RECENT RESULTS ON SIMULATIONS OF HIGH INTENSITY BEAMS IN CYCLOTRONS

A. Adelmann[*], R. Geus and L. Stingelin, PSI, Villigen, Switzerland

Abstract

A summary of activities that envisage a source to target simulation of the PSI cyclotron complex is presented. Our aim is to gain a quantitative understanding of complex phenomena in our machines and beam lines, including three-dimensional space charge effects, beam cavity interaction, collimation, beam neutralisation and extraction mechanism. An introduction to the mathematical and computational methods used is presented together with a comparison of simulations with measurements.

INTRODUCTION

The aim of our code and methods development is to make the step from *qualitative* to *quantitative* predictions. This requires the accurate three-dimensional modeling of large and complicated accelerator structures including space charge, beam lines, collimation, and in the future secondary effects. This effort is in line with the high intensity upgrade described in [2].

The required three-dimensional modeling will ultimately demand high performance computing (HPC) resources such as clusters of workstations or symmetric multiprocessor systems (SMP). In order to efficiently integrate existing code, we use the object oriented programing (OOP) paradigm, resulting in clearly structured and reusable software or software-components.

We use mks units if specified otherwise. $\beta = v/c$ and γ denotes the relativistic factor. By *self consistent* we understand the inclusion of space charge from the particle distribution in the electrostatic limit.

SPACE CHARGE SOLVERS

When modeling space charge dominated beams, one of the key elements is accurate and fast Poisson solvers. The fundamental steps in calculating $\vec{E}$, the electric self field are: a Lorentz transformation in the beam's rest frame $x' = \mathcal{L}(x)$, interpolation of $q_i(x')$ to obtain ρ, the charge density on a discrete space (grid). Then we solve:

$$-\triangle u = \frac{\rho}{\epsilon_0} \quad \text{on } \Omega, \tag{1}$$

$$u = 0 \quad \text{on } \partial\Omega, \tag{2}$$

where $\Omega \subset \mathcal{R}^3$ is a bounded domain, $x \in \Omega$ and ϵ_0 is the permittivity of vacuum. We then obtain the fields in the beam frame:

$$\vec{E}' = -\nabla u, \quad \vec{B}' = 0 \tag{3}$$

[*] andreas.adelmann@psi.ch

followed by an interpolation: $\vec{E}'(x')$ at particle positions x' from $\vec{E}'$. The back transformation to the beam frame $\{\vec{E}(x), \vec{B}(x)\} = \mathcal{L}^{-1}(\vec{E}')$ ends the field calculation. Dirichlet, open or periodic boundary can be used.

Particle Mesh Solver

One of the most used solvers are direct FFT based solvers (no discretization of the $\triangle$-operator). We show here the differences with respect to the general scheme previously presented. G denotes the Greens function in open space, while variables with hats are in Fourier space. To obtain the scalar potential in the beam frame, we interpolate ρ and G onto a rectangular grid, followed by a Fourier transform to obtain $\widehat{\rho}$ and $\widehat{G}$. Determining $\widehat{u} = \widehat{\rho} * \widehat{G}$ and transforming them back gives $u = FFT(\widehat{u})^{-1}$. Computing $\vec{E}' = -\nabla u$ using a second order finite difference scheme and interpolate $\vec{E}'(\vec{x'})$ at particle positions $\vec{x'}$ from $\vec{E}'$. The use of FFT reduced the computational complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ with N denoting the grid size. Using parallel FFT's one can easily parallelize this scheme and integrate it into a particle tracking program [6]. Open or periodic boundary conditions can be used.

A Novel Massively Parallel Poisson Solver

In the case of large and complicated boundaries we propose a finite element based Poisson solver (using trilinear finite elements) with a semi unstructured grid. The resulting linear system of equation is then solved with a multigrid. The same steps apply as for the previously described solver with the exception that Eqs. (1) and (2) are treated differently.

