Start-up and saturation in self-amplified spontaneous emission free-electron lasers using a time-independent analysis

Vinit Kumar and Srinivas Krishnagopal
Beam Physics and FEL Laboratory, Centre for Advanced Technology, Indore 452013, India
(Received 23 May 2001; published 18 December 2001)

Numerical simulation of self-amplified spontaneous emission (SASE) in free-electron lasers (FELs) is typically performed using time-dependent computer codes, which take large CPU time and require large memory. Recently, Yu [Phys. Rev. E **58**, 4991 (1998)] has shown that one can even use a time-independent code for this purpose (where the requirement on CPU time and memory is significantly reduced) by modifying it to include multiple phase-space buckets and using a scaling relation between the output power and the number of simulation particles, which is valid only in the linear regime. In this paper, we take a fresh look at the problem and show that incorporating multiple buckets in TDA3D is not needed to simulate the SASE process. We give a new interpretation of time-independent simulations of the SASE process and present detailed justification for using a single-frequency steady-state simulation code for the study of evolution of shot noise. We further extend the simulation studies to the nonlinear regime by modifying the code TDA3D to take the incoherent input power. We use this technique to study the start-up and saturation of the TTF-II FEL at DESY and discuss the results.

DOI: 10.1103/PhysRevE.65.016503 PACS number(s): 41.60.Cr, 52.59.Px

I. INTRODUCTION

Free-electron lasers (FELs) working on the principle of self-amplified spontaneous emission (SASE) are being seen as bright and tunable x-ray lasers of the future. The theory of SASE FELs has been developed by many researchers [1–8], and many groups have demonstrated proof-of-principle experiments at long wavelengths [9–14]. More recently, the SASE principle has been demonstrated at visible [15] and at x-ray ultraviolet [16] wavelengths. Presently, there are two major proposals to build SASE FELs at x-ray wavelengths, one at SLAC [17], which is designed to lase at 1.5 Å, and the other at DESY [18], which is designed to lase at 1 Å. Since important physics and technology issues remain to be resolved, due attention is being given to prototype experiments (e.g., TTF-I/II at DESY [16] and LEUTL at APS [15]), and to the development of fast and reliable simulation techniques for detailed studies of the SASE process.

Numerical simulation of the SASE process poses several challenges. First, the discreteness of charge in the electron bunch has to be taken into account in a fully fledged way since it plays an important role in the basic process of SASE. However, the number of electrons in a typical electron bunch injected in a SASE FEL is so large that it is not possible to consider the trajectories of individual electrons. Hence, one needs some innovative scheme such that the process can be correctly simulated even with fewer macroparticles. Second, one has to take into account the broad bandwidth of SASE radiation. It is well known that the bandwidth of SASE radiation is not transform limited, and this bandwidth has to be taken into account even while performing steady-state simulations. Typically this requires the use of time-dependent codes that allow the radiation to grow over a finite bandwidth. The time-dependent codes, however, take large CPU time and require large memory.

Recently, Yu [19,20] has argued that one can use a time-independent code (such as TDA3D [21]) after modifying it to include *multiple* phase-space buckets, to calculate the output power in SASE FELs. He shows that if one uses N_u buckets in the code (where N_u is the number of undulator periods), then the code will correctly calculate the total output power. He also presents simulation results that agree well with the theoretical prediction. However, there are some problems with this argument, as well as with the simulations performed, which we discuss in this paper. We show that it is not necessary to invoke multiple phase-space buckets in order to correctly calculate the output power in SASE using a time-independent code.

In the following section, we discuss basic simulation issues in SASE FELs. We then move on to time-independent SASE simulations in Sec. III where we discuss multiple-bucket as well as single-bucket simulations. We explicitly show that the time-independent code need not be modified to include multiple phase-space buckets in order to simulate the SASE process correctly. We then explain, in Sec. IV, why a single-frequency steady-state code like TDA3D can be used for SASE calculations even without invoking the concept of multiple phase-space buckets.

The analysis presented in Ref. [19] is expected to be valid only up to the linear regime. In Sec. V we extend these calculations up to the saturation regime. As an example, we have performed start-up simulations for the TTF-II FEL at DESY and found the results to be in agreement with those performed using a time-dependent code like GENESIS [22]. We also use these techniques to study the option of seeding a SASE FEL with the incoherent radiation from a third-generation light source. Finally, we present some conclusions.

II. SIMULATION ISSUES IN SASE FELS

The SASE process is one in which broadband incoherent shot noise from undulating electrons is amplified to high-power coherent radiation due to the collective instability excited in the electron beam-undulator-radiation system. Since the shot noise is an outcome of the discreteness of electrons,

it is essential to take this discreteness into account in order to correctly simulate the start-up of SASE FELs. The situation here is completely different from the process of coherent amplification, which is modeled more frequently. In the case of coherent amplification, the input radiation is supposed to be much stronger than the shot noise. Hence, there is no need to include the evolution of shot noise in the problem. Consequently, amplification of the input radiation does not depend on the discreteness of charge in the electron bunch. The electron bunch can therefore be modeled typically as a collection of fewer macroparticles ($\sim 10^3$; much less than the actual number of particles). Each macroparticle represents a large number of electrons and is assumed to have a charge and mass many times that of a single electron. The important thing here is that the *ratio* of charge to mass is same as that of an electron. Consequently, the macroparticles follow the same dynamical equations as the electrons. The electron dynamics therefore does not get affected by incorporating the concept of macroparticles in the simulation.

