Parallelization of the Two-Dimensional Wigner Monte Carlo Method

Josef Weinbub(✉), Paul Ellinghaus, and Siegfried Selberherr

Institute for Microelectronics, TU Wien, Vienna, Austria
{weinbub,ellinghaus,selberherr}@iue.tuwien.ac.at

Abstract. A parallelization approach for two-dimensional Wigner Monte Carlo quantum simulations using the signed particle method is introduced. The approach is based on a domain decomposition technique, effectually reducing the memory requirements of each parallel computational unit. We depict design and implementation specifics for a message passing interface-based implementation, used in the Wigner Ensemble Monte Carlo simulator, part of the free open source ViennaWD simulation package. Benchmark and simulation results are presented for a time-dependent, two-dimensional problem using five randomly placed point charges. Although additional communication is required, our method offers excellent parallel efficiency for large-scale high-performance computing platforms. Our approach significantly increases the feasibility of computationally highly intricate two-dimensional Wigner Monte Carlo investigations of quantum electron transport in nanostructures.

1 Introduction

The Wigner formalism [11] provides an attractive alternative to the non-equilibrium Green's function formalism, as it provides a reformulation of quantum mechanics - usually defined through operators and wave functions - in the phase space using functions and variables [6]. Thereby, the Wigner formalism provides a more intuitive description which also facilitates the reuse of many classical concepts and notions. Several methods have been applied to solve the Wigner equation of which the stochastic Wigner Monte Carlo method, using the signed-particle technique [4,5], has emerged as probably the most promising approach: it has made multi-dimensional Wigner simulations viable for the first time [8]. An efficient distributed parallel computation approach is the next crucial step to facilitate the use of Wigner simulations to investigate actual devices.

Wigner Monte Carlo simulations have been made computationally feasible by the annihilation step, required to counterbalance the continuous generation of particles [7]. However, the memory demand of the annihilation algorithm itself is proportional to the dimensionality and resolution of the phase space represented in the simulation, which can lead to exorbitant requirements.

All in all, Wigner Monte Carlo quantum simulations suffer not just from compute intensive operations but also from vast memory demands; the latter

I. Lirkov et al. (Eds.): LSSC 2015, LNCS 9374, pp. 309–316, 2015.
DOI: 10.1007/978-3-319-26520-9_34

is much more severe, as more realistic simulations are beyond reach on single workstations with their limited memory.

Conventional parallelization approaches for Monte Carlo methods, using domain replication, are *embarrassingly parallel* [1]. The particle ensemble is split amongst computational units, where each sub-ensemble is treated completely independently. This necessitates domain replication, when working in a distributed-memory context (as is the de facto standard for large-scale parallel computations) to avoid additional communication. Such an approach offers excellent parallel efficiency, however, domain replication is not feasible for the Wigner Monte Carlo method due to the huge memory demands associated with the annihilation algorithm, quickly exceeding the typically available memory on a single computational unit. Further contributing to the challenge of implementing scalable Wigner Monte Carlo simulations is the fact that particle annihilation must be performed in unison across the global simulation domain [2]; a synchronization step between each individual time step is required, impeding parallel efficiency.

We present a message passing interface (MPI)-based domain decomposition approach for two-dimensional problems, which avoids domain replications and thus drastically reduces the memory requirements for each parallel computational unit. This work extends previous investigations of one-dimensional problems [2] to two-dimensional scenarios. Our approach to partition the simulation domain and to accelerate the overall simulation process is discussed for the Wigner Ensemble Monte Carlo simulator, which is part of the free open source ViennaWD simulation package [10]. The parallel efficiency is evaluated based on the execution times of a representative time-dependent, two-dimensional example using randomly placed point charges. With our approach we not only tackle the challenge of reducing simulation times, but much more importantly, we enable to conduct highly memory-intensive Wigner Monte Carlo quantum simulations in the first place.

2 Parallel Algorithm for Two-Dimensional Problems

The parallelization strategy for two-dimensional problems is based on previous investigations regarding one-dimensional problems [2]. The domain decomposition approach entails splitting up the spatial domain amongst processes. Each process represents a subdomain (i.e. a part of the global domain) and only treats particles, which fall within its own subdomain. Thereby, the memory requirements to represent the phase-space, and all other space-dependent quantities, are scaled down with the number of processes (subdomains) used. As the particle ensemble evolves, the particles travel between subdomains. This necessitates an inter-MPI process communication layer, representing spatially neighboring subdomains; a centralized communication where all worker processes transfer data via a single master process is avoided, which would significantly limit parallel scalability.

