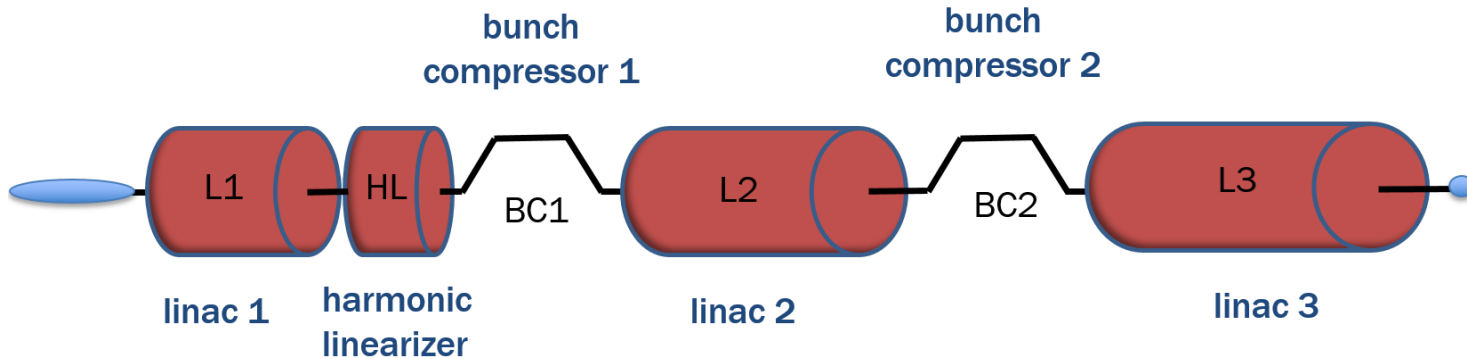


Practice Session Homework



- Initial beam conditions: 100pC, 100 MeV, total 3 mm beam with longitudinally uniform distribution in z , Gaussian energy distribution with standard deviation 2keV, no chirp, transverse round upright 4D Gaussian distribution with 0.2 mm RMS size, and 0.5 μm normalized emittance.
- Linac parameters: L1 (1.3GHz, 20m, 16MV/m), $k = 0.61685$
HL (3.9GHz, 5m, 10MV/m), $k = 0.61685$
L2 (1.3GHz, 200m, 16MV/m), $k = 0.61685$
L3 (1.3GHz, 400m, 16MV/m), $k = 0.61685$
BC1 $R_{56} = 5 \text{ cm}$, $L = 10\text{m}$, $k = 0.27416$
BC2 $R_{56} = 5 \text{ cm}$, $L = 20\text{m}$, $k = 0.27416$
- Undulator parameters: undulator period: 2cm, strength (K): 1.5.

Practice Session Homework Problem

- 1) Find the final electron beam bunch length at the end of L3, estimate FEL radiation wavelength, bandwidth, and power with initial initial emittance, compute speedup of your parallel program
- 2) Find the final electron beam energy spread at the end of L3, estimate FEL radiation wavelength, bandwidth, and power with initial emittance, compute speedup of your parallel program
- 3) Find the final electron beam emittance at the end of L3, estimate FEL radiation wavelength, bandwidth, and power with initial peak current and maximum final energy, compute speedup of your parallel program

Sampling of Distribution

- Sample the Gaussian probability density function:

$$\phi'(y|0, 1) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}y^2\right], \quad -\infty < y < \infty,$$

- Form a 2D Gaussian probability density function:

$$f(y_1, y_2) = \phi'(y_1|0, 1) \phi'(y_2|0, 1) = \frac{1}{2\pi} \exp\left[-\frac{1}{2}(y_1^2 + y_2^2)\right].$$

- Change the coordinate:

$$Y_1 = R \cos \Phi,$$

$$Y_2 = R \sin \Phi,$$

$$\phi'(y_1)\phi'(y_2) dy_1 dy_2 = \left(\exp\left[-\frac{1}{2}r^2\right] r dr \right) \left(\frac{1}{2\pi} d\phi \right).$$

