha = aw/aw0,
! taper d/dz alpha,
! number of integration steps per undulator period (integer),
! ux and uy - relative magnetic field strengths in x and y, respectively
! kbnx_arr and kbny_arr - betatron wavenumber (k_beta = 2pi / lambda_beta)
! of constant 'strong' focusing channel applied over undulator
!
```

NOT A FULLY WORKING EXAMPLE

```
UN  'planepole'  30   1.0   0.0   30   0.0   1.0   0.0    0.0
DR   10
CH   74.14043583535108  3.521233e-3  2895.967896575498
DR   10
```

Lattice elements: UN: undulator, CH: chicane, MO: energy modulation, DR: drift, QU: focusing

About Puffin

How to Run

Beam file

```
&NBLIST
nbeams = 1
dtype = 'simple'
/

&BLIST
sSigmaE = 5.0E-6, 5.0E-6, 13.00E-15, 1.0, 1.0, 2.6E-5
sLenE = 40.0E-6, 40.0E-6, 136.6E-15, 1.0, 1.0, 0.0093
iMPsZ2PerWave = 30
!sQe = 800E-12
Ipk = 500
emitx = 3.61E-11
emity = 3.61E-11
chirp = 0
alphax = 1.251302
alphay = -1.078255
qRndEj_G = .false.
sSigEj_G = 0.4
qMatched_A = .false.
TrLdMeth = 2
nseqparts = 500
!qFixCharge = .false.
/
```

About Puffin

How to Run

Seed file

```
&NSLIST
nseeds = 2
dtype = 'simple'
/

&SLIST
freqf = 0.4167, 0.5
sA0_X = 1.0, 2.0
sA0_Y = 1.0, 2.0
sSigmaF = 5, 5, 5, 5, 1.0E-8, 1.0E-8
qFlatTop = .true., .true.
meanZ2 = 0.0, 45
