Fast algorithm for transient current through open quantum systems

King Tai Cheung, Bin Fu, Zhizhou Yu, and Jian Wang*

Department of Physics and the Center of Theoretical and Computational Physics, The University of Hong Kong, Pokfulam Road, Hong Kong, China

(Received 25 October 2016; revised manuscript received 8 February 2017; published 16 March 2017)

Transient current calculation is essential to study the response time and capture the peak transient current for preventing meltdown of nanochips in nanoelectronics. Its calculation is known to be extremely time consuming with the best scaling TN^3 where N is the dimension of the device and T is the number of time steps. The dynamical response of the system is usually probed by sending a steplike pulse and monitoring its transient behavior. Here, we provide a fast algorithm to study the transient behavior due to the steplike pulse. This algorithm consists of two parts: algorithm I reduces the computational complexity to T^0N^3 for large systems as long as $T < N$; algorithm II employs the fast multipole technique and achieves scaling T^0N^3 whenever $T < N^2$ beyond which it becomes $T \log_2 N$ for even longer time. Hence it is of order $O(1)$ if $T < N^2$. Benchmark calculation has been done on graphene nanoribbons with $N = 10^4$ and $T = 10^8$. This algorithm allows us to tackle many large scale transient problems including magnetic tunneling junctions and ferroelectric tunneling junctions.

DOI: 10.1103/PhysRevB.95.125422

I. INTRODUCTION

At the heart of growing demands for nanotechnology is the need for ultrafast transistors whose response time is one of the key performance indicators. The response of a general quantum open system can be probed by sending a steplike pulse across the system and monitored by its transient current over times, making transient dynamics a very important problem. Many experimental data show that most of the molecular device characteristics are closely related to material and chemical details of the device structure. Therefore, first-principles analysis, that makes quantitative and predictive analysis of device characteristics especially its dynamic properties without relying on any phenomenological parameter, becomes a central problem of nanoelectronics.

The theoretical study of transient current dates back to 20 years ago when the exact solution in the wide-band limit (WBL) was obtained by Wingreen *et al.* [1]. Since then the transient current has been studied extensively using various methods [2], including the scattering wave function [3,4], nonequilibrium Green's-function (NEGF) [5–8] approach, and density-matrix method [9]. The major obstacle of theoretical investigation on the first-principles transient current is its computational complexity. Many attempts were made trying to speed up the calculation [3,4,10–14]. Despite these efforts, the best algorithm to calculate the transient current from first principles going beyond the WBL scales like TN^3 using complex absorbing potential (CAP) [15] where T and N are the number of time steps and size of the system respectively. We note that if the WBL is used, the scaling is reduced [12]. However, to capture the feature of band structure of the lead and the interaction between the lead and scattering region the WBL is not a good approximation in the first-principles calculation.

As a result, most of the first-principles investigations on transient dynamics were limited to small and simple one-dimensional systems. There are a number of problems such as magnetic tunneling junctions (MTJs) [16] and ferroelectric tunneling junctions [17], where the system is two dimensional or even three dimensional in nature. For these systems, a large number of k points N_k has to be sampled in the first Brillouin to capture accurately the band structure of the system. For a MTJ structure like Fe-MgO-Fe, at least $N_k = 10^4$ k points must be used to give a converged transmission coefficient [18]. This makes the time consuming transient calculation N_k times longer which is an almost impossible task even with a high performance supercomputer. Clearly it is urgent to develop better algorithms to reduce the computational complexity.

In this paper, we develop a fast algorithm based on NEGF-CAP formalism to calculate transient current for a steplike pulse as a function of time step T which could be helpful in speeding up the first-principles transient calculation. The computational time of this algorithm is independent of T as long as $T < N^2$ where N is the system size [19]. Hence our algorithm is order $O(1)$ as long as $T < N^2$. Four important ingredients are essential to achieve this: (1) the availability of an exact solution of transient current based on NEGF that goes beyond WBL; (2) the use of CAP so that the transient current can be expressed in terms of poles of Green's function; (3) within NEGF-CAP formalism the transient current can be calculated separately in space and time domain making the $O(1)$ algorithm possible; at this point the computational complexity reduces to $N^3 + TN^2$ (algorithm I); (4) the exploitation of Vandermonde matrix enables us to use the fast multipole method (FMM) [20,21] and fast Fourier transform (FFT) to further reduce the scaling to $N^3 + N^2 \log_2 N$ for $T < N^2$ and large N, therefore completely independent of T (algorithm II). To verify the computational complexity, we carry out benchmark calculations on graphene nanoribbons using the tight-binding model. A calculation is also done for a system with $N = 10\,200$ and $T = 10^8$ confirming the $O(1)$ scaling. This fast algorithm makes the computational complexity of transient current calculation comparable to that of static calculation.

*jianwang@hku.hk

II. THEORETICAL FORMALISM

For a general open quantum system with multiple leads under a steplike bias pulse, the Hamiltonian is given by

$$H(t) = \sum_{k\alpha} \epsilon_{k\alpha} \hat{c}^{\dagger}_{k\alpha} \hat{c}_{k\alpha} + \sum_{n} [\epsilon_n + U_n(t)] \hat{d}^{\dagger}_n \hat{d}_n + \sum_{k\alpha n} h_{k\alpha n} \hat{c}^{\dagger}_{k\alpha} \hat{d}_n + \text{c.c.},$$

where $c^{\dagger}$ (c) denotes the electron creation (annihilation) operator in the lead region. The first term in this equation corresponds to the Hamiltonian of leads with ϵ_{k_α} the energy of lead α which contains external bias voltage $v_\alpha(t) = V_\alpha \theta(\pm t)$. The second and third terms represent the Hamiltonian in the central scattering region and its coupling to leads, respectively. Here we have included the time-dependent internal response $U_n(t)$ in the scattering region due to the external bias [22]. Taking $q = \hbar = 1$, the time-dependent terminal current $I_\alpha(t)$ of lead α is defined as [15,23]

$$I_\alpha(t) = \text{ReTr}(\overline{\Gamma}_\alpha [2H(t) - i\partial_t] G^<(t,t) \overline{\Gamma}_\alpha), \tag{1}$$

where $\overline{\Gamma}_\alpha$ is an auxiliary projection matrix which is used for measuring the transient current passing through the lead α. Here $G^<$ and $H(t)$ are the lesser Green's function and the Hamiltonian of the central scattering region, respectively. For a steplike pulse an exact solution for $G^<$ has been obtained by Maciejko *et al.* [6] which goes beyond the WBL. Now we consider the case of upward pulse $v_\alpha(t) = V_\alpha \theta(t)$. In order to have the exact solution in Ref. [6], we assume that $U_n(t) = U_{n,\text{eq}} + (U_{n,\text{neq}} - U_{n,\text{eq}})\theta(t)$ where $U_{n,\text{eq}}$ is the equilibrium potential while $U_{n,\text{neq}}$ is the nonequilibrium potential at the long-time limit. As a result of this instantaneous approximation, at $t = 0$, the system is in equilibrium with Hamiltonian H_{eq} while at $t > 0$ the system is in a nonequilibrium state with a time-independent Hamiltonian H_{neq}.