The same is true of the radiation dynamics for the process of coherent amplification, as long as the contribution from shot noise is ignored. The evolution of shot noise is dictated by a term proportional to $|\langle e^{i\psi}\rangle|^2$ in the equation for radiation dynamics (where ψ is the ponderomotive phase of the electron, and the averaging is done over all the electrons in the bunch being considered) . Hence, in order to shut the shot noise off, in the case of coherent amplification, phases of these macroparticles are to be adjusted artificially such that $\langle e^{i\psi}\rangle = 0$. There are several schemes [21] for initializing the phase to ensure that $\langle e^{i\psi}\rangle = 0$. One such scheme is the quiet-start scheme, which we discuss henceforth in this paper. In this scheme, only a small number of phases, distributed evenly around zero, are filled identically. Thus, shot noise is effectively shut off, and one does not need to analyze the trajectories of individual electrons in order to simulate the process of coherent amplification; it is sufficient to analyze the trajectories of fewer macroparticles.

In SASE FELs, however, it is not straightforward to apply the concept of macroparticles in the simulation. Here one has to consider the evolution of shot noise, and hence the initial value of the term $\langle e^{i\psi}\rangle$ cannot be ignored. One has to consider the random distribution of phases. If the phases are random, it can be shown using the central limit theorem that the mean value of $|\langle e^{i\psi}\rangle|^2$ is proportional to $1/N$, where N is the number of particles over which the averaging is performed in the term $\langle e^{i\psi}\rangle$. If one is using fewer macroparticles in the simulation, N is reduced by many orders of magnitude compared to the actual case. As a result, the shot noise, i.e., the mean value of the term $\langle e^{i\psi}\rangle$, is artificially enhanced by many orders of magnitude. In order to avoid this, in principle, one should simulate the actual number of electrons in a bunch: typically $\sim 10^{8-10}$ or even more. Obviously, with the existing computer technology, it is not possible to follow individual electrons such that this discreteness could be taken into account in a fully fledged way. One has to have some innovative scheme to simulate the evolution of shot noise using fewer macroparticles.

One important aspect of SASE simulations is the statistical nature of the shot noise. It is well established, experimentally, that this gives rise to bunch-to-bunch intensity fluctuations of the SASE radiation. Hence, in order to correctly calculate the average power in the SASE radiation, one has to take an ensemble average over different random initializations of the electrons in phase space. Typically this would increase the computational time by an order of magnitude or more.

Another important issue is that one has to take the broad bandwidth of the SASE radiation into account. In principle, this can be achieved if one uses a time-dependent computer code in which the electron and radiation beams are assumed to have a finite pulse structure. The entire radiation pulse is assumed to be propagating with a central frequency. The fact that the radiation beam is allowed to have a pulse structure means that the radiation is allowed to grow over a band of frequencies around the central frequency. For example, in a time-dependent code, if the radiation field is calculated over individual thin slices separated by a distance Δz in the electron bunch, the bandwidth is given by the Nyquist frequency $\omega_N = \pi c/\Delta z$. Thus, such a simulation, in principle, allows the radiation to grow in the frequency range $\{\omega_s - \omega_N, \omega_s + \omega_N\}$. By properly choosing the simulation parameters, one can ensure that the SASE spectrum falls within this bandwidth.

Time-dependent codes like GENESIS [22] and GINGER [23] simulate the evolution of shot noise by using an artificial distribution of electron phases, where a controlled amount of randomness is put in such that one gets a predetermined and realistic value for the term $\langle e^{i\psi}\rangle$ even for the smaller number of macroparticles used in the simulation. One thus gets over the problem of enhanced shot noise because of fewer macroparticles used in the simulation. In this way, both the discreteness and the broadband width are included in the analysis by using a time-dependent code. The disadvantage with time-dependent codes is that they require large CPU time and large amount of computer memory compared to time-independent codes like TDA3D. Since studies of start-up in SASE FELs require an ensemble average over many runs (as explained above), the requirement on CPU time is multiplied many fold.

III. USING TIME-INDEPENDENT CODES FOR SIMULATING SASE FELS

A. Multiple-bucket simulations

Time-independent codes do not allow the radiation pulse to have a finite width. The radiation is assumed to be a single-frequency, infinite wave train. The possibility of using a time-independent computer code for simulating SASE FELs, under some special circumstances, has been extensively explored by Yu [19,20]. As discussed in these papers, the time-independent code TDA3D has been modified to include multiple phase-space buckets. The original version of the code uses a single phase-space bucket, i.e., ponderomotive phases of electrons can be distributed only in the range $\{-\pi, +\pi\}$. The modified code allows the phases to be distributed, and evolve, over the range $\{-n_l\pi, +n_l\pi\}$, where n_l is an integer. Thus, one has n_l phase-space buckets in this case. This is then interpreted as if one is simulating a ficti-