The applied domain decomposition technique assigns each MPI process to a unique subdomain by splitting the simulation domain uniformly, as illustrated

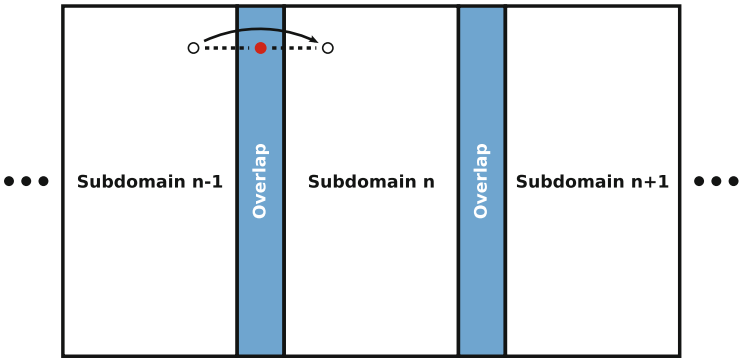

Fig. 1. The simulation domain is splitted uniformly. Each subdomain is assigned to a separate MPI process. If a particle (black circle) enters the overlap area (red), it is transmitted to the neighboring subdomain (Color figure online).

in Fig. 1. A so-called *slab* or one-dimensional decompositioning method is used to partition the simulation domain, meaning that one direction is partitioned, whereas the second direction (in a two-dimensional setting) is kept untouched. Although such a partitioning technique theoretically tends to limit the parallelization efficiency (e.g. the maximum number of utilizable MPI processes is limited to the number of grid elements in the direction of the partitioning, as one MPI process has to be at least responsible for one grid element), the method provides more than enough parallel processing potential for today's relevant problem scenarios (cf. Sect. 3). This is even more so, when the communication is aligned with the partitioning, meaning that the majority of particles primarily propagate in the unpartitioned direction, minimizing the need for communication which in itself further increases parallel scalability.

The subdomains are assigned to MPI processes in a sequential order, inherently providing an MPI/subdomain neighbor-identification mechanism. The primary MPI communication consists of non-blocking direct neighbor communication. After each time step a lightweight message is used to globally trigger an annihilation step within each MPI process. The transfer (communication) of particles between processes only occurs once at the end of each time-step. This necessitates a small overlap between adjacent subdomains, which serves a similar purpose as a *ghost layer* used in conventional domain decomposition techniques [3]. The overlap is used to identify particles traveling towards a neighboring subdomain, which ultimately get transferred to the respective neighbor subdomain at the end of every time step. A larger overlap between subdomains simplifies the transfer of particles between processes (as particles are required to be transferred less often), however, this introduces a larger data redundancy, which negatively affects parallel efficiency. The exact extent of the overlap is a simulation parameter which it should consider the maximum distance a particle can travel within the chosen time-step as well as its direction of travel.

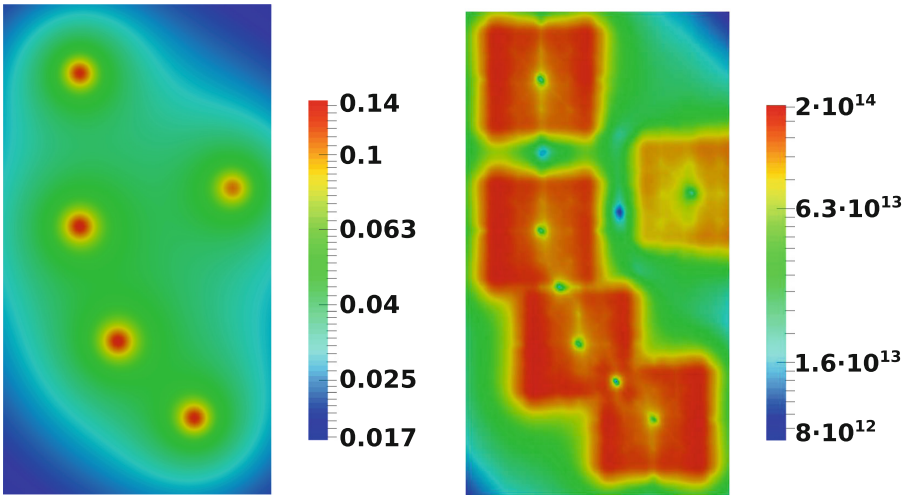

Fig. 2. The potential barrier profile V [eV] (**left**) and the corresponding particle generation rate $\gamma\,[s^{-1}]$ (**right**) is shown for the simulation domain (Color figure online).