Semi-Unstructured Grid The use of a structured grid Ω_h has several advantages in comparison to a pure unstructured grid. One of them is the small storage requirement, since the discretisation stencil is a fixed stencil independent of the grid point. Other advantages are the superconvergence of the gradient and the natural construction of coarse grids. To be able to discretize more general domains, we apply so called *semi-unstructured* or embedded structured grids as depicted in Fig. 1. These grids consist of a large structured grid in the interior of the domain and an unstructured grid, which is only contained in boundary cells. A detailed description of semi-unstructured grids for general domains is given in [3]. Here, we describe only the main properties of semi-unstructured grids. A semi-unstructured grid generation is based on the structured grid Ω_h, and leads to the following objects: *interior cells*, *boundary cells* and *exterior cells*. The boundary of Ω cuts the boundary cells. This cut is approximated by triangles for ev-
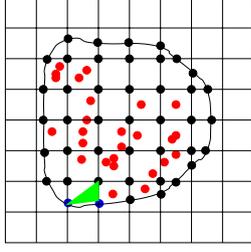
Figure 1: Color: Semi-unstructured grid with particles (red) and edge cells (green)

ery boundary cell. The union of all these triangles and all interior cells is the *discretisation domain* Ω_h. The semi-unstructured grid is the set of *nodal points*

$$\mathcal{N}_h \quad := \quad \mathcal{N}_{h,i} \cup \partial\mathcal{N}_h, \qquad (4)$$

where $\partial\mathcal{N}_h$ are the boundary nodal points and $\mathcal{N}_{h,i} \subset \Omega_h$ are the interior nodal points. The boundary nodal points are constructed in such a way that every boundary nodal point $p \in \mathcal{N}_h$ is contained in the interior of an edge of a boundary cell. Several advantages of the structured grid Ω_h still remain for the semi-unstructured grid $\mathcal{N}_h$. One of them is the low storage requirement, since the discretisation stencils of the structured grid are constant. Another is the natural construction of coarse grids up to a very coarse grid. Such constructions are important for obtaining an optimal multilevel iterative solver. Furthermore, the structured grid inside of the domain leads to a local super-convergence of the gradient. To obtain a finite element discretization of (4) we use linear elements. The discretized $\triangle$-operator results in a sparse linear system of equations:

$$A\mathbf{x} = B \qquad (5)$$

where A is the mesh dependent stiffness matrix and B represents the source term.

The Multigrid Solver A multigrid algorithm [5] is used to solve the linear system (5) of equations resulting from the finite element discretisation of $\mathcal{N}_h$ and the corresponding differential operator is based on a sequence of fine and coarse grids

$$\Omega_{h_1} \subset \Omega_{h_2} \subset \Omega_{h_2} \subset \cdots \subset \Omega_{h_n} \qquad (6)$$

and restriction and prolongation operators

$$\begin{aligned} R_{h_i} &: \Omega_{h_{i+1}} \to \Omega_{h_i} \\ P_{h_i} &: \Omega_{h_{i-1}} \to \Omega_{h_i}. \end{aligned} \qquad (7)$$

Restriction and prolongation has to be applied to FEM-spaces and to the differential operators from fine to coarse grid (and vice versa). Depending on the grid and the operator, additional structures must be provided. Performance results of the parallel Poisson solver and the parallel grid generators is shown in Tab. 1 for a toy Poisson problem

where $\Omega = \mathcal{S}^3$ (sphere). We show in Tab. 1 the scalability of the grid generator and the solver. The data in Tab. 1 is given for the grid generation (in column 3) and for one multigrid iteration (in column 5) with an Gauss-Seidel smoother. Tab. 1 shows excellent scalability with respect to the problem size M which is equivalent to say we can handle in the order of 10^{11} *macro particles* in a simulation with reasonable computing time. For this scaling study we use the Seaborg (IBM SP-3) computer at NERSC.

P	M	$T_g P/M$	T	TP/M
8	625,464	3.5e-3	3.1	3.9e-5
32	306,080	8.5e-3	0.78	8.1e-5
248	4,751,744	5.90e-3	1.2	6.2e-5
248	36,998,619	7.50e-3	7.7	5.1e-5
960	23,312,735	4.85e-3	4	1.64e-4
2025	405,242,845	6.60e-3	10.7	5.3e-5
4075	7,166,171,845	8.76e-3	160	9.9e-5

Table 1: Scalability of the parallel grid generator $T_g P/M$ and the Poisson solver showing also T, the time in seconds for one Multigrid step

Efficient Parallelization Automatic parallelization of a code can only be achieved if the code is implemented in a suitable language, for example C++ and MPI (Message Passing Interface) augmented with the concept of expression templates originally proposed in [4]. This ansatz allows C++ to achieve the same performance on vector and matrix expressions as with Fortran. Expression templates are also used in MAD9P, explained in the subsequent section.