tious electron-beam distribution with longitudinal periodic structure of $n_l\lambda_s$ (where λ_s is the radiation wavelength). Consequently, a periodic boundary condition is set on the electron beam with period equal to $n_l\lambda_s$. Such a system allows only a discrete radiation spectrum where frequencies are uniformly separated by a spacing, which is ω_s/n_l, where ω_s is the radiation frequency. By properly choosing the number of phase-space buckets, one can alter this spacing. The number of phase-space buckets has been chosen to be N_u in Refs. [19,20], where N_u is the number of undulator periods. In this case, the relative frequency spacing $\Delta\omega/\omega_s$ is given by $1/N_u$. The relative ''full width'' of the SASE spectrum $(\Delta\omega/\omega_s)_{SASE}$ is shown to be $\approx(1/N_u)\sqrt{L_u/4L_G}$ in Refs. [6,19], where L_u and L_G are the undulator length and the power gain length, respectively. If the radiation frequency ω_s coincides with the peak of the SASE spectrum, then it is obvious that the condition for only one spectral line to fall within the full width of the SASE spectrum is that $(\Delta\omega/\omega_s)_{SASE}<2\Delta\omega/\omega_s$. Clearly, this is satisfied as long as the undulator length is much shorter than 16 power gain lengths. As long as this criterion is met, it is justified to use the single-frequency code since only one spectral line falls within the full width of the SASE spectrum. The output power, which the code gives, is the power integrated over a bandwidth (equal to the frequency spacing) around the central frequency ω_s. Hence, the output power can also be written as $(\omega_s/N_u)(dP/d\omega)_{\omega_s}$.

Next, we consider the issue regarding the number of particles to be used in the simulation. In the approach followed in the above references, only a portion of the electron bunch, having a length equal to N_u radiation wavelengths, is being used in the simulation. If I is the electron-beam current in the bunch, the actual number of electrons over this distance is N_uN_λ, where N_λ is the number of electrons over one radiation wavelength and is given by $I\lambda_s/ec$. It has been shown in Ref. [19], by carefully following each step in the analytical calculation of SASE power using the linearized Maxwell-Vlasov equations, that the average output power $\langle P\rangle\propto 1/N$, where N is the number of particles used in the simulation. In our opinion, this is essentially a manifestation of the fact that in the linear regime, which is basically the small-signal regime, the gain is intensity independent. Consequently, the output power is directly proportional to the input power. As discussed earlier, the initial shot noise is proportional to $1/N$. Hence, the average output power $\langle P\rangle\propto 1/N$. However, in the nonlinear regime, gain is intensity dependent and the output power is not expected to be linearly related to the input power in that case. Hence, the above scaling relation is valid only in the linear regime.

Using this scaling relation in the linear regime, one can perform the actual simulation with fewer macroparticles, say N_{sim}, and scale the output power accordingly to get the actual value. For example, if P_{sim} is the average power calculated using simulations, the actual power P is given by

$$P=\frac{ecP_{sim}N_{sim}}{n_l\lambda_s I}, \tag{1}$$

where N_{sim} is the total number of simulations particles distributed over n_l phase-space buckets. Note that $n_l=N_u$ in the analysis presented here. This equation has been used to calculate the total power generated in the SASE process for some specific FEL designs [19,20]. In particular, in Ref. [20], simulation studies have been performed for the BNL Cornell-Wiggler *A* SASE experiment using the modified code. The parameters for this experiment are: $\gamma=82$, r_{beam} (beam size)$=340$ μm, $\lambda_u=3.3$ cm, $a_u=1.018$, $N_u=60$, and $\lambda_s=5.05$ μm. First, the calculations are performed for $I=10$ A, where the radiation is mostly spontaneous emission. Using a total number of 72 000 simulation particles (i.e., using 60 phase-space buckets and putting 1200 particles in each bucket; $N_u=60$), the code gives the output power $P_{sim}=40$ W after averaging over many runs. (Note that one needs to do an averaging over many runs owing to the statistical nature of the process.) Using Eq. (1) (and putting $n_l=N_u$), one gets the actual power to be 0.04 W. This agrees quite well with the theoretically predicted value of 0.043 W for spontaneous emission. Next, the calculations are performed for higher currents where the self-amplification of spontaneous emission starts taking place. The output power as calculated by the code, after scaling down using Eq. (1), is compared with the theoretically predicted value and the comparison has been found to be good, validating this simulation technique.