3 Results

This section investigates a time-dependent, two-dimensional problem (rectangular simulation domain, spatial dimensions are 70 nm × 128 nm) with respect to parallel execution performance. The total number of particles is limited to $32 \cdot 10^7$ particles; the simulation is initialized with $3 \cdot 10^3$ particles. Reflective boundary conditions are used for all boundaries, meaning that no particles leave the simulation domain. The coherence length is 30 nm and the lattice temperature is 300 K. The system is simulated for 200 fs using a 0.5 fs time step.

Five point charges are spread over the simulation domain, each giving rise to particle generation, concentrated within half of the coherence length around it (Fig. 2). The simulation is parallelized via 16, 32, 64, and 128 MPI processes using the VSC-2 supercomputer [9]. One VSC-2 computational node provides 16 cores (two 8-core AMD Opteron Magny Cours 6132HE 2.2 GHz) and 32 GB of system memory; the nodes are connected via an InfiniBand QDR network.

Figure 3 shows the parallel execution performance of our approach which gives an almost perfect, linear parallel scalability. For this setup, a simulation which would otherwise take around 9.3 h (extrapolated, assuming linear scaling relative to 16 MPI processes), takes about 35 min when using 16 MPI processes or around 5 min when using 128 MPI processes. This fact clearly shows the significance of our parallelization approach as it drastically accelerates the simulation process, allowing to considerably increase the pace of research.

In comparison to earlier investigations regarding one-dimensional problems [2], our domain decompositioning approach works even better for two-dimensional

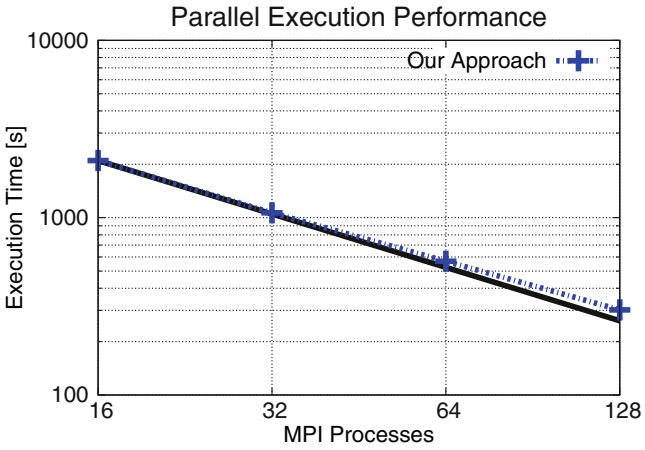

Fig. 3. The execution performance of our two-dimensional parallelization approach (dashed, blue line), shown relative to ideal scalability (black line), offers an almost perfect parallel scaling behavior (Color figure online).

cases and increased particle numbers. The workload per MPI process drastically increases, outweighing any potential communication overhead. In our two-dimensional investigations, we capped the total number of particles at $32 \cdot 10^7$ particles, as compared to $8 \cdot 10^6$, $16 \cdot 10^6$, and $32 \cdot 10^6$ particles in our previous one-dimensional investigations. We, therefore, allow around an order of magnitude more particles to take part in the simulation. An increase in the number of particles is also required in two dimensions to increase the statistical confidence, since the phase space is bigger. We compute the maximum number of particles for each MPI process by dividing the maximum size by the number of MPI processes. Therefore, in the case of using 128 MPI processes each process is responsible for at most $25 \cdot 10^5$ particles, which offers enough workload per process to outweigh the communication overhead required after each time step. Also, the presence of several point charges gives rise to a very high generation rate; the entire simulation domain is rather quickly populated with particles, inherently increasing the load balance over all MPI processes.

Figure 4 depicts the total number of particles (the sum of positively and negatively signed particles) for different time steps as computed via a parallelized simulation. Over time, the entire simulation domain is flooded with particles, however, local maxima occur around the point charges. The maxima take on a rectangular shape (spatial dimensions correspond to the coherence length) due to the taken implementation of the Wigner potential calculation for a given point. While the total number of particles gives an indication of the computational load, the positive and negative particles compensate each other when calculating physically meaningful quantities, like the density shown in Fig. 5.