qRndFj_G = .false., .false.
sSigFj_G = 0.1, 0.1
/
```

ONLY INCLUDES IDEAL FIELD

Technical comparison between Puffin and Genesis 1.3 V4

General

- Puffin is not limited to the resonant frequency
    - One can have two different seed wavelength in one simulation as long as their wavelength is close (I have not checked the limit of this)

- It has a scaled and unscaled mode (and there is no checking mechanism for the most part)
    - However, there are dumps in SI in addition the the scaled dumps, and one could read these and see if they made sense

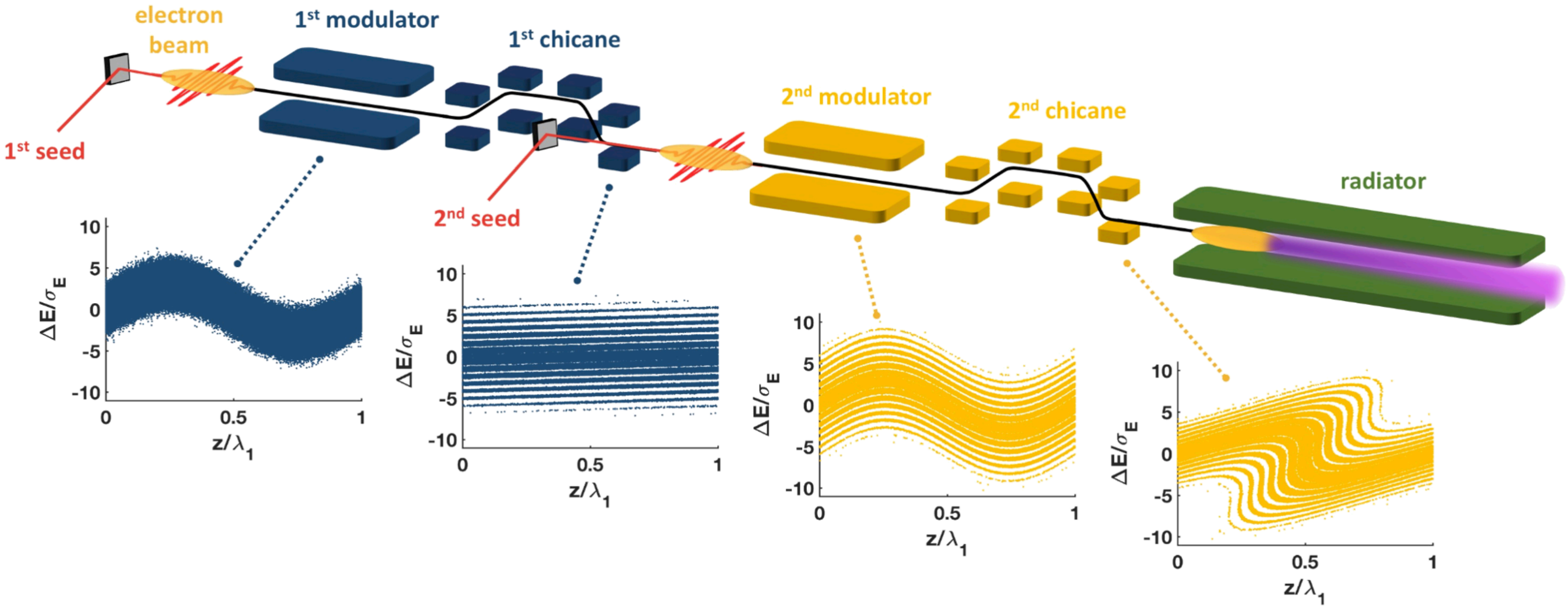
- There is not the possibility to add or remove dump flags, dumps of integrated (similar to out file of Genesis) or full dumps can be set periodically
    - If you want full beam dump you also get field dump
    - So even if the simulation could take similar resources as Genesis when output period is set to as long as the simulations (where one would get 0 and n dump only), investigatory runs can take up a lot of time, resources, and space*

Seeding specific

- There is no resorting implemented and “z slices” can not be changed in one simulation
- The chicane model is very simple (appropriate for phase shifter and possibly our small chicane but definitely fails for the over shearing chicane)