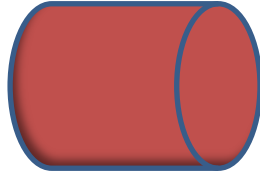
$$\Phi = 2\pi\xi_2.$$

$$Y_1 = [-2 \log \xi_1]^{\frac{1}{2}} \cos 2\pi\xi_2,$$

$$Y_2 = [-2 \log \xi_1]^{\frac{1}{2}} \sin 2\pi\xi_2;$$

Longitudinal Dynamics Inside the Accelerator for Problem 1 and 2

RF cavity



bunch compressor



z : longitudinal relative position deviation w.r.p. reference particle position

$\Delta\gamma$: longitudinal normalized relative energy deviation w.r.p. reference particle energy

$$z^+ = z_1 + \frac{L_{\text{acc}}}{2} \Delta\gamma_1 / (\gamma_{01} \beta_{01})^3$$

$$\gamma_0^+ = \gamma_{01} + \frac{L_{\text{acc}}}{2} \frac{qV_{\text{acc}}}{mc^2} \cos(\phi_0)$$

$$\Delta\gamma_2 = \Delta\gamma_1 + L_{\text{acc}} \frac{qV_{\text{acc}}}{mc^2} (\cos(\phi_0 - kz^+) - \cos(\phi_0))$$

$$z_2 = z^+ + \frac{L_{\text{acc}}}{2} \Delta\gamma_2 / (\gamma_0^+ \beta_0^+)^3$$

$$\gamma_{02} = \gamma_0^+ + \frac{L_{\text{acc}}}{2} \frac{qV_{\text{acc}}}{mc^2} \cos(\phi_0)$$

$$z = z + R_{56} \frac{\Delta\gamma}{\gamma_0} + T_{566} \left(\frac{\Delta\gamma}{\gamma_0} \right)^2 + U_{5666} \left(\frac{\Delta\gamma}{\gamma_0} \right)^3$$