Note that our fast algorithm relies critically on this approximation where two static potentials are needed at $t = 0^-$ and $t = 0^+$. For instance, our method fails if $U_n(t) = U_{n,\text{eq}} + U_{n,\text{neq}} \cos(\omega t)$. With this instantaneous assumption, our method could be extended to the first-principles calculation where only two static self-consistent Coulomb potentials can be obtained by the usual NEGF method combined with density functional theory (DFT) calculation [24]. In terms of spectral function $A_\alpha(\epsilon,t)$, the lesser Green's function $G^<$ is given by [6]

$$G^<(t,t) = i \sum_\alpha \int \frac{d\epsilon}{2\pi} f(\epsilon) A_\alpha(\epsilon,t) \Gamma_\alpha(\epsilon) A^{\dagger}_\alpha(\epsilon,t). \tag{2}$$

For the upward steplike bias pulse the $A_\alpha(\epsilon,t)$ is [6]

$$A_\alpha(\epsilon,t) = \overline{G}^r(\epsilon + \Delta_\alpha) - \int \frac{d\omega}{2\pi i} \frac{e^{-i(\omega-\epsilon)t} \overline{G}^r(\omega + \Delta_\alpha)}{(\omega - \epsilon + \Delta_\alpha - i0^+)} \times \left[\frac{\Delta_\alpha}{(\omega - \epsilon - i0^+)} + \Delta \tilde{G}^r(\epsilon) \right] \equiv A_{1\alpha}(\epsilon + \Delta_\alpha) + \int d\omega e^{-i(\omega-\epsilon)t} A_{2\alpha}(\omega,\epsilon), \tag{3}$$

where $\overline{G}^r$ and $\tilde{G}^r$ are the nonequilibrium and equilibrium retarded Green's function respectively, Δ_α is the amplitude of external bias $-V_\alpha$, and $\Delta = U_{\text{neq}} - U_{\text{eq}}$ is a matrix where the subscript "neq" and "eq" refer to nonequilibrium and equilibrium potentials, respectively.

Despite the simplification from the conventional double time $G^<(t,t')$ to single time $G^<(t,t)$ used in Eq. (1), the computational cost to obtain $G^<$ remains very demanding due to the following reasons: (1) Consider $A_\alpha(\epsilon,t)$ with a matrix size of N; matrix multiplications $\overline{G}^r(\omega + \Delta_\alpha)$ and $\tilde{G}^r(\epsilon)$ in the integrand of Eq. (3) require computational complexity of $O(N^3)$ for each time step. As a result, the total computational cost over a period of time is at least $O(TN^3)$ where T is the number of time steps. (2) Double integrations in energy space are required for $G^<$. The presence of numerous quasiresonant states whose energies are close to the real energy axis makes the energy integration in A_α extremely difficult to converge. This problem can be overcome using the CAP method [25]. The essence of the CAP method is to replace each semi-infinite lead by a finite region of CAP while keeping the transmission coefficient of the system unchanged. In addition, it has been demonstrated in Ref. [15] that the first-principles results of transient current for molecular junctions obtained from the exact numerical method (non-WBL) and the CAP method are exactly the same. Using the CAP method, the poles of the Green's function can be obtained easily and the spectral function can be calculated analytically using the residue theorem. Expanding the Fermi function using Padé spectrum decomposition (PSD) [26] further allows us to calculate the transient current separately in space and time domain making the $O(T^0N^3)$ algorithm possible.

Now we illustrate how to achieve our algorithm for the transient current calculation, i.e., $I_\alpha(t_j)$ for $j = 1,2,\ldots,T$. Substituting Eq. (3) into Eq. (2), $G^<(t,t)$ can be written as

$$G^<(t,t) = (i/\pi)\left[B_1 + \int d\omega d\omega' e^{-i(\omega-\omega')t} B_2(\omega,\omega') + \sum_\alpha \int d\epsilon d\omega' e^{i(\omega'-\epsilon)t} f(\epsilon) A_{1\alpha} W_\alpha A^{\dagger}_{2\alpha} + \text{c.c.} \right], \tag{4}$$

where $B_1 = \int d\epsilon f(\epsilon) \sum_\alpha A_{1\alpha} W_\alpha A^{\dagger}_{1\alpha}$, $B_2(\omega,\omega') = \int d\epsilon f(\epsilon) \sum_\alpha A_{2\alpha}(\omega,\epsilon) W_\alpha A^{\dagger}_{2\alpha}(\omega',\epsilon)$, and W_α is the CAP matrix. In terms of poles of the Green's function and the Fermi distribution function, we have (see Appendix A)

$$\begin{aligned} G^<(t,t) &= (i/\pi)\left[B_1 + \sum_{nm} e^{-i(\epsilon_n - \epsilon^*_m)t} \bar{B}_2(\epsilon_n, \epsilon^*_m) \right. \\ &\quad + \sum_\alpha \sum_{lm} e^{-i(\tilde{\epsilon}_l - \epsilon^*_m + \Delta_\alpha)t} \bar{f}(\tilde{\epsilon}_l) \bar{B}_{3\alpha}(\tilde{\epsilon}_l, \epsilon^*_m) + \text{c.c.} \\ &\quad \left. + \sum_\alpha \sum_{nm} e^{-i(\epsilon_n - \epsilon^*_m)t} f(\epsilon_n - \Delta_\alpha) \bar{B}_{4\alpha}(\epsilon_n, \epsilon^*_m) + \text{c.c.} \right], \end{aligned} \tag{5}$$