- For the time being only ideal seed laser are possible

Seeding simulations in Puffin

With in the S2E framework



FLASH2020+ Start to End Simulations and Categories

The Planned structure for S2E simulations and incorporating different categories

Start with an on-crest file & design optics File in MAD

Add compression, adjust optics & Track in Elegant

Impact-Z

Model beam realistically from the Gun

Seed1 and Seed2

Chi2 3D

Genesis/Puffin

Beam & Seed1 in Mod 1

Track Chicane 1 in Elegant

Beam & Seed2 in Mod 2

Choose the best model for Chicane 2

(a) Genesis

(b) SelaV

(c) Tracking Single Particle in Elegant

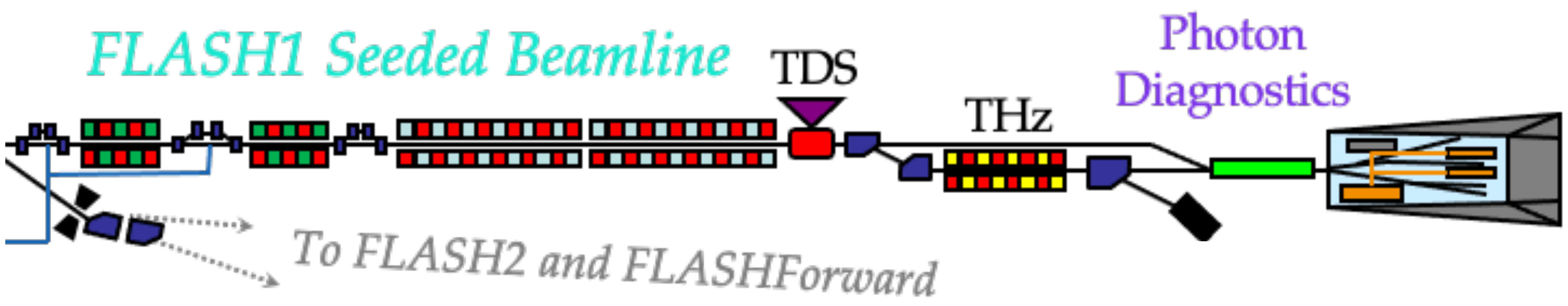
Radiators

Synchrotron Radiation Workshop (SRW) Transports to the Photon Beamline

End

FLASH2020+ Start to End Simulations and Categories

Codes for externally seeded FEL simulations



Seed1 and Seed2

Chi2 3D

Resolving issues with integration

(a) Genesis

Genesis / Puffin

(b) SelaV

Chicane 1

Elegant

Chicane 2

(c) Tracking Single Particle in Elegant

Radiators

Beam & Seed1 in Mod 1

Beam & Seed2 in Mod 2

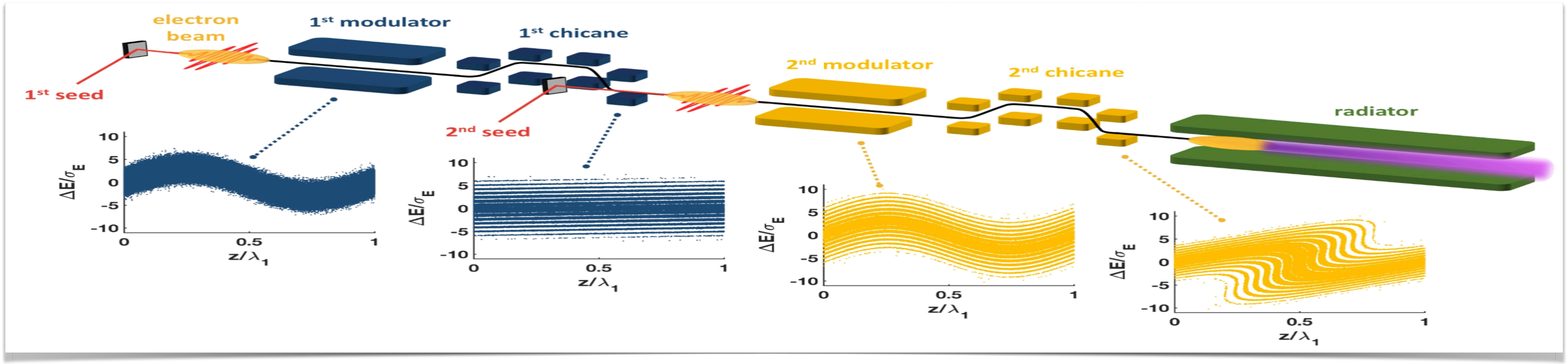
End with Transport to Photon BL

Synchrotron Radiation Workshop