PARTICLE TRACKING USING MAPS

MAD9P (**M**ethodical **A**ccelerator **D**esign version **9** - **p**arallel) is a general purpose parallel particle tracking program including three-dimensional space charge calculation [6].

Mathematical and Physical Model

MAD9P is based on the Vlasov-Maxwell equations. In this model, particle motion is governed by external fields and a mean-field approach for the space-charge fields. Particle collisions and radiation are neglected. The total Hamiltonian for a beam line element can be written as a sum of two parts, $\mathcal{H} = \mathcal{H}_1 + \mathcal{H}_2$, which correspond to the external and space charge contributions. A second-order integration algorithm (split operator) for a single step is then given by

$$\mathcal{M}_k(\tau) = \mathcal{M}_k^1(\tau/2)\, \mathcal{M}_k^2(\tau)\, \mathcal{M}_k^1(\tau/2) + \mathcal{O}(\tau^3) \qquad (8)$$

where τ denotes the step size, $\mathcal{M}_k^1$ is the map corresponding to $\mathcal{H}_1$ obtained by differential algebra methods from a

general relativistic Hamiltonian and $\mathcal{M}_k^2$ is the map corresponding to $\mathcal{H}_2$. $\mathcal{M}_k^2$ is obtained by discretizing the resulting Poisson problem on a rectangular mesh using Fourier techniques, as described in the second section of this paper. Open and periodic boundary conditions can be chosen. Once the physical elements are put together in an arbitrary way the elements are assumed to be perfectly aligned. To every beam element belongs a corresponding transfer map $\mathcal{M}_k^2$ which maps every initial condition ζ^i of the six dimensional phase space onto a final condition ζ^f by

$$\zeta^f = \mathcal{M}_k^2 \zeta^i. \qquad (9)$$

MAD9P derives $\mathcal{M}^2$ by a *Lie algebraic method*. The fact that the negative Poisson bracket of the Hamiltonian and the density function f is just the derivative of the density function with respect to the time leads to

$$f(t) = e^{-[t, \mathcal{H}_2]} \cdot f(0). \qquad (10)$$

$e^{[t, \mathcal{H}_2]}$ corresponds to $\mathcal{M}^2$ which can now be expanded in a Taylor series.

Next we show three applications with results, all the way from the 870 keV injection line to the first turns of Injector 2 including the buncher and collimation.

B870 Injection Line

The starting point for all B870 injection line calculations is a 4-dimensional transverse phase space distribution, which has been proven to be physically satisfactory in the daily operation of the beam line. The longitudinal dimensions are uniform in space and momenta. The initially DC beam is modeled by using a characteristic longitudinal beam length of $\beta\lambda$, where λ is the wave length of the RF. The double gap buncher is modeled by (analytic) sinusoidal momenta modulation of the beam. Fig. 2 shows the horizontal beam envelope (similar results are obtained in the vertical direction after fitting the 4-dimensional transverse distribution and a global space-charge neutralisation factor f_e using a stochastic fit algorithm based on Simulated An-
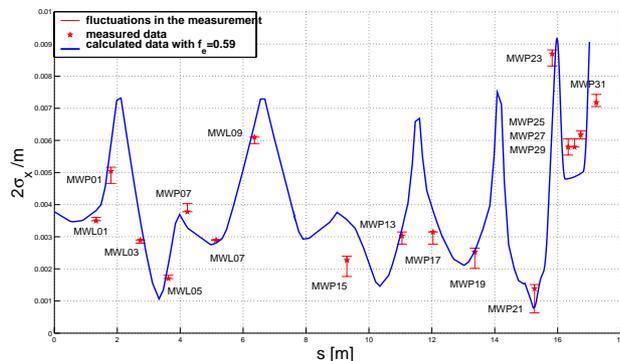
Figure 2: Color: Horizontal beam profiles

nealing. Define F as,

$$F = \sum_{n=1}^{\#monitors} \left(X_{mea}(s_n) - X_{sim}(s_n)\right)^2, \qquad (11)$$

