Differentiable self-consistent space-charge simulation for accelerator design

Ji Qiang[*]

Lawrence Berkeley National Laboratory, Berkeley, California 94720, USA

The nonlinear space-charge effects in a high intensity or high brightness accelerator can have a significant impact on the beam properties through the accelerator. These effects are included in the accelerator design via self-consistent multiparticle tracking simulations. In order to study the sensitivity of the final beam's properties with respect to the accelerator design parameters, one has to carry out the time-consuming space-charge simulation multiple times. In this paper, we propose a differentiable self-consistent space-charge simulation model that enables the study of such sensitivity through only one simulation. This model differs from previous applications of the truncated power series algebra by connecting the beam properties directly with the accelerator design parameters in the presence of collective space-charge effects so that the local derivative of the beam properties with respect to the design parameters can be computed during the simulation. Such a model can also be used with gradient-based numerical optimizers for accelerator design optimizations including the self-consistent space-charge effects.

I. INTRODUCTION

The nonlinear space-charge effects from the Coulomb interaction inside a charged particle beam can have a significant impact on the beam transport through an accelerator by causing beam emittance growth, halo formation, and even particle losses. These effects are normally studied in the accelerator design via self-consistent simulations. To simulate the space-charge effects self-consistently, multiparticle tracking with a particle-in-cell method has been widely employed in the accelerator community [1–11]. However, none of these codes directly calculates the derivatives of the beam property with respect to the accelerator machine parameters.

The derivatives of the beam property with respect to the accelerator machine parameters are important in the accelerator design. These derivatives provide quantitative measurements of the sensitivity of the final beam property with respect to the machine parameters. Such sensitivities can be used to set the tolerance limits of the corresponding machine parameters. In a typical machine design, the sensitivity can be obtained by running the space-charge simulation multiple times, each time with a small change in a single machine parameter. A numerical scheme such as the finite difference method is used to calculate the

derivative of the beam property with respect to that parameter. In an accelerator, there can be hundreds and thousands machine parameters. To compute the derivatives with respect to all machine parameters using the finite difference method will involve hundreds and thousands self-consistent space-charge simulations and can be very time consuming. In this paper, we propose a differentiable self-consistent space-charge simulation model. The derivatives of the final beam property with respect to the entire machine parameters can be obtained through a single simulation. Such a differentiable space-charge simulation can also be integrated into a gradient-based optimizer for accelerator design optimization.

The differentiable simulator is a simulator that can automatically compute derivatives of the simulation result with respect to its input parameters. This can be done through automatic differentiation [12] that has been widely used in the artificial intelligent/machine learning (AI/ML) community to train a neural network through gradient-based optimization method. A number of AI/ML frameworks such as PyTorch [13] and TensorFlow [14] include this capability. Recently, a differentiable model was developed to optimize magnet design by implementing the nonparametric version of the Preisach model output using the PyTorch framework [15]. In this paper, instead of using the automatic differentiation framework of the AI/ML community, we proposed a differentiable self-consistent space-charge simulation model using the truncated power series algebra that was developed in the accelerator community. This method can also be implemented in many other available self-consistent space-charge codes for high intensity, high brightness accelerator design study.

[*]jqiang@lbl.gov

The truncated power series algebra (TPSA) was first introduced by Berz in 1989 for accelerator applications [16]. Since then, it has been used to calculate transfer maps of beamline elements in several optics codes [17–20]. It was also used to calculate the space-charge potential/fields in the fast multipole method [21–23] and to solve the Poisson's equation [24]. Recently, it was also used to extract transfer maps in the presence of space-charge effects [25–27]. However, so far, these applications of the TPSA have no direct connections to the control parameters in the accelerator design. In this study, we present a new application of the TPSA to the accelerator design in the self-consistent space-charge simulation. In this differentiable space-charge model, the beam properties such as emittance are connected to the accelerator control parameters. The novelty of this model is to combine the TPSA and the self-consistent space-charge algorithm to compute the local derivative of beam parameters with respect to beamline control variables. This enables to study the sensitivity of the final beam property with respect to these control variables through only one simulation, which is different from the other uncertainty quantification methods such as the surrogate model method in which a number of simulations have to be used to train the model [28].

The organization of this paper is as follows: after the introduction, we introduce the truncated power series algebra in Sec. II, present a differentiable space-charge model in Sec. III, two application examples in Sec. IV, and draw conclusions in Sec. V.