We now present a couple of observations about this simulation technique and its interpretation. First of all, it is not very clear why one should use exactly N_u phase-space buckets. For example, one could use, say, $2N_u$ phase-space buckets and still argue that the use of a single-frequency code is justified as long as $L_u<4L_G$. Second, a time-independent code like TDA3D takes only the *phases* of the particles as input, it does not take the *actual positions* of the particles along the bunch as input. A cursory look at the dynamical equations, which the code solves numerically, reveals that the code does not distinguish between the dynamics of two particles having phases separated by an integral multiple of 2π. Hence, it is not possible to put information in the code regarding the number of wavelengths over which all the particles being simulated are distributed. In particular, it is not possible for the code to distinguish whether the N_{sim} particles are distributed over n_l phase-space buckets or just a single phase-space bucket. Even if one initializes the phases over ''n_l'' buckets, for the trigonometric operators appearing in the equations for the particle and radiation dynamics, only the value of phases modulo 2π is important. Consequently, for this operator, all the particles are effectively collapsed into a single phase-space bucket. Hence, one expects the output power calculated by the code (i.e., P_{sim}) to be independent of the number of phase-space buckets used in the simulation. P_{sim} calculated in this way, for a fixed number of simulation particles (N_{sim}), when put in Eq. (1), however, yields different numbers for the actual output power P for different values for the number of phase-space buckets (n_l) used in the simulation. In particular, if one uses $n_l=1$, as we have argued the code does, then Eq. (1) gives a value for the actual output power that is different from the prediction with $n_l=N_u$. Hence, there is a fundamental inconsistency between the analysis presented in Ref. [19] and the mathemati-

cal structure of the code. The agreement between the code (which uses $n_l=1$) and the theory (which uses $n_l=N_u$) therefore seems fortuitous. In the following section, we provide a new interpretation of the theory that reconciles theory and simulation. First, however, we present simulation results with single bucket.

B. Results from single-bucket simulations

In order to make a comparison between the two cases: (i) single and (ii) multiple phase-space bucket initialization in TDA3D, we have modified the code such that one can initialize the ponderomotive phases of the particles over N_{buck} phase-space buckets. The phases are initialized over a single phase-space bucket, i.e., over the range $\{-\pi,+\pi\}$, in the original version of TDA3D. Here, we initialize the phases of the simulation particles over a range $\{-N_{buck}\pi,+N_{buck}\pi\}$. Also, the phases are confined in this range as they evolve, by adding or subtracting an integer times 2π whenever they go out of this range. By putting $N_{buck}=1$, one recovers the single-bucket case. By putting $N_{buck}=N_u$, one performs the simulations of Refs. [19,20].

The parameters used in the simulation here are the same as in the last section. These correspond to the BNL Cornell-Wiggler *A* SASE experiment and are the same as those used in the simulations presented in Ref. [20]. First, we study the case of spontaneous emission by taking a small current, $I=10$ A. We use a total number of 72 000 particles in a single phase-space bucket, i.e., we put $N_{buck}=1$ in our simulation. This is same as 1 200 particles each in 60 phase-space buckets as in the simulations done in Ref. [20]. We have used random initialization of particle phases and used a very small input power of 10^{-10} W. In principle, we should use zero input power to study the evolution of shot noise. It is, however, not possible to run the code with exactly zero input power since various normalization constants take the indeterminate form (0/0) and the code gives overflow errors. We have checked that the input power is small enough so that the output power is entirely determined by the shot noise of the simulation particle and is not affected by this number. For example, we get exactly the same result if we start with a power of 10^{-20} W. In order to compare our simulation results with the analytic results, we also switch off the natural focusing due to the undulator in the code and change the initialization of the electron beam in the transverse (x,y) plane such that it uses a step function profile instead of the waterbag model.

Performing the simulations at the radiation wavelength of 5.05 μm (at which the spontaneous emission is peaked in this case), we get the output power $P_{sim}=42$ W after averaging over 30 runs. The number of electrons over one radiation wavelength $\lambda_s=5.05$ μm in the real situation, for $I=10$ A, will be $N_\lambda=1.1\times10^6$. Putting this value of N_λ and $N_{sim}=72\,000$ in Eq. (1), one gets the actual power $P=0.046$ W. This compares well with the theoretically calculated power of 0.043 W and also with the simulation result of 0.04 W using 60 phase-space buckets (since there are 60 periods in the undulator), as described in Ref. [20]. We have also checked that one gets exactly the same result by putting $N_{buck}=60$, i.e., using multiple phase-space buckets.

Next, we perform the simulations for SASE by using a higher current, $I=110$ A. All other parameters remain unchanged. Here, we get an output power of 87.36 kW using the simulations averaged over 30 runs. For this case, using the scaling law we get the actual power $P=8.67$ W. At this current, the spontaneous emission power is expected to be 11×0.043 W, i.e., 0.473 W. The SASE power of 8.67 W is 18 times higher than the spontaneous emission power. Using theoretical calculations described in Ref. [20], SASE power is expected to be 15 times the spontaneous emission power. Hence, here also the single phase-space bucket simulation result agrees well with the theoretical prediction. We have also confirmed that we get exactly the same results for $N_{buck}=60$, even in this case. It is important to note that in both cases, the agreement between the $N_{buck}=60$ and $N_{buck}=1$ results is to within computer accuracy, indicating that the code is unable to distinguish between multiple and single buckets.

IV. REINTERPRETATION OF TIME-INDEPENDENT SASE SIMULATIONS

In the preceding section, we have specifically proven two points. First, we have shown explicitly that the code TDA3D, without any modification, gives the same result as that given by the modified code incorporating multiple phase-space buckets as in Refs. [19,20]. Second, even for the case of single-bucket simulations, in order that simulation results agree with the analytic theory, one has to physically interpret the single phase-space bucket in TDA3D simulation as N_u radiation wavelengths longitudinally. It is to be noted that if one interprets the single-bucket simulations as though the particles were distributed over one radiation wavelength longitudinally, one gets into the ambiguity that the power calculated using Eq. (1) (putting $n_u=1$) does not agree with the analytic prediction. Both these observations are, however, inconsistent with the analysis presented in Ref. [19]. In order to reconcile theory with simulations, we reinterpret the time-independent SASE simulations.

