Ab initio non-equilibrium Green's function studies of electronic devices with several thousand atoms

Wenchang Lu and J. Bernholc

North Carolina State University, Raleigh, NC 27695-7518

Multi-probe non-equilibrium Green's function (NEGF) method has been implemented in our massively parallel DFT-based software, the real space multigrid (RMG) method. Multiple levels of parallelization are used to run efficiently on supercomputers: MPI between nodes, OpenMP in a node, and Cuda for GPU accelerators. Localized atom-centered orbitals, variationally optimized for the lead and conductor regions, respectively, serve as the basis set for NEGF calculations. We describe two applications:

(i) Graphene-based FET transistors

Two-dimensional materials, such as graphene and molybdenum disulfide, have attracted much attention because of their unique properties. Graphene's high mobility make it a very promising material for next generation electronics, but its zero band gap is a hurdle for digital transistors. However, graphene nanoribbons can exhibit band gaps due to quantum confinement, and their electronic properties differ depending on the structures of their edges. Here we systematically study transport properties of graphene-based transistors with up to 5,000 atoms, investigating the effects of nanoribbon index, length, width and gate structure. Surprisingly, we find the that mod(n,3)=0 armchair ribbons would perform best in FET configurations.

(2) Si nanowire p-n junction

With the implemented multi-level parallelization, we are able to simulate two-terminal systems with over 10 thousand atoms. As a first example, we calculated the I-V characteristics of a Si nanowire p-n junction. The nanowire is along the (110) direction in order to minimize the number dangling bonds on the surface. Its diameter is 3 nm. It turns out that due to the long screening length in Si, a channel length of ~24 nm is necessary to observe ideal diode characteristics: exponential current increase in forward bias and nearly zero current with backward bias.