II. TRUNCATED POWER SERIES ALGEBRA

The truncated power series algebra (TPSA) is an effective tool to calculate derivatives of a function with respect to its variables using an algebraic method. Consider the Taylor series approximation of a one-dimensional function $f(x)$ at a point x_0,

$$f(x) = f(x_0) + (x - x_0)f'(x_0) + \frac{1}{2!}(x - x_0)^2 f''(x_0) + \cdots$$
$$+ \frac{1}{N!}(x - x_0)^N f^{(N)}(x_0) \tag{1}$$

the derivatives in the above equation can be calculated using a numerical finite difference method, for example,

$$f'(x_0) = \frac{f(x_0 + \delta) - f(x_0)}{\delta} + O(\delta). \tag{2}$$

Such a way to calculate the derivative introduces numerical errors and requires multiple function evaluations for a multivariable function. In the truncated power series algebra method, the derivative up to Nth order can be regarded as a point in a function space spanned by the bases:

$$1, (x - x_0), \frac{1}{2!}(x - x_0)^2, \dots, \frac{1}{N!}(x - x_0)^N \tag{3}$$

These derivatives can be represented as a vector:

$$Df_{x_0} = [f(x_0), f'(x_0), f''(x_0), \dots, f^{(N)}(x_0)]. \tag{4}$$

Such a vector is also called a TPSA variable. For example, for a constant c, its derivative representation is $Dc = [c, 0, 0, \dots, 0]$, and for a variable x, $Dx = [x, 1, 0, \dots, 0]$. By using the above derivative vector representation, the derivatives of function with respect to its variable can be written as the function of that derivative vector, i.e.,

$$Df_x = f(Dx). \tag{5}$$

The computing of the derivatives of a function becomes the algebraic function evaluations.

The evaluation of the derivative vector inside a function can be broken down as the operations of addition and multiplication. Given two derivative vectors $Df_{x_0} = [f(x_0), f'(x_0), f''(x_0), \dots, f^{(N)}(x_0)] = [a_0, a_1, a_2, \dots, a_N]$ and $Df_{x_1} = [f(x_1), f'(x_1), f''(x_1), \dots, f^{(N)}(x_1)] = [b_0, b_1, b_2, \dots, b_N]$, the sum of two vectors will be

$$Df_{x_0} + Df_{x_1} = [a_0 + b_0, a_1 + b_1, a_2 + b_2, \dots, a_N + b_N] \tag{6}$$

The multiplication of these two vectors will be

$$Df_{x_0} \times Df_{x_1} = [a_0 b_0, a_0 b_1 + a_1 b_0, a_0 b_2 + 2a_1 b_1$$
$$+ a_2 b_0, \dots, c_N], \tag{7}$$

where $c_N = \sum_{k=0}^{N} \frac{N!}{k!(N-k)!} a_k b_{N-k}$. Using the above rules of addition and multiplication, the operation of a derivative vector inside a function can be calculated algebraically. For example, the reciprocal of a derivative vector $1/Df_{x_0}$ can be calculated as

$$Df_{x_0}^{-1} = [a_0, a_1, a_2, \dots, a_N]^{-1}$$
$$= \left[\frac{1}{a_0}, -\frac{a_1}{a_0^2}, \frac{2a_1^2}{a_0^3} - \frac{a_2}{a_0^2}, \cdots\right] \tag{8}$$

For a concrete example, $f(x) = \frac{1}{1+x+x^2}$, one can use the above derivative vector operations to obtain the first and the second derivative of this function at $x = 1$. That is, $x = 1$, $Dx = [1, 1, 0]$, and

$$Df_1 = f(D1)$$
$$= \frac{1}{1 + [1, 1, 0] + [1, 1, 0]^2}$$
$$= \left[\frac{1}{3}, -\frac{1}{3}, \frac{4}{9}\right], \tag{9}$$

This yields the first derivative $f'(1) = -\frac{1}{3}$ and the second derivative $f''(1) = \frac{4}{9}$. The truncated power series algebra changes the calculation of the derivatives of a function with respect to its individual variable into the evaluation of a function of derivative vector, i.e., a function of TPSA variable.

The above single variable function example can be extended to a multiple variable function with the more complicated multiplication rule for the derivative vector [29]. The truncated power series algebra libraries have been developed to handle some general special functions such as the exponential function and the trigonometry function [30,31].