We start by having a look at the following equation for the evolution of radiation dynamics, which is used by a time-independent code like TDA3D:

$$\left[2ik_s\frac{\partial}{\partial z}+\nabla_\perp^2\right]a_s e^{i\phi_s}=-\frac{eZ_0I}{m_ec^2}\left\langle\frac{a_u e^{-i\psi}}{\gamma}\right\rangle, \qquad (2)$$

where γ is the energy of the particle in units of the rest mass, z is the position along the undulator axis, a_u is the dimensionless rms undulator parameter, a_s is the dimensionless radiation field, k_s is the radiation wave number, ϕ_s is the phase of the radiation field, ψ is the pondermotive phase of the particle, Z_0 is the vacuum impedance, I is the beam current, and c is the speed of light. One recalls that while deriving this equation, one needs to do an averaging over a length, which is equal to an integral multiple of the radiation wavelength. The term $\langle a_u e^{-i\psi}/\gamma\rangle$ on the right-hand side of the equation is a result of that averaging, which is over all the particles in this length.

In the case of a time-independent code, one does not consider the slippage between the electron and the radiation pulses. Consequently, the radiation field evolves as a result of interaction with the same set of particles throughout the length of the undulator. In the realistic case, the electron bunch keeps slipping behind the radiation pulse. The radiation field at a given position along the bunch thus keeps evolving due to interaction with different slices in the bunch. In this way, the radiation field at a given position along the bunch interacts with all the electrons that are up to one slippage distance ($N_u\lambda_s$) apart from it. Hence, in a realistic situation, the total number of electrons with which the radiation field interacts is N_uN_λ. This is the number of electrons which, in principle, should be used in the time-independent simulations and over which the term $\langle a_u e^{-i\psi}/\gamma\rangle$ on the right-hand side of the equation should be averaged. Consequently, Eq. (2) is interpreted as being averaged over a longitudinal distance, which is equal to $N_u\lambda_s$. Hence, the radiation field calculated after solving such an equation is only supposed to give a value that is averaged over a length scale $N_u\lambda_s$. Converted to time, it is interpreted as averaged over a time scale $N_u\lambda_s/c$.

Next, we give the interpretation to the result obtained by using a time-independent code. First of all, the code uses an equation that is averaged over a time scale of $T=N_u\lambda_s/c$. It is obvious that one cannot study the details of the temporal structure at a time scale less than T using these equations. However, for time scale greater than T, one *can*, in principle, use these equations to study the temporal structure of the radiation amplitude; this is precisely what is done in time-dependent simulations. Hence, if $E(t)e^{-i\omega_s t}$ is the radiation field as experienced by an observer at the exit of the undulator along the undulator axis, $E(t)$ is supposed to vary only for a time scale greater than T. Consequently, $E(t)$ is to be treated as a constant over time scale less than T. The time-averaged equation [Eq. (2)], however, *does* allow $E(t)$ to vary over a time scale greater than T. The frequency spectrum of the radiation for the observer, in such a situation, will be given by

$$E(\omega)\propto\int_{-\infty}^{+\infty}E(t)e^{i(\omega-\omega_s)t}\,dt. \tag{3}$$

Now, it is obvious that for $(\omega-\omega_s)>1/T$, the term $e^{i(\omega-\omega_s)t}$ will be oscillating rapidly with a time period less than T. On this time scale, $E(t)$ can be treated as constant. Consequently the integrand will be rapidly oscillating and the value of the integral will tend to zero. For $(\omega-\omega_s)<1/T$, the term $e^{i(\omega-\omega_s)t}$ will be oscillating slowly with a time period greater than T. On this time scale, $E(t)$ *can* have variations. Consequently, the integrand is no more a simple oscillatory function and hence it will not tend to zero. Hence, we prove that the radiation spectrum $E(\omega)$ is allowed to have a finite bandwidth of $1/T=c/N_u\lambda_s$, if one is using equations that are averaged over a time scale T. In the case of time-independent simulations, the temporal variations within the pulse are averaged out since one does not consider the temporal structure of the radiation pulse. In the frequency domain, this means that the output power as given by the code is integrated over this bandwidth, i.e., $\Delta\omega/\omega_s=1/N_u$. In other words, the output power as given by the code can be interpreted as $(\omega_s/N_u)(dP/d\omega)_{\omega_s}$, where $(dP/d\omega)_{\omega_s}$ is the power spectrum averaged over the bandwidth (ω_s/N_u). Hence, it is justified to use a time-independent computer code as long as the "full bandwidth" of the SASE spectrum is narrower than the bandwidth accepted by the code. Putting in the exact expressions, here also one gets the same condition, i.e., the undulator length must be less than 16 power gain lengths in order that the time-independent code can be used. The time-dependent code however has the advantage that it can be used to resolve the frequency spectrum better since a large number of slices are used in the simulation. In the case of time-independent simulations, one effectively uses only one slice. As a result of this, the resolution is the same as the bandwidth.