this function is a measure of the degree of conformity between simulation and profile monitor measurements, where $X_{mea}(s_n)$ is a measured rms quantity at the position s_n along the beam line and $X_{sim}(s_n)$ is the corresponding calculated quantity obtained by MAD9P. The fitting procedure then minimizes F in Eq. (11). As shown in Fig. 2 we obtain good agreement between measurement and simulation. The space-charge neutralisation factor $f_e = 0.59$ obtained is in the expected range (for reference see [6]). The discrepancy in $MWP15$ is not fully understood. The deviations seen at $MWP25$ to $MWP31$ are related to the buncher and the high dispersive region in this part of the beam line. More detailed modeling is needed in order to minimize the gap between theory and observation.

Injector 2, Coasting Beam

A model of the Injector 2 lattice based on hard-edge elements is used for various coasting beam simulations. The 2D results of Adam [1], which predict a stable round distribution in horizontal- longitudinal configuration space has been verified (see Fig. 3) with the full three-dimensional model. The data shown in Fig. 3 are for $5\ MeV$ and $1\ mA$. The effect of the beam intensity on the development of the
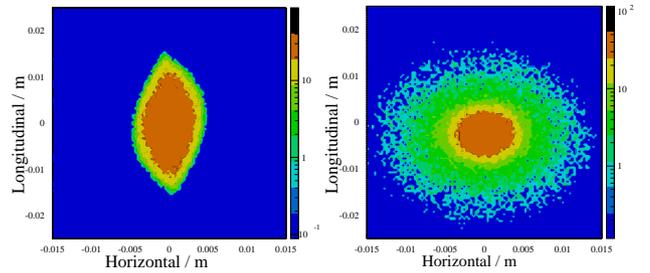
Figure 3: Color: Charge density in a.u.: Turn 1 and 60.

rms beam sizes in the horizontal and longitudinal directions is shown in Fig. 4 for 60 turns. The strong oscillations in the first few turns are due to an initial 'mismatch' of the beam. The fact that the rms beam size increases with increasing beam current strongly suggests that the matching of the incoming beam has to be adapted to the beam intensity and might be the key to a very fast development of the desired round and stable distribution. Those simulations suggests again, that the concept of an isochronous cyclotron is well suited for high intensity operation. More research is however needed in order to making predictions for the redesign of the B870 line, and to allow operation beyond the 2 mA presently achieved.

Injector 2 Including Collimation

To fix the non trivial initial conditions we start with one turn and the estimated particle distribution from [6]. After lengthy precision work on positioning the collimators and finetuning details of injections, we were able to simulate the very beginning of Injector 2 with satisfactory results. The amount of beam deposition on some collimators

Figure 4: Color: Horizontal and Longitudinal rms beam sizes at different intensities over 60 turns.

as well as the collimation process shown in Fig. 5 are well in agreement with observation. The z-axis is the direction of beam propagation and the x-axis points to the center of the cyclotron. Looking at Fig. 5 makes it clear that the

Figure 5: Color: Spatial particle density in a.u before KIP1 and after KIP2

bunch center rotates itself, the lower arm is expanding and the bunch has been collimated at the right place. The acceleration model in MAD9P is thus usable and promises detailed beam dynamic analysis in the near future.

TRACKING IN RF-STRUCTURES

To further increase the accuracy of the simulation we replace the thin lens approximation by full particle tracking in the RF cavity. Thus, both the electric field $\vec{E}(\vec{x}, t)$ and the magnet field $\vec{B}(\vec{x}, t)$ need to be known in the cavity. Therefore the standing electro-magnetic waves in the cavity with the lowest frequencies have to be determined. This amounts (after separation of time/space variables and after elimination of the magnetic field intensity) to solving the eigenvalue problem

$$
\begin{aligned}
\nabla \times \nabla \times \vec{e}(\vec{x}) &= \lambda\, \vec{e}(\vec{x}), & \vec{x} \in \Omega, \\
\nabla \cdot \vec{e}(\vec{x}) &= 0, & \vec{x} \in \Omega, \\
\vec{n} \times \vec{e} &= 0, & \vec{x} \in \partial\Omega,
\end{aligned}
\tag{12}
$$

where $\lambda = \omega^2/c^2$ and $\vec{e}$ is the normalized electric field intensity.