III. DIFFERENTIABLE SELF-CONSISTENT SPACE-CHARGE MODEL

The above truncated power series algebra is used to develop a differentiable self-consistent space-charge model using TPSA variables. The self-consistent space-charge simulation of a system of N_p particles is done by solving the following Hamilton equations:

$$\frac{d\mathbf{r}_i}{ds} = \frac{\partial H}{\partial \mathbf{p}_i} \tag{10}$$

$$\frac{d\mathbf{p}_i}{ds} = -\frac{\partial H}{\partial \mathbf{r}_i}, \tag{11}$$

where $H(\mathbf{r}_1, \mathbf{p}_1, \mathbf{r}_2, \mathbf{p}_2, \dots, s)$ denotes the Hamiltonian of the system, and $\mathbf{r}_i$ and $\mathbf{p}_i$ denote $6N_p$ or $4N_p$-vector canonical coordinates and momenta of particle i, respectively. Let ζ denote a 6N or 4N vector of coordinates, the above Hamilton's equation can be rewritten as

$$\frac{d\zeta}{ds} = -[H, \zeta], \tag{12}$$

where [,] is the Poisson bracket. A formal solution for the above equation after a single step τ can be written as

$$\zeta(\tau) = \exp[-\tau(:H:)]\zeta(0) \tag{13}$$

Here, we have defined a differential operator $:H:$ as $:H:g = [H, g]$, for arbitrary function g. For a coasting beam, the Hamiltonian can be written as $H = H_1 + H_2$, where

$$H_1 = \sum_{i=1}^{N_p} \mathbf{p}_i^2/2 + \sum_{i=1}^{N_p} qA_z(\mathbf{r}_i) \tag{14}$$

where A_z denotes the longitudinal vector potential associated with the external focusing fields and

$$H_2 = \frac{K}{4} \sum_{i=1}^{N_p} \sum_{j=1}^{N_p} \varphi(\mathbf{r}_i, \mathbf{r}_j) \tag{15}$$

where $K = qI/(2\pi\epsilon_0 p_0 v_0^2 \gamma_0^2)$ is the generalized perveance, q is the charge of particle, I is the beam current, ϵ_0 is the dielectric constant in vacuum, p_0 is the momentum of the reference particle, v_0 is the speed of the reference particle, γ_0 is the relativistic factor of the reference particle, and φ is the space charge Coulomb interaction potential. In this Hamiltonian, the effects of the direct electric potential and the longitudinal vector potential are combined together. For a Hamiltonian that can be written as a sum of two terms $H = H_1 + H_2$, an approximate solution to the above formal solution can be written as

$$\begin{aligned}
\zeta(\tau) &= \exp[-\tau(:H_1: + :H_2:)]\zeta(0) \\
&= \exp\left(-\frac{1}{2}\tau:H_1:\right) \exp(-\tau:H_2:) \\
&\quad \times \exp\left(-\frac{1}{2}\tau:H_1:\right)\zeta(0) + O(\tau^3).
\end{aligned} \tag{16}$$

Let $\exp(-\frac{1}{2}\tau:H_1:)$ define a transfer map $\mathcal{M}_1$ and $\exp(-\tau:H_2:)$ a transfer map $\mathcal{M}_2$, for a single step, the above splitting results in a second order numerical integrator for the original Hamilton's equation as

$$\begin{aligned}
\zeta(\tau) &= \mathcal{M}(\tau)\zeta(0) \\
&= \mathcal{M}_1(\tau/2)\mathcal{M}_2(\tau)\mathcal{M}_1(\tau/2)\zeta(0) + O(\tau^3)
\end{aligned} \tag{17}$$

For the external focusing with quadrupole magnets, the single step transfer map $\mathcal{M}_1$ in the focusing plane can be written as

$$\mathcal{M}_1(\tau) = \begin{pmatrix} \cos(\sqrt{k}\tau) & \frac{1}{\sqrt{k}}\sin(\sqrt{k}\tau) \\ -\sqrt{k}\sin(\sqrt{k}\tau) & \cos(\sqrt{k}\tau) \end{pmatrix} \tag{18}$$

and in the defocusing plane as

$$\mathcal{M}_1(\tau) = \begin{pmatrix} \cosh(\sqrt{k}\tau) & \frac{1}{\sqrt{k}}\sinh(\sqrt{k}\tau) \\ -\sqrt{k}\sinh(\sqrt{k}\tau) & \cosh(\sqrt{k}\tau) \end{pmatrix}, \tag{19}$$

where k is the normalized focusing strength $k = qg/p_0$ and g is the magnetic field gradient. For the space-charge Hamiltonian $H_2(\mathbf{r})$, the single step transfer map $\mathcal{M}_2$ can be written as

$$\mathbf{r}_i(\tau) = \mathbf{r}_i(0) \tag{20}$$

$$\mathbf{p}_i(\tau) = \mathbf{p}_i(0) - \frac{\partial H_2(\mathbf{r})}{\partial \mathbf{r}_i}\tau. \tag{21}$$

The electric Coulomb potential in the Hamiltonian H_2 can be obtained from the solution of the Poisson equation. In the following, we assume that the coasting beam is inside a rectangular perfectly conducting pipe. In this case, the two-dimensional Poisson's equation can be written as

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = -4\pi\rho, \tag{22}$$

where ϕ is the electric potential and ρ is the particle density distribution of the beam.