where ϵ_n and ϵ_m ($n = 1,2,\ldots N$) is the complex energy spectrum of $H_{\text{neq}} - i\sum_\alpha W_\alpha$ in the lower half plane while $\tilde{\epsilon}_l$ is the poles of $f(E)$ using PSD with $l = 1,\ldots N_f$; N_f is the total number of those poles for the adopted Padé approximant;

other parameters in Eq. (5) are given as

$$\bar{B}_2 = -4\pi^2[B_2(\omega,\omega')(\omega-\epsilon_n)(\omega'-\epsilon_m^*)]\Big|_{\omega=\epsilon_n,\omega'=\epsilon_m^*}, \quad (6)$$

$$\bar{B}_{3\alpha} = -2\pi i A_{1\alpha}(\tilde{\epsilon}_l)W_\alpha[A_{2\alpha}^\dagger(\omega',\tilde{\epsilon}_l)(\omega'-\epsilon_m^*)]\Big|_{\omega'=\epsilon_m^*}, \quad (7)$$

$$\bar{B}_{4\alpha} = -4\pi^2[A_{1\alpha}(\epsilon)W_\alpha A_{2\alpha}^\dagger(\omega',\epsilon)(\omega'-\epsilon_m^*) \times(\epsilon-\epsilon_n+\Delta_\alpha)]\Big|_{\epsilon=\epsilon_n-\Delta_\alpha,\omega'=\epsilon_m^*}, \quad (8)$$

and

$$\bar{f} = 2\pi i[f(\epsilon)(\epsilon-\tilde{\epsilon}_l)]\Big|_{\epsilon=\tilde{\epsilon}_l} \quad (9)$$

is the residue of f in the PDS representation.

Within CAP framework, $G^<$ in Eq. (1) is the lesser Green's function of the central scattering region excluding the CAP regions. Substituting the second term of Eq. (5) into the first term in Eq. (1), we find its contribution to current (denoted as I_1)

$$I_1(t) = \mathrm{Re}\sum_{nm} e^{-i(\epsilon_n-\epsilon_m^*)t}\mathrm{Tr}(\overline{\Gamma}_\alpha 2H_{\mathrm{neq}}\bar{B}_2(\epsilon_n,\epsilon_m^*)\overline{\Gamma}_\alpha) \equiv 2\mathrm{Re}\sum_{nm} e^{-i(\epsilon_n-\epsilon_m^*)t}M_{nm}, \quad (10)$$

where the matrix M does not depend on time. We see that the space and time domains have been separated.

Denoting a Vandermonde matrix $V_{jk} = \exp(i\epsilon_k t_j)$ with $k = 1,2,\ldots,N$, $t_j = jdt$, $j = 1,2,\ldots,T$ where dt is the time interval, we have $I_1(t_j) = [V^t(M+M^\dagger)V^*]_{jj}$. Using this approach, we finally obtain (see Appendix C for details)

$$I_L(t_j) = I_{0L} + [V^tM_1V^*]_{jj} + \left(\sum_{\alpha=L,R}[V^tM_{2\alpha}\tilde{V}_\alpha^*]_{jj} + \mathrm{c.c.}\right), \quad (11)$$

where $\tilde{V}_{\alpha jk} = \exp[i(\tilde{\epsilon}_k+\Delta_\alpha)t_j]$ is a $T\times N_f$ matrix, M_1 is a $N\times N$ matrix while M_2 is a $N\times N_f$ matrix. Since ϵ_k is the complex energy in the lower half plane, V_{jk} goes to zero at large j. Hence I_{0L} is the long-time limit of transient current which can be calculated using the Landauer-Buttiker formula. The time-dependent part of the transient current can be separated into a real-space calculation (calculation of M_1 and $M_{2\alpha}$) and then a matrix multiplication involving time. We note that at room temperature the Fermi function can be accurately approximated by 15 or 20 Padé approximants. Hence the calculation of $[V^tM_1V^*]_{jj} + (\sum_\alpha[V^tM_{2\alpha}\tilde{V}_\alpha^*]_{jj} + \mathrm{c.c.})$ can be combined to give TN^2 computational complexity.

Now we examine the computational complexity of this algorithm (denoted as algorithm I). Clearly the total computational complexity is N^3+TN^2. At this stage, the algorithm is not $O(1)$ yet. In part III, we will show that matrix multiplication V^tM, where M is M_1 or $M_{2\alpha}$, can be done using the FMM and FFT (denoted as algorithm II). This will reduce the computational complexity of V^tM from TN^2 to $T\log_2 N$. Hence for $T<N^2$, the computational complexity is $N^3+N^2\log_2 N$. For $T>N^2$, the scaling is $N^3+T\log_2 N$. However, for large T, the physics comes into play. Since ϵ_j is the complex energy of the resonant state, $V_{jT} = \exp(-i\epsilon_j dtT)$ decays quickly to zero before $T=N^2$. For a graphene nanoribbon with $N=10^4$ (see details below), the maximum value of $V_{jT} = \exp(-i\epsilon_j dtT)$ is 10^{-3} when $T=N$ and $dt=1$ fs. Consequently all the matrix elements are zero for $T=10N$. Hence for large systems, the chance to go beyond $T=N^2$ is small. In this sense, algorithm II is order $O(1)$ algorithm.

III. FAST MULTIPOLE METHOD

The fast multipole method [20] has been widely used and has been ranked among the top ten best algorithms in the 20th century [27]. It is extremely efficient for large N. In the following, we will illustrate how to speed up the calculation of transient current defined in Eq. (12) below. We want to calculate the following quantity:

$$I(t_j) = \sum_{n,m}\exp(-i\epsilon_n t_j)M_{nm}\exp(i\epsilon_m^* t_j), \quad (12)$$

where the matrix M can be expressed in terms of vectors as $M = (c_0,c_1,\ldots,c_{N-1})$ and $V_{nj} = \exp(-i\epsilon_n t_j)$ is a Vandermonde matrix with $t_j = jdt$ and $j = 1,2,\ldots T$. Equation (12) is of the form V^tMV^* where t stands for transpose. In the following, we outline how to calculate V^tc using $\kappa_1 N+\kappa_2 N\log_2 N$ operations where c is a vector of N components and κ_1 and κ_2 are constants.