Next, the interpretation of the scaling law. As discussed earlier in this section, the number of electrons that, in principle, should be used in a time-independent simulation is N_uN_λ, even in the case of a *single* phase-space bucket. These are, in fact, the particles that are distributed over N_u radiation wavelengths. However, these are represented by a distribution of phases in the range $\{-\pi,+\pi\}$. Now, if one uses N_{sim} macroparticles in the simulation (instead of N_uN_λ particles in the actual case) and gets an average output power of P_{sim}, the average output power for the real situation will be given by Eq. (1), where $n_l=N_u$. We prove it even for the case of a single phase-space bucket.

Hence, we conclude that one can use a time-independent code like TDA3D, without modifying it to include multiple phase-space buckets, to calculate the average output power as well as power spectrum.

V. EXTENDING THE SIMULATIONS TO THE SATURATION REGIME

The scaling relation between the output power and the number of simulation particles, as discussed in the preceding sections, is valid only in the linear regime. The above analysis is therefore applicable only in the linear regime. In order to extend the analysis beyond the linear regime, we perform the simulation in two stages. In the first stage, which is few gain lengths along the undulator and where the system is in the linear regime, the output power is calculated using simulations as mentioned above, and then scaled down using the scaling relation. In the second stage, i.e., over the remaining portion of the undulator, this scaled power is used as an incoherent seed and is allowed to evolve by shutting off the shot noise. Here, the input radiation is much stronger than the shot noise level. Hence, the shot noise can be ignored. In order to shut off the shot noise, the ponderomotive phases of the electrons are initialized using the quiet-start scheme, as is usually done. The input radiation is, however, not expected to have transverse coherence at this stage. In order to incorporate incoherent seed radiation in TDA3D, we had earlier made some modifications in the code [24] where the input radiation is not assumed to be in the TEM_{00} mode. In fact, the distribution of phases is assumed to be random in the transverse plane. Since there is no shot noise involved here,

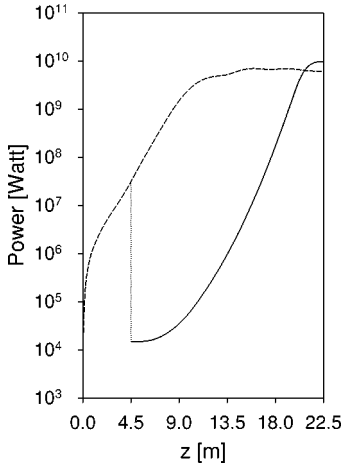

FIG. 1. Plot of radiation power along the length z of the undulator for the TTF-II FEL. The dashed curve shows evolution of the radiation power from the shot noise, for a particular run, using 16 500 particles. As mentioned in the text, this power is scaled down, using Eq. (1), to get the actual power. Since the scaling relationship is valid only in the linear regime, we truncate this calculation at 4.5 m, which is the exit of the first undulator. The solid curve shows the evolution of the radiation power along the remaining four (second to fifth) undulators, where the simulation is performed using quiet-start initialization and starting with a power level actually obtained from the dashed curve and then scaling it down. Note that one does not need to apply any scaling law here, the solid curve shows the evolution of actual power.

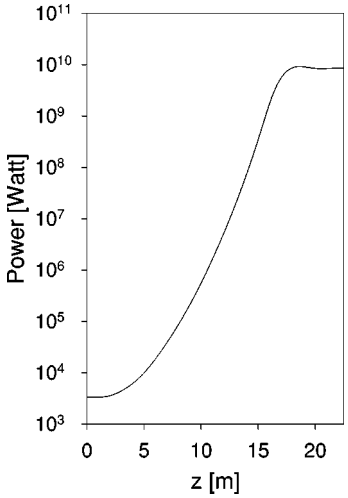

FIG. 2. Seeding of TTF-II FEL using incoherent radiation from a third-generation light source. The parameters used in the simulation are described in Sec. V. Note that the saturation occurs at $z=18$ m, compared to $z=21.5$ m in Fig. 2, where the radiation evolves from shot noise.

one can perform the simulations with fewer simulation particles without the need of any scaling law. Unlike the first stage of the simulation, here one can perform the simulation even up to the saturation regime.