We use the weak formulation proposed by Kikuchi [7]

Find $(\lambda, \vec{e}, p) \in \mathbb{R} \times H_0(\mathrm{curl}; \Omega) \times H_0^1(\Omega)$ such that $\vec{e} \neq \vec{0}$ and

$$
\begin{aligned}
\text{(a)} \quad & (\nabla \times \vec{e}, \nabla \times \vec{\Psi}) + (\nabla p, \vec{\Psi}) = \lambda(\vec{e}, \vec{\Psi}) \\
& \hspace{4cm} \forall \vec{\Psi} \in H_0(\mathrm{curl}; \Omega) \\
\text{(b)} \quad & (\vec{e}, \nabla q) = 0, \hspace{2cm} \forall q \in H_0^1(\Omega)
\end{aligned}
\tag{13}
$$

where, p is a Lagrange multiplier.

Finite Element Discretisation

We discretise the field $\vec{e}$ in Eq. (13) by second order edge elements proposed by Nédélec [7] and the Lagrange multiplier p by second order node elements. This yields a large sparse constrained matrix eigenvalue problem of the form

$$
A\mathbf{x} = \lambda M \mathbf{x} \qquad C^T \mathbf{x} = \mathbf{0}.
\tag{14}
$$

where A is symmetric positive semi-definite and M is symmetric positive definite. C has full rank. The number of columns of C, i.e. the number of constraints, is about one eighth of the order of A and M! However, with this particular mixed finite element discretization we have

$$
C^T \mathbf{x} = \mathbf{0} \iff \mathbf{x} \perp_M \mathcal{N}(A).
\tag{15}
$$

This means that the eigenvalues of Eq. (14) are equal to the *positive* eigenvalues of

$$
A\mathbf{x} = \lambda M \mathbf{x}
\tag{16}
$$

and furthermore (16) has multiple zero eigenvalues with eigenspace $\mathcal{R}(Y)$ where $Y = M^{-1}C$ are the so–called ghost modes. Y can easily be obtained from the incidence matrix of the underlying finite element grid.

To prevent our algorithms from converging to the nullspace and computing the zero eigenvalue, we orthogonalise vectors against $\mathcal{R}(C)$ when necessary, i.e. we apply the M-orthogonal projector

$$
\mathbf{x} \leftarrow (I - YH^{-1}C^T)\mathbf{x}.
\tag{17}
$$

The matrix $H = C^T Y$ is the Poisson matrix corresponding to the second order nodal elements.

Eigenvalue Solvers

The eigenvalue problem (14) is solved numerically using our own implementation of the Jacobi-Davidson (JD) algorithm introduced by Sleijpen-van der Vorst [7]. Search space dimensions vary between 10 and 20 for computing 10 eigenpairs. The two main tasks in the algorithm are the correction equation ($\tilde{\rho} = \tilde{\mathbf{q}}^T A \tilde{\mathbf{q}}$)

$$
(I - M\tilde{\mathbf{q}}\tilde{\mathbf{q}}^T)(A - \tilde{\rho}M)(I - \tilde{\mathbf{q}}\tilde{\mathbf{q}}^T M)\mathbf{t} = -\mathbf{r},
\tag{18}
$$

subject to $\tilde{\mathbf{q}}^T M \mathbf{t} = 0$ that has to be solved in each iteration step and solving with H in Eq. (17) to enforce $C^T \mathbf{t} = \mathbf{0}$.

Hierarchical Basis Preconditioning

The preconditioner for Eq. (18) has the form

$$(I - M\tilde{\mathbf{q}}\tilde{\mathbf{q}}^T)K(I - \tilde{\mathbf{q}}\tilde{\mathbf{q}}^T M), \quad K \approx (A - \sigma M), \quad (19)$$

for some *fixed* shift σ. The hierarchical bases of the finite element spaces entail a 2×2 block structure of the involved matrices,

$$A^\sigma = A - \sigma M = \begin{bmatrix} A_{11}^\sigma & A_{12}^\sigma \\ A_{21}^\sigma & A_{22}^\sigma \end{bmatrix}. \quad (20)$$

Our preferred preconditioner approximately solves systems with A^σ by one step of block symmetric Gauss-Seidel iteration,

$$K = \begin{bmatrix} A_{11}^\sigma & \\ A_{21}^\sigma & \tilde{A}_{22}^\sigma \end{bmatrix} \begin{bmatrix} A_{11}^\sigma & \\ & \tilde{A}_{22}^\sigma \end{bmatrix}^{-1} \begin{bmatrix} A_{11}^\sigma & A_{12}^\sigma \\ & \tilde{A}_{22}^\sigma \end{bmatrix},$$

where $\tilde{A}_{22} \approx A_{22}$ denotes one step of point Jacobi iteration with A_{22}. Systems with A_{11} are solved approximately by the ML multilevel solver [8].