The boundary conditions for the electric potential inside the rectangular perfectly conducting pipe are as follows:

$$\phi(x = 0, y) = 0 \tag{23}$$

$$\phi(x = a, y) = 0 \tag{24}$$

$$\phi(x, y = 0) = 0 \tag{25}$$

$$\phi(x, y = b) = 0, \tag{26}$$

where a is the horizontal width of the pipe and b is the vertical width of the pipe.

Given the boundary conditions in Eqs. (23)–(26), using a Galerkin spectral approximation method, one obtains the space-charge Hamiltonian H_2 as [32]:

$$H_2 = 4\pi \frac{K}{ab} \frac{1}{N_P} \sum_{i=1}^{N_p} \sum_{j=1}^{N_p} \sum_{l=1}^{N_l} \sum_{m=1}^{N_m} \frac{1}{\gamma_{lm}^2} \sin(\alpha_l x_j)$$
$$\times \sin(\beta_m y_j) \sin(\alpha_l x_i) \sin(\beta_m y_i) \tag{27}$$

The resultant one-step symplectic transfer map $\mathcal{M}_2$ of the particle i with this Hamiltonian is given as

$$p_{xi}(\tau) = p_{xi}(0) - \tau \frac{K}{2} \sum_{l=1}^{N_l} \sum_{m=1}^{N_m} \phi^{lm} \alpha_l \cos(\alpha_l x_i) \sin(\beta_m y_i)$$

$$p_{yi}(\tau) = p_{yi}(0) - \tau \frac{K}{2} \sum_{l=1}^{N_l} \sum_{m=1}^{N_m} \phi^{lm} \beta_m \sin(\alpha_l x_i) \cos(\beta_m y_i), \tag{28}$$

where the space-charge potential in the spectral domain is given as

$$\phi^{lm} = 4\pi \frac{4}{ab} \frac{1}{N_P} \sum_{j=1}^{N_p} \frac{1}{\gamma_{lm}^2} \sin(\alpha_l x_j) \sin(\beta_m y_j). \tag{29}$$

Here, both p_{xi} and p_{yi} are normalized by the reference particle momentum p_0.

In the differentiable space-charge simulation, the above space-charge model will be rewritten using TPSA variables.

The phase space coordinates $\mathbf{r}_i$ and $\mathbf{p}_i$ of the particle i will be replaced by the corresponding TPSA variables $D\mathbf{r}_i$ and $D\mathbf{p}_i$ defined in the last section. The potential ϕ^{lm} is replaced by $D\phi^{lm}$ of the TPSA variable. The momentum updates after a single step due to the space-charge effects are given by

$$Dp_{xi}(\tau) = Dp_{xi}(0)$$
$$- D\tau \frac{K}{2} \sum_{l=1}^{N_l} \sum_{m=1}^{N_m} D\phi^{lm} \alpha_l \cos(\alpha_l Dx_i) \sin(\beta_m Dy_i)$$

$$Dp_{yi}(\tau) = Dp_{yi}(0)$$
$$- D\tau \frac{K}{2} \sum_{l=1}^{N_l} \sum_{m=1}^{N_m} D\phi^{lm} \beta_m \sin(\alpha_l Dx_i) \cos(\beta_m Dy_i), \tag{30}$$

where the space-charge potential in TPSA variable is

$$D\phi^{lm} = 4\pi \frac{4}{ab} \frac{1}{N_P} \sum_{j=1}^{N_p} \frac{1}{\gamma_{lm}^2} \sin(\alpha_l Dx_j) \sin(\beta_m Dy_j) \tag{31}$$

The map corresponding to the external quadrupole field can also be written in TPSA variable as

$$\mathcal{M}_1(\tau) = \begin{pmatrix} \cos(\sqrt{Dk}D\tau) & \frac{1}{\sqrt{Dk}} \sin(\sqrt{Dk}D\tau) \\ -\sqrt{Dk} \sin(\sqrt{Dk}D\tau) & \cos(\sqrt{Dk}D\tau) \end{pmatrix} \tag{32}$$

where Dk is the TPSA variable of the quadrupole focusing strength, and $D\tau$ is the TPSA variable of the step size that is the quadrupole length divided by the number of steps. A similar expression can be written in the defocusing plane of the quadrupole.

The charged particle beam initial distribution parameters, beam energy, and current can also be written using TPSA variables if needed. The final beam properties such as emittances are defined using the TPSA variables. The horizontal emittance $D\epsilon_x$ is given as

$$D\epsilon_x = \sqrt{D\langle x^2 \rangle D\langle p_x^2 \rangle - (D\langle x p_x \rangle)^2}, \tag{33}$$

where

$$D\langle x^2 \rangle = \sum_{i=1}^{N_p} (Dx_i)^2 \tag{34}$$

$$D\langle p_x^2 \rangle = \sum_{i=1}^{N_p} (Dp_{xi})^2 \tag{35}$$

$$D\langle xp_x \rangle = \sum_{i=1}^{N_p} Dx_i Dp_{xi}. \qquad (36)$$

The vertical emittance has a similar expression with x replaced by y.