Benchmarking and comparing three approaches (a-c) for Seeded Beamline

FLASH2020+ Start to End Simulations and Categories

Codes for externally seeded FEL simulations



Beam & Seed1 in Mod 1

Seed1 Genesis

Chi2 3D

Seed1 Puffin

Chicane 1

Elegant

Beam & Seed2 in Mod 2

Seed2 Genesis

Chi2 3D

Seed2 Puffin

Chicane 2

SelaV

Elegant

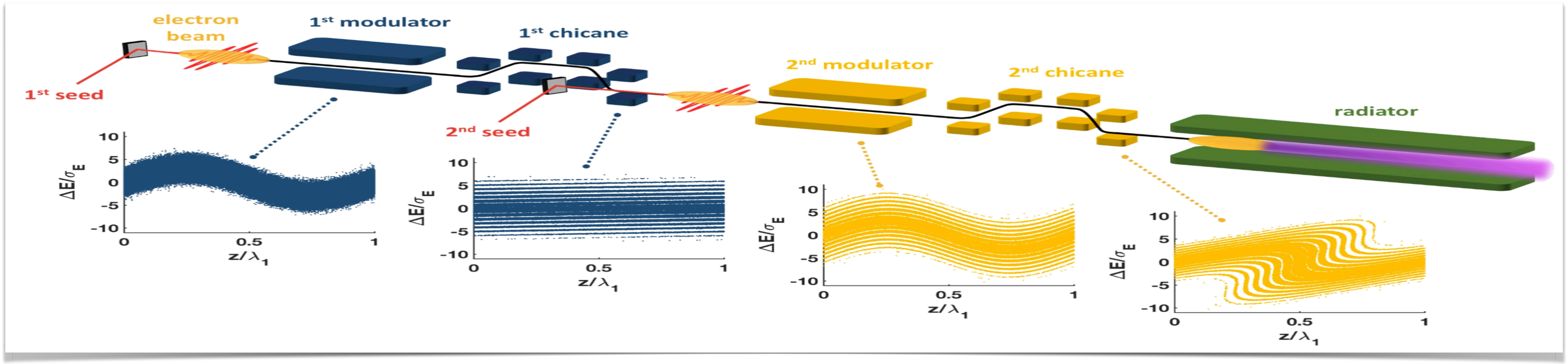
Radiators

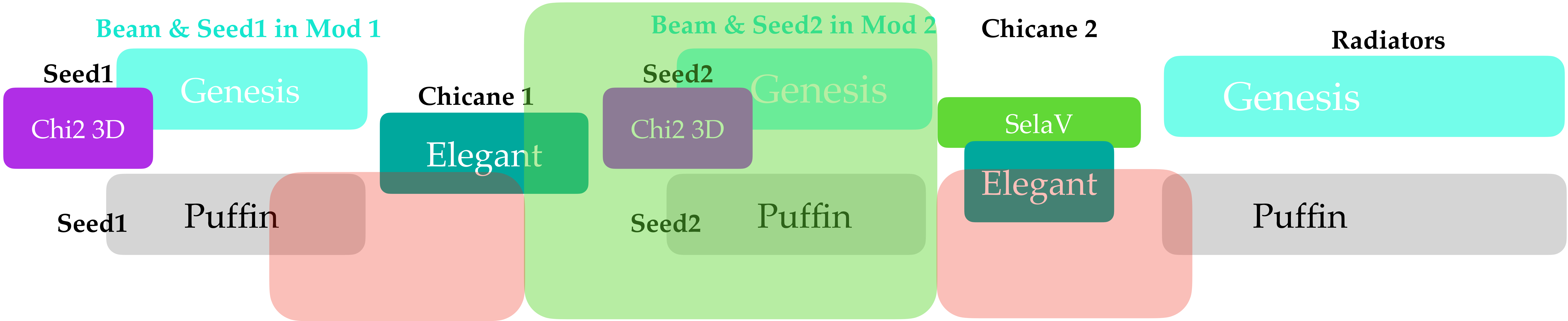
Genesis

Puffin

FLASH2020+ Start to End Simulations and Categories

Codes for externally seeded FEL simulations





Seeding simulations in Puffin

Status and Prerequisites

- **Status**
  - Output of elegant can be upsampled (Modified script from the UK-FEL collaboration) and changed to Puffin or Genesis beam format then imported to either simulations
  - In Puffin:
    - As an input to a radiator this works fine
    - As an input to the modulator (with laser field added) ->work in progress (Not sure how the seed input is scaled or not)
  - Chicane definition will most likely work for the second EEHG chicane
- **Prerequisites**
  - Benchmark only simple simulations
  - Finding another approach to benchmarking S2E

Tentative Conclusions and Outlook

Considering Start to End simulations and Benchmarking