Parallelisation

The eigenfield solver is implemented in C++ using Trilinos [9], a collection of robust parallel solver algorithms for large-scale scientific applications. The matrices are distributed across the processors using the ParMETIS parallel graph partitioning and sparse matrix ordering package. Parallel execution mainly occurs in sparse matrix-vector multiplication with A, M and C and in the multilevel preconditioner.

Numerical Results
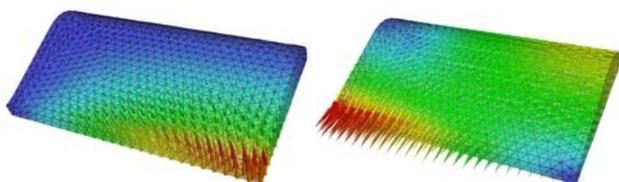
Figure 6: E- (left) and B-fields (right) of the fundamental mode in the new cavity of the PSI 590 MeV ring cylotron.

The calculated E- and B-fields in the cavity (cf. Fig. 6) will be soon integrated into the MAD9P tracking code.

BEAM-CAVITY INTERACTION

A mode-expansion method [10] is used for the representation of beam-excited fields in the ring cyclotron. For the determination of the mode amplitudes and phases it is required to calculate the parameters of zero beam-current trajectories from cyclotron injection to extraction. These particle motions are integrated by a fourth order Runge-Kutta algorithm based on a third order Taylor-expansion of the static magnetic fields.

The ESIL eigenmode solver in Omega3P allows to calculate Higher Order Modes (HOMs) of the entire cyclotron-structure as basis functions for the mode-expansion. A set of 280 eigenmodes with resonance frequencies close to harmonics of the beam-crossing frequency is found and 30 particularly critical modes are selected. Their eigenfields are interpolated onto a structured, cylindrical grid, located in the midplane.

Subsequent tracking of about 100000 macro-particles with a PIC-Needle model for fast space-charge corrections is used for the calculation of the distribution-propagation in the cyclotron. The parallelization with OpenMP yields typical execution-times of one minute per turn on a HP-superdome, using about eight CPUs. Comparing the beam-shapes of a simulation with consideration of HOMs to results from a HOMless simulation indicates that the effect of beam-excited fields onto the beam-quality is relatively small for the beam-currents of about 2mA.

REFERENCES

[1] S.R. Adam, "Methoden zur Berechnung der Longitudinalen Raumladungs Effekte in Isochronzyklotrons", PhD Thesis, ETH Zurich (1985).

[2] H.R. Fitze, "Developments at PSI (including new RF cavity)" Cyclotrons 2004.

[3] C. Pflaum, "Semi-unstructured grids, Computing" (2001) 67.

[4] T. Veldhuizen, "Expression templates", C++ Report (1995) 7.

[5] U. Trottenberg, C.W. Oosterlee, A. Schüller, "Multigrid", Academic Press (2001).

[6] A. Adelmann, "3D Simulations of Space Charge Effects in Particle Beams", PhD Thesis, ETH Zurich (2002).

[7] R. Geus, "The Jacobi–Davidson algorithm for solving large sparse symmetric eigenvalue problems", PhD Thesis No. 14734, ETH Zurich (2002).

[8] J. Hu and C. Tong and R.S. Tuminaro, "ML 2.0 Smoothed Aggregation User's Guide", Sandia National Laboratories (2000).

[9] M.A. Heroux and J.M. Willenbring, "Trilinos Users Guide", Sandia National Laboratories (2003).

[10] L. Stingelin et al., "Beam Cavity Interaction Simulation for the PSI Ring-Cyclotron", Cyclotrons 2004.