In order to illustrate the above procedure, we next present the results of start-up simulation studies performed for the parameters of TTF-II. The parameters used in the simulation are taken from [22] and are $\gamma=2000$, $\sigma_\gamma=2$, $r_{beam}=50$ μm, $I_{peak}=2500$ A, $\lambda_u=2.73$ cm, $a_u=0.895$, $L_u=22.5$ m, and $\lambda_s=6.18$ nm. We notice that the undulator consists of five sections, each 4.5 m long. The gain length for these parameters is 1.05 m. Hence, each undulator is 4–5 gain lengths long. We divide the simulation into two stages. In the first stage, we simulate the evolution of power only in the first undulator where the system is expected to be in the linear regime since it is only few gain lengths long. Here, we study the evolution of shot noise in the same way as discussed in Sec. III. Figure 1 shows the evolution of shot noise for a particular run. We get an output power of 46 MW at the end of the first undulator after averaging over 30 runs. A total number of 16 500 particles were used in the simulation. Note that if the shot noise simulations are carried out over the full undulator length, i.e., up to 22.5 m, one finds that the power saturates at ~ 10 m with a power level of ~ 6 GW. This simulation is, however, not useful beyond the linear regime since the scaling relationship for calculating the actual power from this result is not expected to be valid in the saturation regime. Hence, we use this simulation only for predicting the evolution of power up to the first undulator ($z=4.5$ m), where the linear regime persists. Using the scaling law, the actual power at the exit of the first undulator is calculated to be 14.8 kW. This is the power of the incoherent seed radiation, which is used in the second stage of the simulation where we consider the evolution of power over the remaining four undulators, i.e., from $z=4.5$ m to $z=22.5$ m. As discussed earlier, in this stage, we use the quiet-start scheme for initializing the phases, which means that the shot noise is neglected. Figure 1 shows the evolution of power for this stage also. We find that the power saturates at $z\approx 21.5$ m. The saturated power is ~ 8.7 GW. These results match nicely with those presented in Ref. [22] using a time-dependent code GENESIS. There, the saturation length is ~ 20 m, and the saturated power is ~ 4 GW. Keeping in mind that many realistic effects, e.g., gain degradation due to slippage, etc. are considered properly in GENESIS, this comparison is reasonably good. Thus, we see that a time-independent code like TDA3D can be used to simulate the SASE process successfully, even up to the saturation regime.

As an example of the above simulation technique, we explore the possibility of seeding a SASE FEL (TTF-II FEL, in this case) using incoherent light from a third-generation light source. Taking a peak flux of 10^{20} (photons/sec)/0.1% bandwidth from a state of the art third-generation light source, one gets a peak power of ~ 3.3 kW over a bandwidth of 0.1% centered around a wavelength of 6.18 nm. Using this as an incoherent seed, the evolution of power is studied along the length of the undulator as shown in Fig. 2. The technique used is identical to the one used in the second stage of the simulation described earlier in this section. It is clear from Fig. 2 that the power saturates at a length of 18 m. Consequently, the total undulator length can be reduced by ~ 3.5 m if one seeds the FEL from a third-generation light source. This reduction in length could be important, since one of the major challenges in building a SASE FEL lies in making a long undulator.

VI. DISCUSSIONS AND CONCLUSIONS

In this paper, we have explored the possibility of using the time-independent code TDA3D to study the evolution of radiation in the SASE process. We have shown that although the code seemingly uses a single-frequency radiation field, it essentially allows radiation to grow over a bandwidth $\Delta\omega/\omega_s = 1/N_u$. In our interpretation, which is different from that given in Ref. [19], we have attributed this to the fact that the equations, which the code solves to evolve the radiation field, are averaged over a time scale $T = N_u\lambda_R/c$. As a result of this, when the code is used to study the evolution of shot noise, the output power given by the code is essentially integrated over this bandwidth. Consequently, as long as the bandwidth of the SASE radiation is sharper than $1/N_u$, the time-dependent code is expected to give the total power for the SASE process, integrated over the entire bandwidth.

It is interesting to point out that it is possible to include multiple frequencies, which are harmonics of a fundamental frequency (say, ω_0) in a time-independent analysis in order to take the broad bandwidth of the SASE radiation into account, as has been done by Freund and co-workers [25,26]. In the analysis presented in Refs. [25,26], using the simulation code MEDUSA, they have applied this approach to study nonlinear harmonic generation in FELs. In principle, their technique can also be used to study the start-up from shot noise in the same way as in this paper. However, in the results presented in Refs. [25,26], they have only considered the case of a seeded amplifier where the initial radiation power, at the fundamental wavelength, is several times the spontaneous power per gain length. The Maxwell equation, which they have used to study the evolution of radiation, is averaged over a time scale of $2\pi/\omega_0$ in their analysis. Converted to length, this means that the equations are averaged over a length scale of λ_0. This would mean, as explained in Sec. IV, that the radiation is evolving as a result of interaction with electrons that are spread over λ_0. In the actual case, the radiation evolves as a result of interaction with electrons that are spread over the slippage length, i.e., $N_u\lambda_s$. This puts the condition that $\lambda_0 = N_u\lambda_s$ and hence, $\omega_0 = \omega_s/N_u$. Under this condition, and with shot noise initialization, MEDUSA can be used in the same way as we have used TDA3D in this paper. In fact, such an analysis would be even more general, because it would include the effect of nonlinear harmonic interactions in studying the evolution of shot noise in SASE FELs.

We emphasize that in order to use a time-independent code like TDA3D to study the evolution of SASE power, it is not necessary to modify the code to include multiple phase-space buckets as in Refs. [19,20]. Even if one modifies the code to include multiple phase-space buckets, it really does not make any difference. Hence, one can use the code in its original form, where only a single phase-space bucket is considered, to study the evolution of radiation power. It should be noted that the single phase-space bucket here is supposed to contain all the particles that are spread over a slippage length.