IV. APPLICATION EXAMPLES

In the following, we will use two application examples to illustrate the above differentiable self-consistent space-charge simulation model: one for parameter sensitivity study and the other one for design optimization study.

In the first illustrative application, we studied the sensitivity of the final emittances of a 1-GeV coasting proton beam transporting through a transverse focusing FODO lattice inside a rectangular perfectly conducting pipe with respect to the lattice parameters and the initial beam parameters. A schematic plot of the FODO lattice is shown in Fig. 1. It consists of a drift (D1) of 0.2 m, a quadrupole (Q1) of 0.1 m, and a focusing strength of 29.6/m^2 for transverse focusing, another drift (D2) of 0.4 m, another quadrupole (Q2) of the same length as the first quadrupole but the opposite sign of focusing, and another drift (D3) of length 0.2 m. The rectangular pipe has an aperture size of 13 by 13 mm. The zero current phase advance of the FODO lattice is 87.0°. The current of the proton beam is 200 A with 1 mm mrad normalized emittance, which results in a depressed phase advance of 63.1°. The initial distribution is assumed to be a four-dimensional Gaussian distribution given by

$$f(x, p_x, y, p_y) \propto \exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + 2xp_x\frac{\mu_{xp_x}}{\sigma_x\sigma_{p_x}} + \frac{p_x^2}{\sigma_{p_x}^2}\right)\right]$$
$$\times \exp\left[-\frac{1}{2}\left(\frac{y^2}{\sigma_y^2} + 2yp_y\frac{\mu_{yp_y}}{\sigma_y\sigma_{p_y}} + \frac{p_y^2}{\sigma_{p_y}^2}\right)\right] \qquad (37)$$

The parameters in the above distribution are chosen to be rms matched through the FODO lattice including the space-charge effects.

We first checked the sensitivity of the final emittances with respect to seven FODO lattice parameters, i.e., lengths of the drift and quadrupole elements, and focusing strengths of two quadrupole elements. In this example, we used 5000 macroparticles and 12×12 spectral modes in
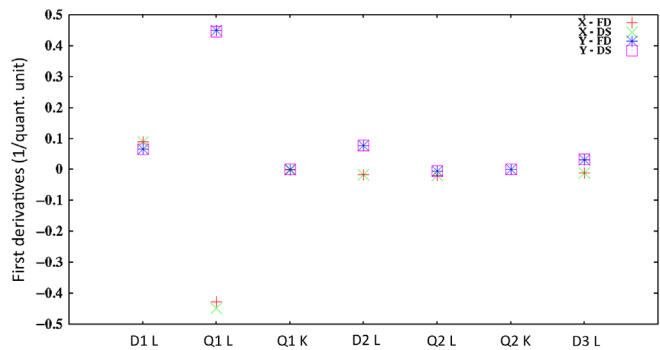
FIG. 2. Derivatives of final horizontal and vertical emittances with respect to seven lattice parameters from the single differentiable self-consistent space-charge simulation and from the finite difference approximation to the first derivative.

the space-charge simulation. The sensitivity is measured by the first derivatives of the final emittances (normalized by initial emittance) with respect to these lattice parameters. Figure 2 shows the sensitivities of the final horizontal and vertical emittances with respect to the seven lattice parameters from the single differentiable self-consistent space-charge simulation and from the numerical finite difference approximation to the first derivative using eight space-charge simulations. Here, the seven lattice parameters are drift one length (D1 L), quadrupole one length (Q1 L), quadrupole one focusing strength (Q1 k), drift two length (D2 L), quadrupole two length (Q2 L), quadrupole two strength (Q2 k), drift three length (D3 L). It is seen that the first derivatives calculated from the differentiable space-charge model and from the finite difference approximation agree with each other quite well. This provides a verification of the differentiable self-consistent space-charge simulation model. From this figure, one can see that the final emittance is much more sensitive to the length of the first quadrupole than the other lattice parameters such as drift lengths and quadrupole strengths. The change in the first quadrupole length causes significant beam envelope variation and results in a large final emittance change.