Setting $a_j = \exp(-i\epsilon_j dt)$ and denoting T the number of time steps, then $b = V^tc$ is equivalent to $b_n = \sum_{j=0}^{N-1} c_j(a_j)^n$. A direct computation shows that the entries of $b = V^tc$ are the first T coefficients of the Taylor expansion of

$$S(x) = \sum_{j=0}^{N-1}\frac{c_j}{1-a_jx} = \sum_n^\infty\sum_{j=0}^{N-1}c_j(a_jx)^n = \sum_n b_nx^n, \quad (13)$$

where $b_n = \sum_{j=0}^{N-1}c_j(a_j)^n$. Denoting $\bar{S}(x) = \sum_{n=0}^{T-1}b_nx^n$ and setting $x=\omega_T^l$ with $\omega_T = \exp(i2\pi/T)$ we can calculate $\bar{S}(\omega_T^l)$ which is the Fourier transform of b_n,

$$\bar{S}(\omega_T^l) = \sum_{j=0}^{N-1}\sum_{n=0}^{T-1}c_ja_j^n\omega_T^{nl} = \sum_{j=0}^{N-1}c_j\frac{1-(a_j\omega_T^l)^T}{1-a_j\omega_T^l} = \omega_T^{-l}\sum_{j=0}^{N-1}\frac{c_j(1-a_j^T)}{(1/\omega_T)^l-a_j},$$

where we have used $[\omega_T]^T = 1$. Note that the fast multipole method (FMM) aims to calculate $v_l = \sum_j c_j/(x_l-a_j)$ with $O(N)$ operations instead of N^2 operations. Hence $\bar{S}(\omega_T^l)$ can be obtained using FMM, from which we calculate b_n using FFT.

Now we estimate the computational complexity for $T\leqslant N$. For FMM we need $\kappa_1\max(T,N)$ operations where κ_1 is about $40\log_2(1/\tau)$ with τ the tolerance [32]. For FFT the computational complexity is at most $\kappa_2 N\log_2 N$ where κ_2 is a coefficient for FFT calculation [32]. To compute V^tM where M has N vectors, we have to calculate V^tc N times. Hence the total computational complexity is $\kappa_1N^2+\kappa_2N^2\log_2 N$. This

algorithm is denoted as algorithm IIa while the algorithm for $T < N^2$ discussed below is denoted as algorithm IIb.

For very large T up to $T = N^2$ (if $N = 10^4$ we have $T = 10^8$), we will show that the computational complexity is $\kappa_1 N^2 + 2\kappa_2 N^2 \log_2 N$. In fact, it is easy to see that $I(t_j)$ defined in Eq. (12) are the first T coefficients of the Taylor expansion of

$$S(x) = \sum_{n,m=0}^{N-1} \frac{M_{nm}}{1 - a_n a_m^* x} \tag{14}$$

$$= \sum_{j}^{\infty} \sum_{n,m=0}^{N-1} M_{nm} (a_n a_m^*)^j x^j = \sum_j I(t_j) x^j, \tag{15}$$

where $a_n = \exp(-i\epsilon_n dt)$. Now we define two new vectors u and d which have N^2 components with $u^t = (c_0^t, c_1^t, \ldots, c_{N-1}^t)$ [recall our definition $M = (c_0, c_1, \ldots, c_{N-1})$] and $d^t = (a_0^* a^t, a_1^* a^t, \ldots, a_{N-1}^* a^t)$, where once again t stands for transpose. With the new vectors defined, $S(x)$ in Eq. (14) is expressed as

$$S(x) = \sum_{j=0}^{N^2-1} \frac{u_j}{1 - d_j x}, \tag{16}$$

which is exactly the same form as Eq. (13). The only difference is that c and a in Eq. (13) have N components and S has to be calculated N times while u and d in Eq. (16) have N^2 components and we calculate S defined according to Eq. (16) just once. Therefore the computational complexity is $\kappa_1 N^2 + \kappa_2 N^2 \log_2 N^2$. If $T = nN$ with $n = 1, 2, \ldots N$, it is not difficult to show that the computational complexity is $\kappa_1 TN/n + \kappa_2 T(N/n) \log_2(nN) = \kappa_1 N^2 + \kappa_2 N^2 \log_2(nN)$.

To summarize, the computational complexity of Eq. (12) is $\kappa_1 N^2 + 2\kappa_2 N^2 \log_2 N$ for $T < N^2$. It is easy to show that for $T > N^2$ the scaling is dominated by $\kappa_2 T \log_2 N$. However, for large T, the physics comes into play. Since $a_j = \exp(-i\epsilon_j dt)$ with ϵ_j the energy of resonant state, a_j^T quickly decays to zero before $T = N^2$ and hence no need to go up for $T > N^2$.

IV. NUMERICAL TEST

To demonstrate the power of this algorithm, we calculate the transient current in a graphene nanoribbon. Graphene is a well-known intrinsic two-dimensional (2D) material with many exotic properties [28,29]. Its transient behavior in response to a steplike pulse was studied in the literature [7,30,31]. Tuovin *et al.* [7] explored the metal-graphene-metal system at zero temperature under the effect of ribbon length, width, and bias and found the present of a several-hundreds-femtoseconds oscillation period in transient current, caused by the lead-ribbon reflections; Again at zero temperature, Perfetto *et al.* [30] studied the phenomenon of two temporal plateaus that appeared in the transient current of wide graphene nanoribbon (width $W \geqslant 20$ nm) and deduced that the two had arisen from diverse origins; for zigzag ribbon, Xie *et al.* [31], investigated the difference in the current response for symmetric and asymmetric systems. While in all these studies the transient current through a central region of pure graphene nanoribbons under a step bias has been studied for both armchair and zigzag structures under different circumstances,