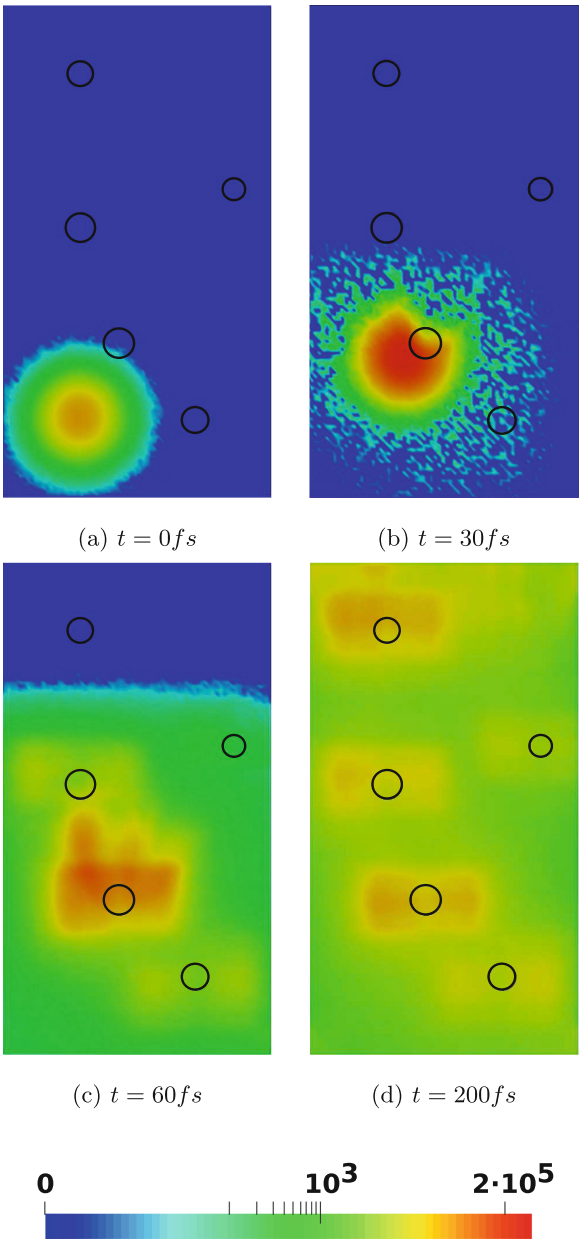

Fig. 4. Number of particles (positive+negative) for different time steps (a-d). The initial wave package (a) propagates upwards and slightly to the right. Reflecting boundary conditions are used for all four boundaries. Black circles indicate point charges (Color figure online).