Next, we checked the sensitivity of the final proton beam emittance with respect to the initial beam parameters. These beam parameters are proton beam energy (Eng.), beam current (Cur.), and six beam distribution parameters (σ_x, σ_{p_x}, μ_{xp_x}, σ_y, σ_{p_y}, and μ_{yp_y}). These parameters were represented using TPSA variables. A number of macroparticle coordinates (also in TPSA variables) were generated from the regular sampling method using the beam distribution parameters. The quadrupole magnetic field gradient was used instead of the focusing strength since the latter includes the proton beam energy. The change in the beam energy affects both the transverse focusing and the space-charge effects. Figure 3 shows the first derivatives of the final emittances with respect to the above eight parameters. It is seen that the final emittances are more sensitive to the initial distribution parameter σs.
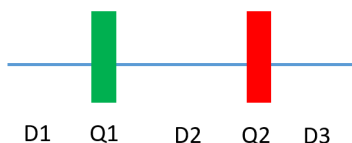

FIG. 1. Schematic plot of a FODO lattice in the first application example.
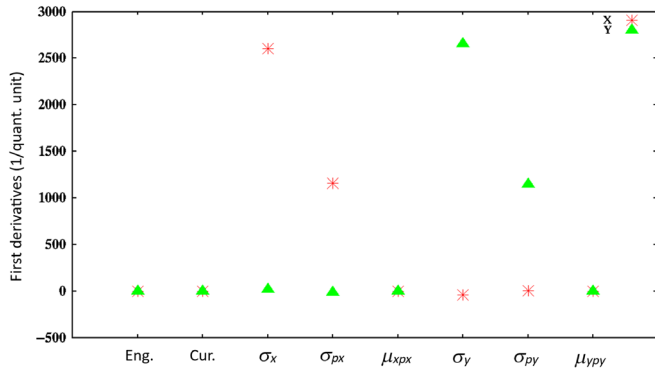
FIG. 3. Derivatives of final horizontal and vertical emittance with respect to eight beam parameters.

The perturbation of these parameters affects the matching of the initial distribution to the FODO lattice and can cause significant emittance growth with a mismatched beam [33,34].

The differentiable self-consistent space-charge simulation through the accelerator lattice produces the final beam properties and their derivatives with respect to the lattice control parameters at the exit of the accelerator as shown in the first example. These derivatives can be used in a gradient-based parameter optimizer for accelerator lattice control parameter optimization. In the following example, we integrated the differentiable self-consistent space-charge simulation model into a conjugate gradient optimizer to attain the quadrupole strengths inside a matching section in front of a periodic FODO lattice.

FIG. 4. Schematic plot of a FODO lattice used in the second application example. The four quadrupoles inside the dashed line box were used to match the initial distribution to the periodic FODO lattice.

A schematic plot of the matching section lattice and the periodic FODO lattice is shown in Fig. 4. The first four quadrupoles in the figure were used to match an initial distribution to the given Twiss parameters at the entrance to the periodic FODO lattice. The Polak-Ribiere conjugate gradient optimization method [35] was used to minimize the objective function that is defined as follows:

$$f(\mathbf{k}) = \frac{(\beta_x(\mathbf{k}) - \beta_{xt})^2}{\beta_{xt}^2} + (\alpha_x(\mathbf{k}) - \alpha_{xt})^2$$
$$+ \frac{(\beta_y(\mathbf{k}) - \beta_{yt})^2}{\beta_{yt}^2} + (\alpha_y(\mathbf{k}) - \alpha_{yt})^2, \quad (38)$$

where $\mathbf{k}$ is a set of control variables, $\alpha_{xt}, \beta_{xt}, \alpha_{yt},$ and β_{yt} are the target Twiss parameters at the entrance to the periodic lattice, and the α_x, β_x, α_y, and β_y are the beam Twiss parameters calculated from the self-consistent space-charge simulation. These calculated beam Twiss parameters depend on the focusing strengths of the quadrupoles inside the matching section. These strengths are control variables in the above objective function. Using the above differentiable space-charge simulation model, the first derivatives of the objective function with respect to the four control variables were obtained in addition to the objective function value. These derivatives were used to construct a conjugate direction of the gradient direction to guide the search for the minimum solution.