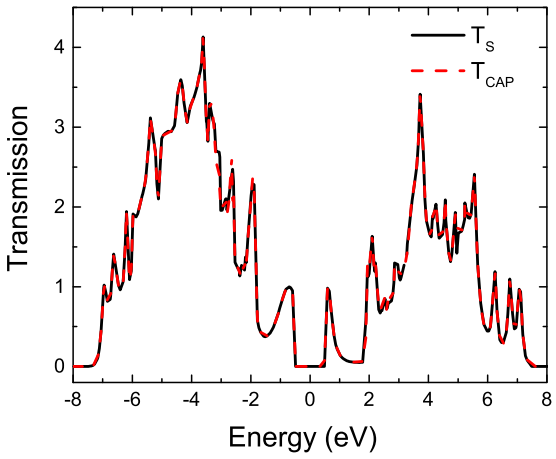

FIG. 1. The transmission coefficient of the zigzag graphene nanoribbon for a system of 10 000 atoms. The solid line (T_S) is the exact numerical result using self-energy of the lead and the dashed line (T_{CAP}) is from CAP.

none of them has considered the cases when barrierlike gated regions exist in central region. Here, we will test our algorithm on a gated zigzag graphene nanoribbon at room temperature using the tight-binding (TB) Hamiltonian given by

$$\hat{H} = -h_0 \sum_{\langle i,j \rangle} \hat{c}_i^\dagger \hat{c}_j - q \sum_i [V_i \theta(t) + V_{g1i} + V_{g2i}] \hat{c}_i^\dagger \hat{c}_i, \tag{17}$$

where $\hat{c}_i^\dagger$ ($\hat{c}_i$) is the creation (annihilation) operator at site i and $h_0 = 2.7$ eV is the nearest hopping constant. Here $V(x) = V_L + (V_R - V_L)x/L$ is the potential landscape due to the external bias with $V_R = -V_L = 0.54$ V and V_{g1} and V_{g2} are gate voltages in regions S_1 and S_2, respectively.

We first confirm that the transient current calculated using this method is the same as that of Ref. [15]. Using 30 layers of CAP, transmission coefficient versus energy was calculated which shows good agreement with the exact solution (Fig. 1 gives the comparison for a graphene nanoribbon with $N =$ 10 000). This also ensures the correct steady-state current. For the transient current, excellent agreement is also obtained between our algorithm and that of Ref. [15] (see Fig. 2). We note that even in the presence of gates, an overshooting at the beginning is still observed, similar to the ungated graphene [31] but the oscillating tail is not observable after the overshooting peak. We also tested with the cases for gated graphene ribbon with a larger width W (not shown) and obtained higher transient current over time which was observed previously [7] for the ungated condition.

We have performed calculations on transient current through a zigzag graphene nanoribbon of 10 000 atoms with $T = 20\,000$ time steps (each time step is 1 fs). The width of the system is two unit cells (16 atoms) while the length of the system is 625 unit cells. Two gate voltages of 2.2 V were applied so that the system is in the tunneling regime. The bias voltage is $v_L = -v_R = 0.5$ V. From Fig. 3, we see a typical behavior of transient current with the current shots up initially and then decreasing to the long-time limit. Our numerical results using FMM (algorithm II) show that 100 ps

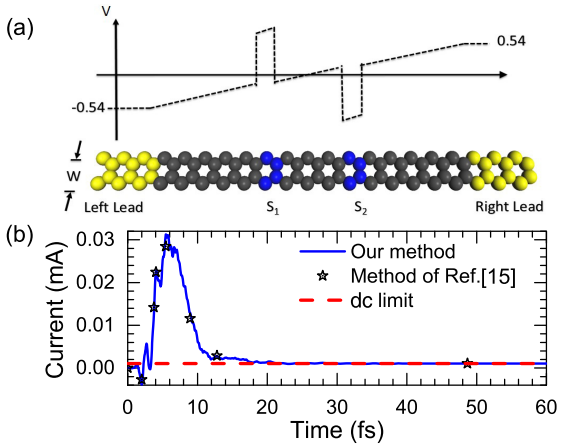

FIG. 2. (a) Configuration of the gated graphene nanoribbon. The S_1 and S_2 gates are of values 0.81 and -0.81V respectively. (b) Transient current of zigzag graphene nanoribbon for a system of 600 atoms. The dashed line is the dc limit.

is needed to reach the dc limit. The oscillatory behavior is due to resonant states in the system.

Now we test the scaling of our algorithm by calculating the transient current for nanoribbons with different system sizes ranging from 600 to 10 200 atoms. We first test algorithm I. Computational time of transient current for three time steps against system sizes N is shown in Fig. 4(a). We have fitted the data using $50N^3 + TN^2$ with very good agreement showing TN^2 scaling for the time-dependent part. For comparison, we have also plotted the computation time using method in Ref. [15]. We found that the number of energy points N_E depends on the spectrum of resonant states of the system. For graphene nanoribbons with 600 atoms, we have used $N_E = 6000$ to converge the integral over Fermi function. Figure 4(a) shows that a speed up factor of 1000 T is achieved at $N = 2400$. The scaling is shown in Fig. 4(b), from which we see that for $T < N$ the computational time is almost independent of the number of time steps.

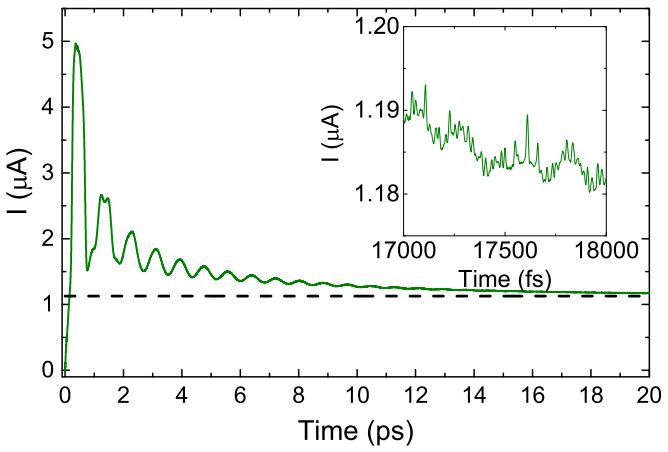

FIG. 3. Transient current of zigzag graphene nanoribbon for a system of 10 000 atoms at temperature of 300 K. The inset is the long-time behavior between 17 and 18 ps. The dashed line is the dc limit of transient current.