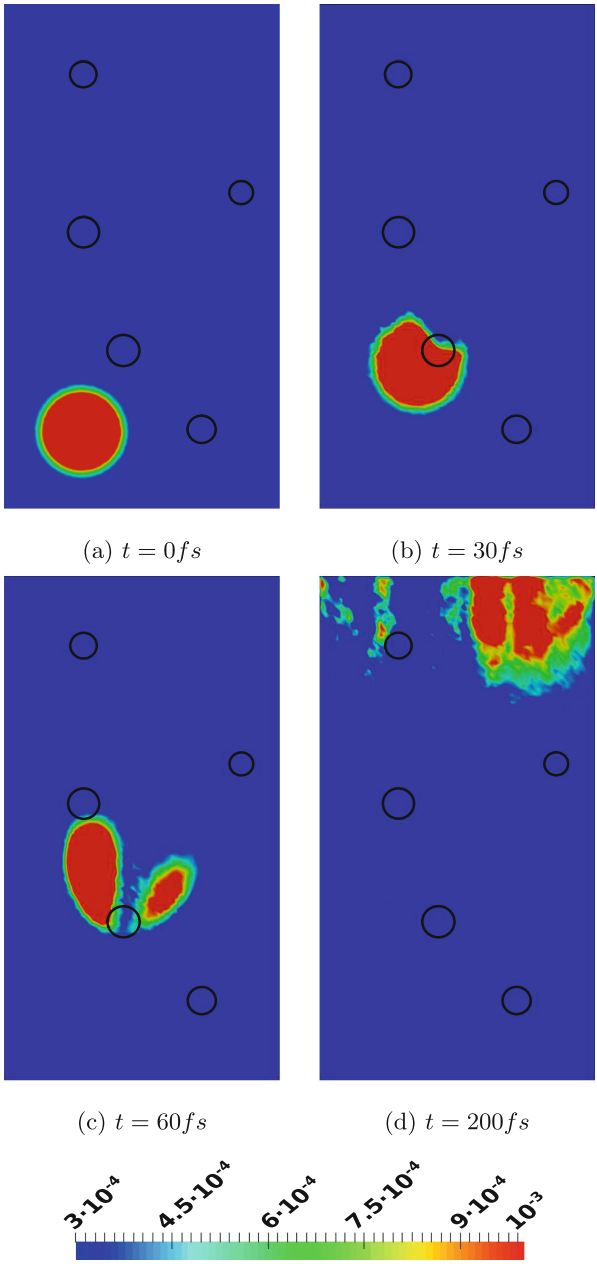

Fig. 5. Normalized density (expressed as a probability) for different time steps (a-d). The initial wave package (a) propagates upwards and slightly to the right. Reflecting boundary conditions are used for all four boundaries. Black circles indicate point charges (Color figure online).

4 Summary

Our approach for parallelizing computationally highly demanding time-dependent, two-dimensional quantum Wigner Monte Carlo simulations has been presented in the context of the MPI-based Wigner Ensemble Monte Carlo simulator, part of the free open source ViennaWD simulation package. The approach uses a domain decomposition technique to distribute the workload among the MPI processes. The conceptual approach for the parallelization technique has been discussed as well as the setup and results of a two-dimensional simulation example. A benchmark depicting the parallel execution performance for 16, 32, 64, and 128 MPI processes shows an almost perfect, linear parallel scalability.

Acknowledgements. The computational results presented have been achieved using the Vienna Scientific Cluster (VSC). The authors thank Mihail Nedjalkov for valuable feedback.

References

1. Dimov, I.: Monte Carlo Methods For Applied Scientists. World Scientific Publishing, Singapore (2008). ISBN 9789810223298
2. Ellinghaus, P., Weinbub, J., Nedjalkov, M., Selberherr, S., Dimov, I.: Distributed-Memory parallelization of the Wigner Monte Carlo method using spatial domain decomposition. J. Comput. Electron. **14**(1), 151–162 (2015). doi:10.1007/s10825-014-0635-3
3. Hager, G., Wellein, G.: Introduction to High Performance Computing for Scientists and Engineers. CRC Press, Boca Raton (2010). ISBN 9781439811924
4. Nedjalkov, M., Kosina, H., Selberherr, S., Ringhofer, C., Ferry, D.K.: Unified particle approach to Wigner-Boltzmann transport in small semiconductor devices. Phys. Rev. B **70**, 115319-1–115319-16 (2004). doi:10.1103/PhysRevB.70.115319
5. Nedjalkov, M., Schwaha, P., Selberherr, S., Sellier, J.M., Vasileska, D.: Wigner quasi-particle attributes - an asymptotic perspective. Appl. Phys. Lett. **102**(16), 163113-1–163113-4 (2013). doi:10.1063/1.4802931
6. Querlioz, D., Dollfus, P.: The Wigner Monte-Carlo Method for Nanoelectronic Devices: Particle Description of Quantum Transport and Decoherence. Wiley, New York (2010). ISBN 9781848211506
7. Sellier, J.M., Nedjalkov, M., Dimov, I., Selberherr, S.: The role of annihilation in a Wigner Monte Carlo approach. In: Lirkov, I., Margenov, S., Waśniewski, J. (eds.) LSSC 2013. LNCS, vol. 8353, pp. 186–193. Springer, Heidelberg (2014). doi:10.1007/978-3-662-43880-0_20
8. Sellier, J., Nedjalkov, M., Dimov, I., Selberherr, S.: Two-dimensional transient Wigner particle model. In: Proceedings of the 18th International Conference on Simulation of Semiconductor Processes and Devices (SISPAD). pp. 404–407 (2013). doi:10.1109/SISPAD.2013.6650660
9. Vienna Scientific Cluster: VSC-2. http://vsc.ac.at/
10. ViennaWD: Wigner Ensemble Monte Carlo Simulator. http://viennawd.sourceforge.net/
11. Wigner, E.: On the quantum correction for thermodynamic equilibrium. Phys. Rev. Lett. **40**, 749–759 (1932). doi:10.1103/PhysRev.40.749