To some extent, TDA3D can also be used to study the power spectrum of the output radiation by using the relationship $P = (\omega_s/N_u)(dP/d\omega)_{\omega_s}$, as has been done in Ref. [20]. However, it should be noted that the power spectrum $(dP/d\omega)_{\omega_s}$ in this expression is essentially averaged over a bandwith $1/N_u$. Hence, this analysis cannot give a resolution better than this. In order to get a better resolution of the power spectrum, one has to take recourse to time-dependent codes like GENESIS and GINGER.

In order to extend the calculation up to the saturation regime, where Eq. (1) is no more valid, we take recourse to the quiet-start initialization of electron phases in the second stage of the simulation as discussed in Sec. V. We emphasize that it is important to take the transverse incoherence of radiation into account at this stage, as we have done. We have explicitly shown in earlier work [24] that if one initializes the radiation field in the TEM_{00} mode, as is usually done, one underestimates the length of the undulator required to reach saturation. In order to avoid this, it is important to initialize the radiation phases using a random distribution.

It may be argued that analytic calculations could be used to estimate the power at the end of the first stage. However, as discussed by Yu [19], analytic results are typically available only when the gain is sufficiently high to assume that the exponentially growing term dominates. *In addition*, simulations also allow one to include various realistic effects such as an unmatched electron beam, undulator field errors, etc.

In summary, we have shown that a time-independent code like TDA3D, without any modification, can be successfully used to study the evolution of SASE radiation, starting from shot noise, and that the analysis can be extended to the saturation regime. Our time-independent simulation results agree quite well with those obtained using time-dependent simulations, thus validating our technique. Since time-independent computer codes take much less computational time and memory as compared to time-dependent codes, this analysis could be very useful for performing detailed optimization simulations.

ACKNOWLEDGMENTS

One of us (V.K.) would like to thank L.-H. Yu and M. P. Singh for useful discussions.

[1] P. Sprangle, R. Smith, and V. Granatstein, in *Infrared and Millimeter Waves*, edited by K. J. Button (Academic, New York, 1979), Vol. I, p. 279.

[2] H. Haus, IEEE J. Quantum Electron. **QE-17**, 1427 (1981).

[3] J. Murphy and C. Pellegrini, J. Opt. Soc. Am. B **1**, 530 (1984).

[4] R. Bonifacio, C. Pellegrini, and L. M. Narducci, Opt. Commun. **50**, 373 (1984).

[5] J. M. Wang and L.-H. Yu, Nucl. Instrum. Methods Phys. Res. A **250**, 484 (1986).

[6] K.-J. Kim, Nucl. Instrum. Methods Phys. Res. A **250**, 396

(1986).
[7] K.-J. Kim, Phys. Rev. Lett. **57**, 1871 (1986).
[8] S. Krinsky and L.-H. Yu, Phys. Rev. A **35**, 3406 (1987).
[9] R. K. Parker *et al.*, Phys. Rev. Lett. **48**, 238 (1982).
[10] T. Orzechowksi *et al.*, Phys. Rev. Lett. **54**, 889 (1985).
[11] D. Kirkpatrick, Nucl. Instrum. Methods Phys. Res. A **285**, 43 (1989).
[12] J. Gardelle, J. Labrouche, and J. L. Rullier, Phys. Rev. Lett. **76**, 4532 (1996).
[13] R. Sheffield, J. Goldstein, and D. Nguyen, in *SPIE Proceedings of LASE '97, San Jose, California* (Materials Research Society, Pittsburgh, 1997), p. 2998.
[14] R. Prazeres *et al.*, Phys. Rev. Lett. **78**, 2124 (1997).
[15] S. V. Milton *et al.*, Phys. Rev. Lett. **85**, 988 (2000).
[16] J. Rossbach, in Proceedings of the Twenty-First International Free-Electron Laser Conference, Durham, NC (unpublished).
[17] R. Tatchyn *et al.*, Nucl. Instrum. Methods Phys. Res. A **375**, 274 (1996).
[18] J. Rossbach *et al.*, Nucl. Instrum. Methods Phys. Res. A **375**, 269 (1996).
[19] L.-H. Yu, Phys. Rev. E **58**, 4991 (1998).
[20] L.-H. Yu, Nucl. Instrum. Methods Phys. Res. A **429**, 213 (1999).
[21] T. M. Tran and J. S. Wurtele, Comput. Phys. Commun. **54**, 263 (1989); P. Jha and J. S. Wurtele, Nucl. Instrum. Methods Phys. Res. A **331**, 447 (1993).
[22] S. Reiche, Nucl. Instrum. Methods Phys. Res. A **429**, 243 (1999).
[23] W. Fawley, A. M. Sessler, and E. T. Scharlemann, Bull. Am. Phys. Soc. **38**, 1327 (1993); W. Fawley, Report No. LBID-2141, CBP Tech Note-104, UC-414, 1995 (unpublished).
[24] V. Kumar and S. Krishnagopal, Nucl. Instrum. Methods Phys. Res. A **445**, 77 (2000).
[25] H. P. Freund *et al.*, IEEE J. Quantum Electron. **QE-36**, 275 (2000).
[26] H. P. Freund and P. G. O'Shea, Phys. Rev. Lett. **84**, 2861 (2000).