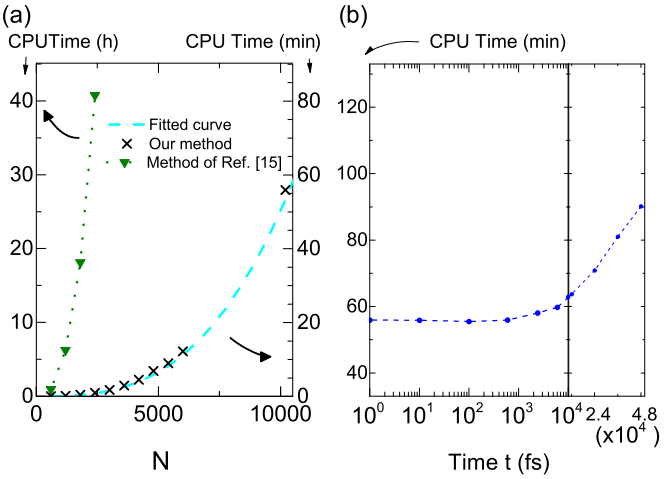

FIG. 4. (a) Scaling of computation time against N at $T = 3$ (each time step is 1 fs). The fitted curve in the form of $50N^3 + TN^2$ is in good agreement with the calculated results (Y axis is on the right). In order to compare the performance of Ref. [15], 6000 energy points were used for integration (Y axis is on the left). (b) Scaling of CPU time against t for $N = 10\,200$ ($t = 100$ fs corresponds to $T = 100$) using algorithm I. Left-hand side: exponential scale in t; Right-hand side: linear scale in t. These show that the computational time is nearly independent of time steps over a range of T but becomes proportional to it at an extremely large number of data points ranging over 10 thousand points.

Now we examine algorithm II which reduces the scaling TN^2 further. Notice that the scaling TN^2 comes from matrix multiplication involving Vandermonde matrix $V^t M_1$. A fast algorithm is available to speed up the calculation involving a structured matrix such as the Vandermonde matrix. As discussed in detail in Sec. III, we can use FMM [20,21] and FFT to carry out the same matrix multiplication using only $\kappa_1 N^2 + \kappa_2 N^2 \log_2 N$ operations provided $T < N^2$. Here the coefficient κ_1 is a large constant that depends only on the tolerance of the calculation τ and the setup of FMM. The theoretical estimate of this coefficient is about $40\log_2(1/\tau)$ where τ is the tolerance [32] in the FMM calculation in which we used $\tau = 10^{-4}$. When implementing FMM, this coefficient is in general larger than the theoretical one.

To test algorithm II, we have calculated the transient current numerically for $N = 10^4$ and $T = 10^8$ using FMM and FFT. The configuration of the system is the same as that appearing in the main text (Fig. 1) except the width W of the system is now 17 times wider with a total of 10 200 ($\sim 10^4$) atoms. The time step is 0.012 fs. The computed transient current using algorithms I and II are shown in Fig. 5. The purpose of this calculation is to test the computational complexity only. All we need to do is to compute

$$\bar{S}(\omega_T^l) = \omega_T^{-l} \sum_{j=0}^{N^2-1} \frac{u_j\left(1 - d_j^T\right)}{(1/\omega_T)^l - d_j} \tag{18}$$

using FMM and then take FFT to obtain $I(t_j)$ where u_j and d_j have been defined just before Eq. (16). Note that u_j has been obtained in the time-independent calculation.

If $(1/\omega_T)^j$ and d_j in Eq. (18) are uniformly distributed on the complex plane, the FMM can be done much faster.

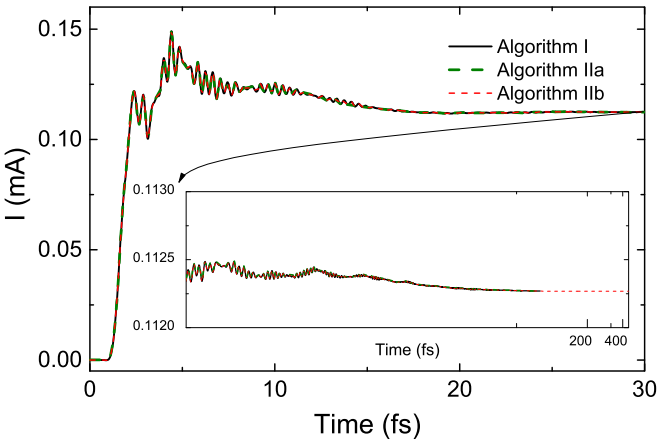

FIG. 5. Transient current calculated by algorithms I, IIa, and IIb; IIa and IIb refer to the cases with FMM methods targeting $T = N$ and $T = N^2$ respectively.

However, as shown in Figs. 6 and 7, the unit distribution of $(1/\omega_T)^j$ and d_j are highly nonuniform in our case. Actually, $(1/\omega_T)^j$ are distributed nonuniformly along the circle (Fig. 6) while d_j are distributed in a sector of unit circle (Fig. 7). This makes the calculation more difficult. For $N = 10^4$ and $T = N^8$, we found that the optimum number of levels in FMM is 10. With ten levels in FMM, over 60% of CPU time was spent on direct sum in FMM calculation.

In Fig. 5, we have tested algorithm IIa which is suitable for $T = N$ and algorithm IIb designed for $T = N^2$ against algorithm I. For $T < N$, the results of algorithm I, algorithms IIa and IIb, are on top of each other. For $T > N$, the calculation was done for $T = N^2$. In Fig. 5, we only show the results for $T < 40\,000$. There is no significant feature in the transient current plot beyond that.

Denote t_1 the CPU time needed for the spacial calculation (order N^3), t_2 needed for the temporal part [matrix multiplication in Eq. (7)]. Using a workstation of Xeon X5650 with

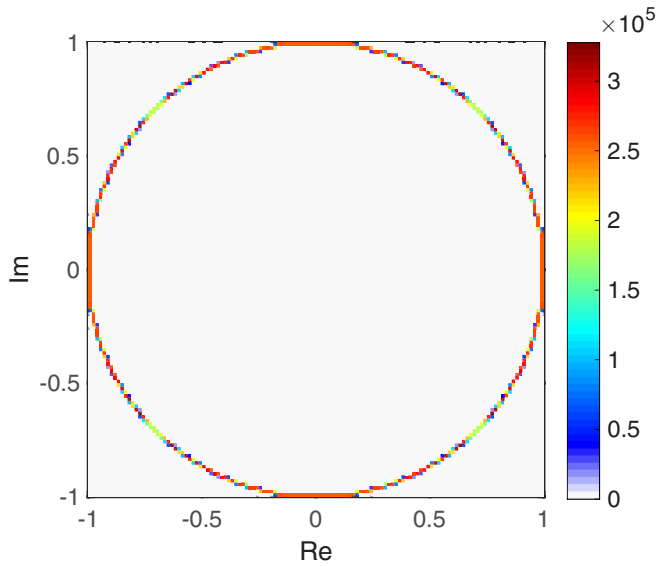

FIG. 6. Distribution of $(1/\omega_T)^j$ on the complex plane.