- S2E: For the external seeding in FLASH1, Puffin in it's current version is missing a few key capabilities to be used for EEHG
- There is the option to model EEHG in puffin with a periodic method, this could be bench marked against what you hear in the later talks today

Outlook:

- A lot more time needs to be invested to make S2E in Puffin work

Questions:

- Are the simulations done for comparison with FERMI data here available somewhere:

B. Garcia et al. COMPARING FEL CODES FOR ADVANCED CONFIGURATIONS, FEL2017, Santa Fe, NM, USA

Tentative Conclusions and Outlook

Considering Start to End simulations and Benchmarking

- S2E: For the external seeding in FLASH1, Puffin in it’s current version is missing a few key capabilities to be used for EEHG
- There is the option to model EEHG in puffin with a periodic method, this could be bench marked against what you hear in the later talks today

Questions:

Are the simulations done for comparison

with FERMI data here available somewhere:

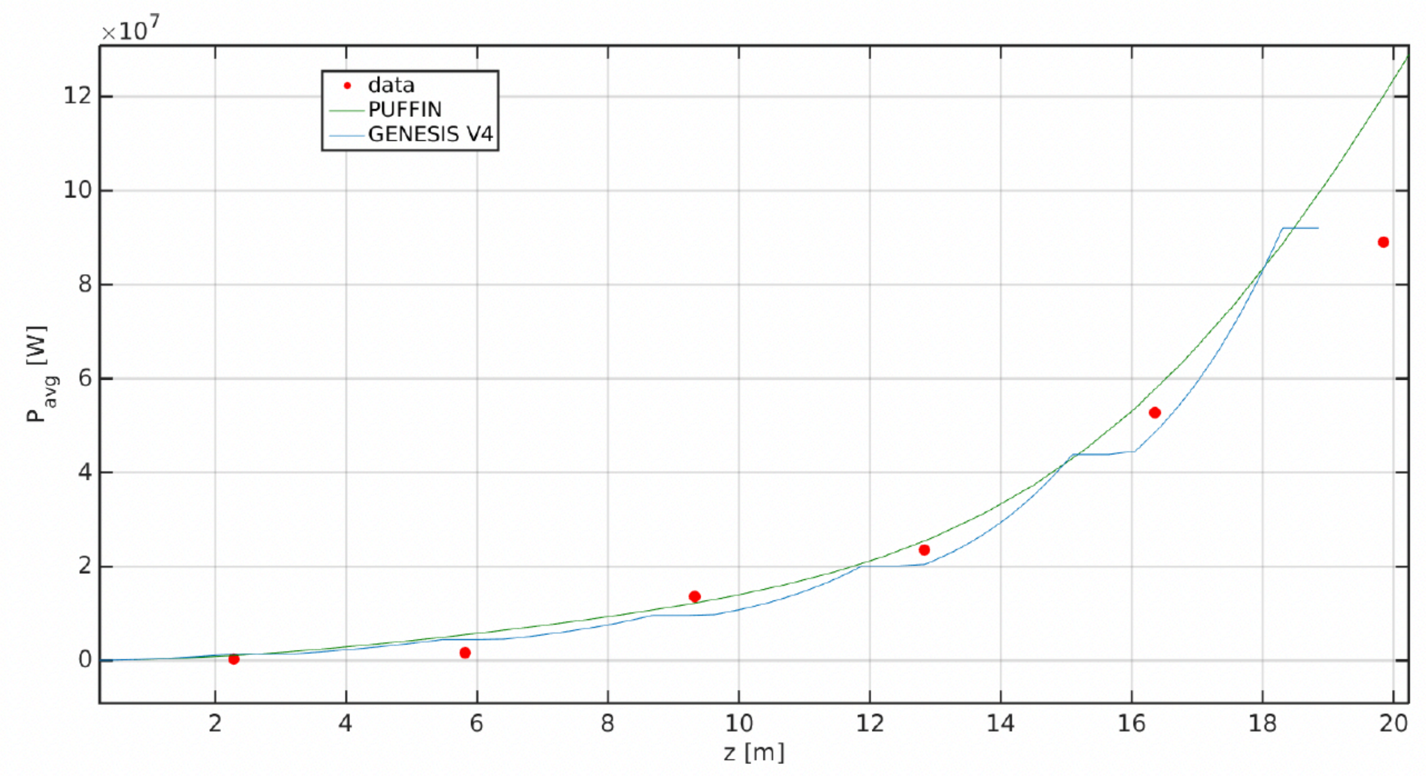


Figure 3: The average power versus z for the FERMI HGHG setup from PUFFIN, GENESIS, and measured data.

B. Garcia et al. COMPARING FEL CODES FOR ADVANCED CONFIGURATIONS, FEL2017, Santa Fe, NM, USA

Thank You