Figure 5 shows the proton beam transverse rms size evolution through the above FODO lattice without the quadrupole matching and with the quadrupole matching including the space-charge effects. Here, the quadrupole strengths inside the matching section without the matching were set based on the zero current–matched solution. It is seen that with a 200-A beam, the space-charge effects are significant so that the initial zero current–matched quadrupole strengths no longer produce a matched rms evolution inside the periodic lattice. After reoptimizing the four quadrupole strengths including the space-charge effects
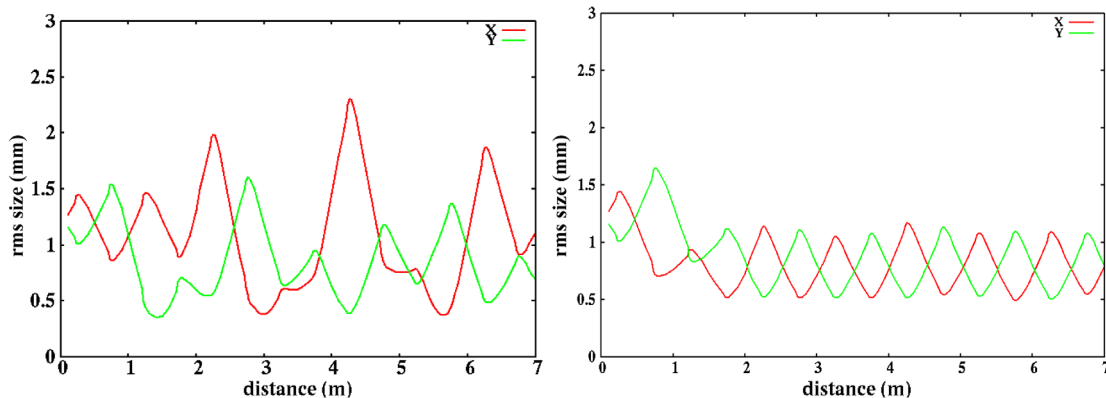

FIG. 5. Transverse rms size evolution without the quadrupole matching (left) and with the quadrupole matching including the space-charge effects (right) through the FODO lattice.

through the self-consistent simulations, the rms evolution inside the periodic lattice becomes well matched and results in much less emittance growth (less than 10%) than that from the mismatched quadrupole setting (greater than 50%).

V. CONCLUSIONS

The self-consistent space-charge simulation is an important part in the high intensity, high brightness accelerator design. In this paper, we proposed a differentiable self-consistent space-charge model that can be used to efficiently study the sensitivity of the final beam properties with respect to the accelerator design parameters. Using the differentiable self-consistent space-charge model, only one simulation is needed to attain the derivatives of the final beam properties with respect to all accelerator design parameters instead of multiple simulations of the conventional space-charge model. The resultant first derivatives measure the sensitivity of the final beam properties with respect to those design parameters. Some highly sensitive machine parameters can be quickly identified after one differentiable self-consistent space-charge simulation. The accuracy of the sensitivity measurements computed using the TPSA method in the differentiable space-charge simulation depends on the accuracy of the simulation model. The more accurate the simulation model, the more accurate these measurements will be.

As an illustration, we presented two application examples. One example computed the sensitivities of the final proton beam emittances with respect to seven lattice parameters or eight beam parameters through a single differentiable self-consistent simulation. The second example showed that the derivatives of the objective function with respect to the quadrupole strengths inside a matching section from the differentiable self-consistent space-charge simulation were used in a conjugate gradient optimizer to attain the space-charge matched solution to a periodic FODO lattice. The success of these two examples shows that the differentiable self-consistent space-charge simulation model can be a useful tool in the accelerator design.

In this study, we used a spectral solver as an illustration of a differentiable space-charge model. In general, some other space-charge solvers such as fast multipole solver or Green function solver can also be used as the differentiable space-charge model as long as the space-charge fields from these solvers are represented using TPSA variables. Furthermore, this can be generalized beyond the space-charge effects. The other collective effects such as beam-beam effects can also be represented using TPSA variables. Together with the particle coordinates and accelerator machine parameters that are represented using the TPSA variables, one can develop a general differentiable simulation tool that includes a variety of collective effects for accelerator design. Such a tool will be useful to study the beam quality sensitivity to the accelerator machine parameters subject to the collective effects through only one simulation.

In the present study, the computational speed of the above differentiable space-charge solver is slow compared with the conventional space-charge solver due to the overhead associated with the computation using TPSA variables in the available TPSA library. From the discussion with the author of the TPSA library used in this study, the computational speed can be substantially improved if only the first-order derivative is needed [36]. A number of performance optimization strategies can be employed to improve the computational efficiency of the library. This will be pursued in future study by working with the TPSA library developers.

[1] A. Friedman, D. P. Grote, and I. Haber, Phys. Fluids B **4,** 2203 (1992).

[2] H. Takeda and J. H. Billen, Recent developments of the accelerator design code PARMILA, in *Proceedings of the 19th International Linear Accelerators Conference, Chicago, IL, 1998* (NTIS, Springfield, VA, 1998), p. 156.

[3] S. Machida and M. Ikegami, AIP Conf. Proc. **448,** 73 (1998).

[4] F. W. Jones and H. O. Schoenauer, in *Proceedings of the 18th Particle Accelerator Conference, New York, 1999* (IEEE, New York, 1999), p. 2933.

[5] J. Qiang, R. D. Ryne, S. Habib, and V. Decyk, J. Comput. Phys. **163,** 434 (2000).