$$\Delta\gamma_2 = \Delta\gamma_1$$

$$R_{56} \approx 2\theta^2 \left(L_{db} + \frac{2}{3} L_b \right)$$

$$T_{566} \approx -\frac{3}{2} R_{56}$$

$$U_{5666} \approx 2R_{56}$$

Transfer Matrix through Linac and Bunch Compressor for Problem 3

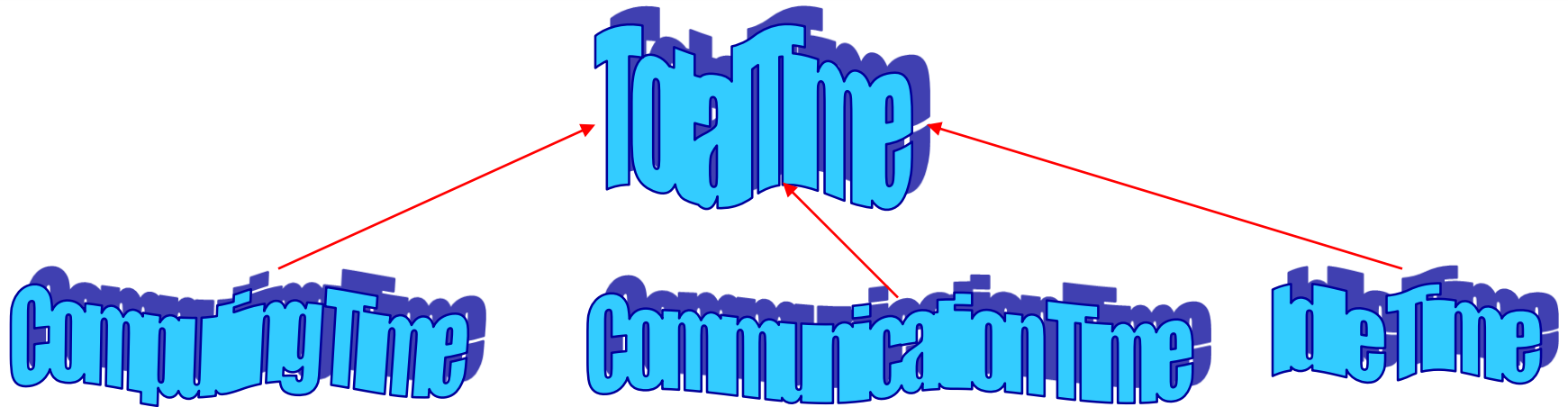
Linac Transfer Matrix

$$\left(\begin{array}{cc|cc|cc} \cos \sqrt{k}z & \frac{1}{\sqrt{k}} \sin \sqrt{k}z & & & & \\ -\sqrt{k} \sin kz & \cos \sqrt{k}z & & & & \\ \hline & & \cos \sqrt{k}z & \frac{1}{\sqrt{k}} \sin \sqrt{k}z & & \\ & & -\sqrt{k} \sin kz & \cos \sqrt{k}z & & \\ \hline & & & & 1 & \frac{1}{\gamma_o^2 \beta_o^2} z \\ & & & & 0 & 1 \end{array} \right)$$

Bunch Compressor Transfer Matrix

$$\left(\begin{array}{cc|cc|cc} \cos \sqrt{k}z & \frac{1}{\sqrt{k}} \sin \sqrt{k}z & & & & \\ -\sqrt{k} \sin kz & \cos \sqrt{k}z & & & & \\ \hline & & \cos \sqrt{k}z & \frac{1}{\sqrt{k}} \sin \sqrt{k}z & & \\ & & -\sqrt{k} \sin kz & \cos \sqrt{k}z & & \\ \hline & & & & 1 & R56 \\ & & & & 0 & 1 \end{array} \right)$$

Parallel Implementation



parallel speedup

$$S(n, P) = \frac{T(n, 1)}{T(n, P)}$$

parallel efficiency

$$E(n, P) = \frac{S(n, P)}{P} = \frac{T(n, 1)}{PT(n, P)}$$

$$T = (1 - \alpha) T + \alpha T$$

α is the fraction of time that cannot be done in parallel

$$T(P) = (1 - \alpha) T/P + \alpha T \quad S(P) = 1/[(1 - \alpha) 1/P + \alpha] \quad S(\infty) = 1/\alpha$$

❖ the smaller fraction of serial time, the better parallel efficiency!!

MPI Code Example

C Language - Blocking Message Passing Routines Example

```
#include "mpi.h"
#include <stdio.h>

int main(argc,argv)
int argc;
char *argv[]; {
int numtasks, rank, dest, source, rc, count, tag=1;
char inmsg, outmsg='x';
MPI_Status Stat;

MPI_Init(&argc,&argv);
MPI_Comm_size(MPI_COMM_WORLD, &numtasks);
MPI_Comm_rank(MPI_COMM_WORLD, &rank);

if (rank == 0) {
    dest = 1;
    source = 1;
    rc = MPI_Send(&outmsg, 1, MPI_CHAR, dest, tag, MPI_COMM_WORLD);
    rc = MPI_Recv(&inmsg, 1, MPI_CHAR, source, tag, MPI_COMM_WORLD, &Stat);
}

else if (rank == 1) {
    dest = 0;
    source = 0;
    rc = MPI_Recv(&inmsg, 1, MPI_CHAR, source, tag, MPI_COMM_WORLD, &Stat);
    rc = MPI_Send(&outmsg, 1, MPI_CHAR, dest, tag, MPI_COMM_WORLD);
}

rc = MPI_Get_count(&Stat, MPI_CHAR, &count);
printf("Task %d: Received %d char(s) from task %d with tag %d \n",
       rank, count, Stat.MPI_SOURCE, Stat.MPI_TAG);

MPI_Finalize();
}
```