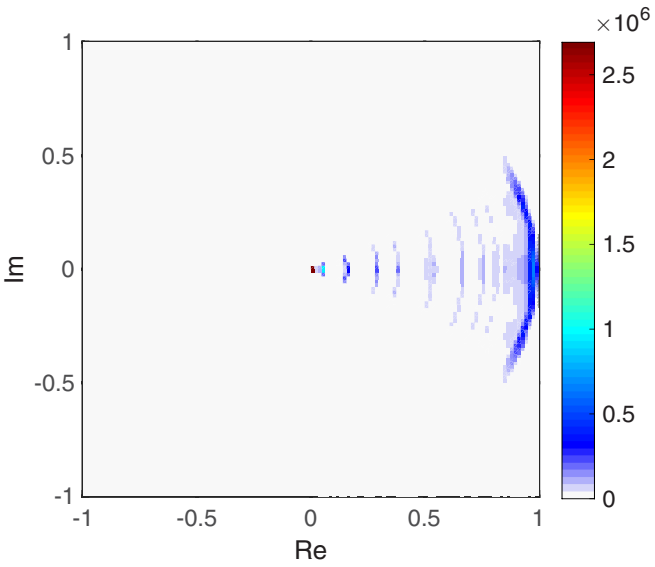

FIG. 7. Distribution of d_j on the complex plane.

12 cores and frequency 2.67 GHz, we obtained $t_1 = 3500$ s using 12 cores and $t_1 = 33\,800$ s using a single core so the efficiency of multithreading is about 80%. For FMM calculation, multithreading could be very inefficient and we have used a single core to perform the calculation. We found $t_2 = 3400$ s for $T = 10^8$ using a single core. We see that for $T = 10^8$ the time spent in the time-dependent part is about one-tenth of the time-independent calculation. This confirms that our method is of order $O(1)$ as long as $T < N^2$. We wish to point out that algorithm II is aimed to calculate the transient current $I(t)$ with time steps $T = N^2$ at one shot with scaling $N^2 \log_2 N$. This scaling remains if we want $I(t)$ with the number of time step less than N^2.

V. DISCUSSION AND CONCLUSION

Since our algorithm is based on the NEGF-CAP formalism, it could be extended to the NEGF-DFT-CAP formalism which performs the first-principles calculation. With the fast algorithm at hand, many applications can be envisaged. For instance, the transient spin current (related to spin transfer torque) using the NEGF-DFT-CAP formalism could be carried out for planar structures where k sampling in the first Brillouin zone is needed. It is straightforward to include k sampling in our method. It is also possible to extend this method to the case when electron-phonon interactions in the Born approximation as well as other dephasing mechanism are present [33]. Finally, first-principles transient photoinduced current on two-dimensional layered materials could be calculated using our method.

ACKNOWLEDGMENTS

This work was financially supported by the Innovation and Technology Commission of the HKSAR (Grant No. ITS/217/14), the University Grant Council (Contract No. AoE/P-04/08) of the Government of HKSAR, and NSF-China under Grant No. 11374246.

APPENDIX A: PADÉ APPROXIMANT

Brute force integration over Fermi function along real energy axis to obtain $G^<(t,t)$ may need thousands of energy points to converge which is very inefficient. To obtain an accurate result while reducing the cost, fast converging Padé spectrum decomposition (PSD) is used for the Fermi function f in Eq. (4) in the main text so that the residue theorem can be applied.

Using the $[n-1/n]$ PSD scheme [26] with the Padé approximant accurate up to $O[(\epsilon/kT)^{4n-1}]$, the Fermi function f can be expressed as

$$f(\epsilon) = \frac{1}{2} - \sum_{j=1}^{n} \frac{2\eta_j \beta\epsilon}{(\beta\epsilon)^2 + \xi_j{}^2}, \tag{A1}$$

where ξ_j and η_j are two sets of constants that can be calculated easily. Using the PSD scheme the analytic form of $G^<$ in Eq. (4) of the main text can be obtained using the residue theorem.

APPENDIX B: CALCULATION OF THE SPECTRAL FUNCTION

We express $\tilde{G}^r(\epsilon)$ and $\overline{G}^r(\epsilon)$, the equilibrium and nonequilibrium retarded Green's functions, respectively in terms of their eigenfunctions by solving the following eigenequations for $H_{\rm eq}$ and $H_{\rm neq}$ [15], i.e.,

$$(H_{\rm eq} - iW)\psi_n^0 = \epsilon_n^0 \psi_n^0, \quad (H_{\rm eq} + iW^\dagger)\phi_n^0 = \epsilon_n^0 \phi_n^0, \tag{B1}$$

where $W = \sum_\alpha W_\alpha$ and similar equations can be defined for $H_{\rm neq}$. Using the eigenfunctions of $H_{\rm eq} - iW$ and $H_{\rm neq} - iW$, we have

$$\tilde{G}^r(\epsilon) = [\epsilon - H_{\rm eq} + iW]^{-1} = \sum_n \frac{|\psi_n^0\rangle\langle\phi_n^0|}{(\epsilon - \epsilon_n^0 + i0^+)}, \tag{B2}$$

$$\overline{G}^r(\epsilon) = [\epsilon - H_{\rm neq} + iW]^{-1} = \sum_n \frac{|\psi_n\rangle\langle\phi_n|}{(\epsilon - \epsilon_n + i0^+)}. \tag{B3}$$

Performing the integral over ω using the residue theorem, the analytic solution of A_α is obtained,

$$A_\alpha(\epsilon,t) = \sum_n \frac{|\psi_n\rangle\langle\phi_n|}{\epsilon + \Delta_\alpha - \epsilon_n + i0^+} + \sum_n \frac{e^{i(\epsilon+\Delta_\alpha-\epsilon_n)t}|\psi_n\rangle\langle\phi_n|}{\epsilon - \epsilon_n + i0^+}\left[\frac{\Delta_\alpha}{\epsilon + \Delta_\alpha - \epsilon_n + i0^+} - \Delta\sum_l \frac{|\psi_l^0\rangle\langle|\phi_l^0|}{\epsilon - \epsilon_l^0 + i0^+}\right], \tag{B4}$$

where $\Delta = H_{\rm neq} - H_{\rm eq}$.