[6] P. N. Ostroumov and K. W. Shepard, Phys. Rev. ST Accel. Beams **3,** 030101 (2000).

[7] J. D. Galambos, S. Danilov, D. Jeon, J. A. Holmes, and D. K. Olsen, F. Neri, and M. Plum, Phys. Rev. ST Accel. Beams **3,** 034201 (2000).

[8] G. Franchetti, I. Hofmann, M. Giovannozzi, M. Martini, and E. Metral, Phys. Rev. ST Accel. Beams **6,** 124201 (2003).

[9] J. Qiang, S. Lidia, R. D. Ryne, and C. Limborg-Deprey, Phys. Rev. ST Accel. Beams **9,** 044204 (2006).

[10] J. Amundson, P. Spentzouris, J. Qiang, and R. Ryne, J. Comput. Phys. **211,** 229 (2006).

[11] http://amas.web.psi.ch/docs/opal/opal_user_guide.pdf.

[12] C. C. Margossian, A review of automatic differentiation and its efficient implementation, WIREs **9,** e1305 (2019).

[13] A. Paszke *et al.*, PyTorch: An imperative style, high-performance deep learning library, in *Advances in Neural Information Processing Systems*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alche-Buc, E. Fox, and R. Garnett (Curran Associates, Inc., Red Hook, New York, 2019), Vol. 32, pp. 8024–8035.

[14] Martin Abadi *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems, https://www.tensorflow.org/ (2015).

[15] R. Roussel *et al.*, Differentiable Preisach modeling for characterization and optimization of accelerator systems with hysteresis, arXiv:2202.07747.

[16] M. Berz, Part. Accel. **24**, 109 (1989).

[17] Y. T. Yan, Zlib: A numerical library for optimal design of truncated power series algebra and map parameterization routines, Report No. SLAC-PUB-7363, 1996.

[18] M. Berz, *Modern Map Methods in Particle Beam Physics* (Academic Press, New York, 1999), http://bt.pa.msu.edu/pub.

[19] E. Forest, F. Schmidt, and E. McIntosh, Introduction to the Polymorphic Tracking Code, Reports No. CERN-SL-2002-044 (AP), KEK-Report 2002-3, 2002.

[20] K. Makino and M. Berz, Nucl. Instrum. Methods Phys. Res., Sect. A **558**, 346 (2006).

[21] H. Zhang and M. Berz, Nucl. Instrum. Methods Phys. Res., Sect. A **645**, 338 (2011).

[22] E. Nissen, Differential algebraic methods for space charge modeling and applications to the University of Maryland electron ring, Ph.D. Thesis, Northern Illinois University, 2011.

[23] H. Zhang, The fast multipole method in the differential algebra framework for the calculation of 3D space charge fields', Ph.D. Thesis, Michigan State University, 2013.

[24] B. Erdelyi, E. Nissen, and S. Manikonda, Commun. Comput. Phys. **17**, 47 (2015).

[25] E. Nissen and B. Erdelyi, Method to extract transfer maps in the presence of space-charge in charged particle beams, in *Proceedings of the International Particle Accelerator Conference, Kyoto, Japan* (ICR, Kyoto, 2010), p. 1967.

[26] A. Gee and E. Erdelyi, Space charge map extraction and analysis in a differential algebraic framework, in *Proceedings of the 54th ICFA Advanced Beam Dynamics Workshop on High-Intensity, East-Lansing, MI* (JACoW, Geneva, Switzerland, 2014).

[27] J. Qiang, Extraction of transfer map including self-consistent space-charge effects in a symplectic tracking code, in *Proceedings of 4th ICFA Mini Space-Charge Workshop* (JACoW, Geneva, Switzerland, 2019).

[28] A. Adelmann, SIAM-ASA J. Uncertainty Quantif. **7**, 383 (2019)

[29] A. Chao, Lecture notes on topics in accelerator physics, Report No. SLAC-PUC-9574.

[30] Y. Hao, https://github.com/YueHao/PyTPSA.

[31] H. Zhang, https://github.com/zhanghe9704/tpsa.

[32] J. Qiang, Phys. Rev. Accel. Beams **21**, 054201 (2018).

[33] J. Qiang, P. L. Colestock, D. Gilpatrick, H. V. Smith, T. P. Wangler, and M. E. Schulze, Phys. Rev. ST Accel. Beams **5**, 124201 (2002).

[34] C. K. Allen *et al.*, Phys. Rev. Lett. **89**, 214802 (2002).

[35] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in Fortran: The Art of Scientific Computing* (Cambridge University Press, Cambridge, England, 1992).

[36] Private communication with Dr. Yue Hao, 2022.