Fortran - Blocking Message Passing Routines Example

```
program ping
include 'mpif.h'

integer numtasks, rank, dest, source, count, tag, ierr
integer stat(MPI_STATUS_SIZE)
character inmsg, outmsg
outmsg = 'x'
tag = 1

call MPI_INIT(ierr)
call MPI_COMM_RANK(MPI_COMM_WORLD, rank, ierr)
call MPI_COMM_SIZE(MPI_COMM_WORLD, numtasks, ierr)

if (rank .eq. 0) then
    dest = 1
    source = 1
    call MPI_SEND(outmsg, 1, MPI_CHARACTER, dest, tag,
    & MPI_COMM_WORLD, ierr)
    call MPI_RECV(inmsg, 1, MPI_CHARACTER, source, tag,
    & MPI_COMM_WORLD, stat, ierr)

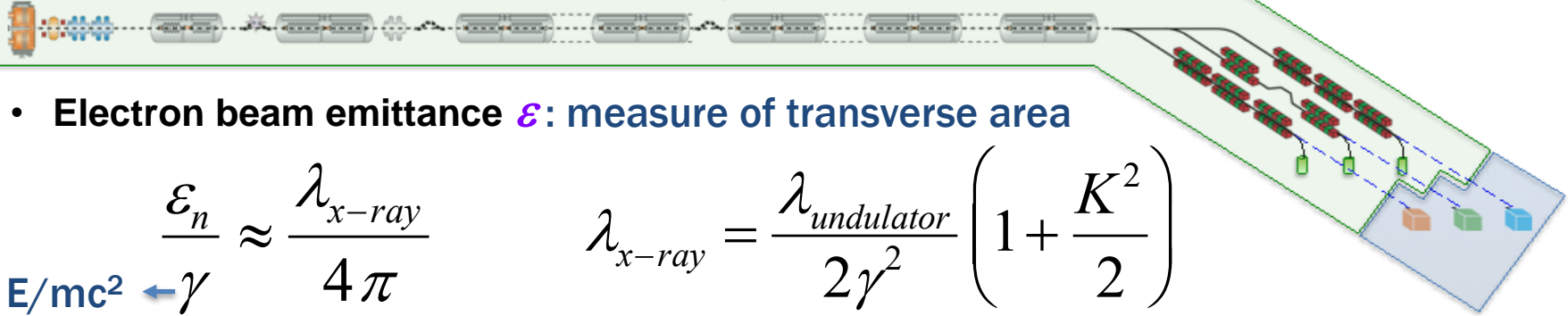
else if (rank .eq. 1) then
    dest = 0
    source = 0
    call MPI_RECV(inmsg, 1, MPI_CHARACTER, source, tag,
    & MPI_COMM_WORLD, stat, ierr)
    call MPI_SEND(outmsg, 1, MPI_CHARACTER, dest, tag,
    & MPI_COMM_WORLD, ierr)
endif

call MPI_GET_COUNT(stat, MPI_CHARACTER, count, ierr)
print *, 'Task ',rank,': Received', count, 'char(s) from task',
& stat(MPI_SOURCE), 'with tag',stat(MPI_TAG)

call MPI_FINALIZE(ierr)

end
```

The FEL Cost and Performance Critically Depend on Electron Beam Brightness



- Electron beam emittance ϵ : measure of transverse area

$$\frac{\epsilon_n}{E/mc^2 \leftarrow \gamma} \approx \frac{\lambda_{x-ray}}{4\pi} \quad \lambda_{x-ray} = \frac{\lambda_{undulator}}{2\gamma^2} \left(1 + \frac{K^2}{2} \right)$$

- Peak current I_{peak}

K : undulator strength

$$P_{sat} \approx 1.6 \rho \left(\frac{L_{G,1D}}{L_G} \right)^2 P_{b,pk} \quad P_{b,pk} = I_{pk} \gamma mc^2 / e$$

- Gain length

$$L_{G0} = \frac{\lambda_u}{4\pi\sqrt{3}\rho}$$

$$\rho = \left(\frac{1}{16} \frac{I_{pk}}{I_A} \frac{K^2 [JJ]^2 \lambda_u^2}{4\pi^2 \gamma^3 \sigma_x^2} \right)^{1/3} \quad JJ(\xi) = J_0(\xi) - J_1(\xi) \quad \xi = \frac{K^2}{4 + 2K^2}$$

- Energy spread σ_E $I_A \sim 17kA$

Ideal electron beam:

- high peak current
- small energy spread
- small emittance

$$B \sim \frac{I}{\epsilon_x \epsilon_y \sigma_E}$$

$$\Delta\omega = \frac{2\sigma_\gamma/\gamma}{\lambda_r} c$$