APPENDIX C: CALCULATION OF THE TRANSIENT CURRENT

Starting from Eq. (1) and in analog to Eq. (6) of the main text, the expressions of the current in Eq. (7) of the main text can be obtained as follows:

$$I_{0L}(t_j) = 2{\rm ReTr}\left[\frac{i}{\pi}\overline{\Gamma}_L H_{\rm neq} B_1 \overline{\Gamma}_L\right],$$

$$M_1 = {\rm ReTr}\left[\frac{i}{\pi}\overline{\Gamma}_L[2H_{\rm neq} - (\epsilon_n - \epsilon_m^*)]\left(\bar{B}_2 + \sum_\alpha f(\epsilon_n - \Delta_\alpha)\bar{B}_{4\alpha}\right)\overline{\Gamma}_L\right],$$

$$M_{2\alpha} = {\rm ReTr}\left[\frac{i}{\pi}\overline{\Gamma}_L[2H_{\rm neq} - (\tilde{\epsilon}_l - \epsilon_m^* + \Delta_\alpha)][f(\epsilon_m^*)\bar{B}_{3\alpha}]\overline{\Gamma}_L\right].$$

The expression of the transient current $I_R(t)$ is similar to Eq. (7).

[1] N. S. Wingreen, A.-P. Jauho, and Y. Meir, Phys. Rev. B **48**, 8487(R) (1993).

[2] J. Wang, J. Comput. Electron. **12**, 343 (2013).

[3] S. Kurth, G. Stefanucci, C.-O. Almbladh, A. Rubio, and E. K. U. Gross, Phys. Rev. B **72**, 035308 (2005).

[4] G. Stefanucci, S. Kurth, A. Rubio, and E. K. U. Gross, Phys. Rev. B **77**, 075339 (2008).

[5] Y. Zhu, J. Maciejko, T. Ji, H. Guo, and J. Wang, Phys. Rev. B **71**, 075317 (2005).

[6] J. Maciejko, J. Wang, and H. Guo, Phys. Rev. B **74**, 085324 (2006).

[7] R. Tuovinen, E. Perfetto, G. Stefanucci, and R. van Leeuwen, Phys. Rev. B **89**, 085131 (2014).

[8] R. Seoane Souto, R. Avriller, R. C. Monreal, A. Martín-Rodero, and A. Levy Yeyati, Phys. Rev. B **92**, 125435 (2015).

[9] X. Zheng, F. Wang, C. Y. Yam, Y. Mo, and G. H. Chen, Phys. Rev. B **75**, 195127 (2007).
[10] L. Zhang, Y. Xing, and J. Wang, Phys. Rev. B **86**, 155438 (2012).
[11] B. Gaury, J. Weston, M. Santin, M. Houzet, C. Groth, and X. Waintal, Phys. Rep. **534**, 1 (2014).
[12] M. Ridley, A. MacKinnon, and L. Kantorovich, J. Phys.: Conf. Ser. **696**, 012017 (2016).
[13] A. Croy and U. Saalmann, Phys. Rev. B **80**, 245311 (2009).
[14] J. Weston and X. Waintal, Phys. Rev. B **93**, 134506 (2016).
[15] L. Zhang, J. Chen, and J. Wang, Phys. Rev. B **87**, 205401 (2013).
[16] Z. Y. Ning, Y. Zhu, J. Wang, and H. Guo, Phys. Rev. Lett. **100**, 056803 (2008).
[17] J. D. Burton and E. Y. Tsymbal, Phys. Rev. Lett. **106**, 157203 (2011).
[18] D. Waldron, V. Timoshevskii, Y. B. Hu, K. Xia, and H. Guo, Phys. Rev. Lett. **97**, 226802 (2006).
[19] The system size N equals to the dimension of the Hamiltonian matrix. For the tight-binding model, N equals the number of lattice sites multiplied by the number of orbitals per site.
[20] V. Rokhlin, J. Comput. Phys. **60**, 187 (1985).
[21] J. Song, C. C. Lu, and W. C. Chew, IEEE Trans. Antennas Propag. **45**, 1488 (1997).
[22] On the Hartree level, we have $\nabla^2 U(x,t) = -4\pi i G^<(x,x,t,t)$, where $G^<(x,x,t,t)$ is the diagonal matrix element of lesser Green's function in the scattering region. In this paper, we avoid solving this time-dependent equation by assuming that the response of internal potential is instantaneous.
[23] Here $\partial_t G^<(t,t) \equiv [\partial_{t_1} G^<(t_1,t_2) + \partial_{t_2} G^<(t_1,t_2)]_{t_1=t_2}$.
[24] J. Taylor, H. Guo, and J. Wang, Phys. Rev. B **63**, 245407 (2001).
[25] J. A. Driscoll and K. Varga, Phys. Rev. B **78**, 245118 (2008).
[26] J. Hu, R.-X. Xu, and Y. Yan, J. Chem. Phys. **133**, 101106 (2010).
[27] B. A. Cipra, SIAM News **33**, 2 (2000).
[28] A. H. Castro Neto, N. M. R. Peres, K. S. Novoselov, and A. K. Geim, Rev. Mod. Phys. **81**, 109 (2009).
[29] Y. Zhang, Y.-W. Tan, H. L. Stormer, and P. Kim, Nature (London) **438**, 201 (2005).
[30] E. Perfetto, G. Stefanucci, and M. Cini, Phys. Rev. B **82**, 035446 (2010).
[31] H. Xie, Y. Kwok, Y. Zhang, F. Jiang, X. Zheng, Y. Yan, and G. Chen, Phys. Status Solidi **250**, 2481 (2013).
[32] N. Yarvin and V. Rokhlin, SIAM J. Numer. Anal. **36**, 629 (1999).
[33] N. Säkkinen, Y. Peng, H. Appel, and R. van Leeuwen, J. Chem. Phys. **143**, 234102 (2015).