International HPC Summer School 2017
2017 June. 26th

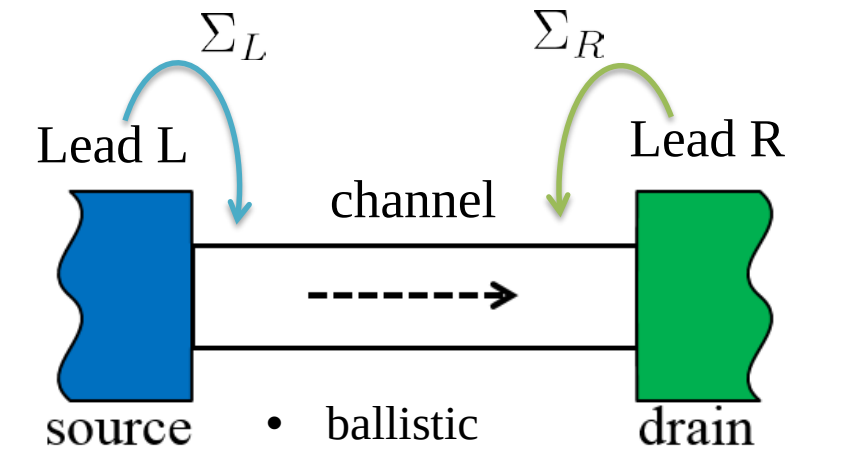
Linear scaling first-principles calculation method for electronic transport

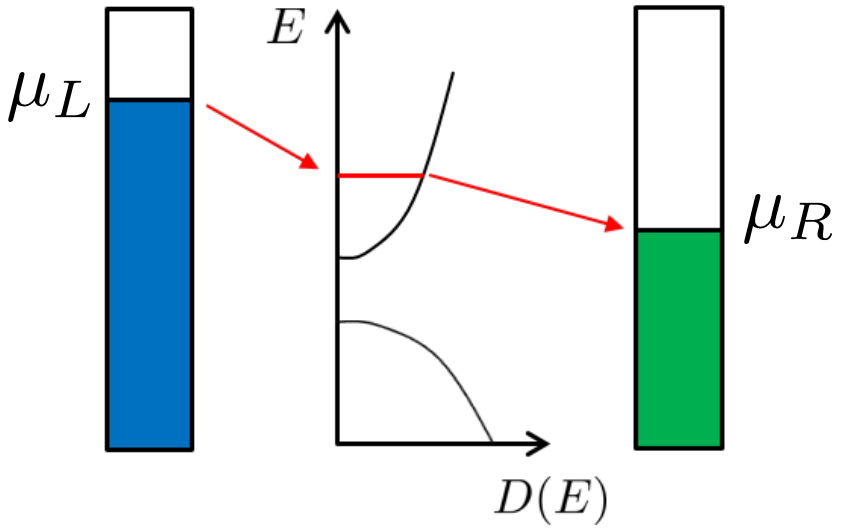
Masahiro FUKUDA, Taisuke OZAKI

Institute for Solid State Physics, University of Tokyo, Japan

Abstract:
Recent technology has achieved nano scale fabrication of the electronic devices. The electronic transport properties in such atomic scale materials, where quantum features are remarkable, are much different from that in macro scale materials. Therefore, electronic transport properties in nanoscale devices need to be investigated on the basis of the first-principles calculation, which is a method of numerical calculations of electronic structure based on quantum mechanics. However, since even small electronic devices consist of more than one million atoms, it goes without saying that HPC is necessary to write program codes to investigate their properties. Recently, we proposed a first-principles electronic transport calculation method whose computational cost is linear to the system size. I will explain the details of the electronic transportation calculation method and a program code of the hyper-parallel and high speed first-principles calculation for electronic transportation.

Electronic transport





The electronic transport in nano devices can be evaluated by non-equilibrium Green's function (NEGF) method.

- **Green's function**

$$G(E) = [ES - H(\rho) - \Sigma_L(E) - \Sigma_R(E)]^{-1}$$

Hamiltonian for DFT (first-principles calculation)

- **Charge current**

$$I \equiv \left.\frac{\partial N_{\text{channel}}}{\partial t}\right|_{\text{left}} = -\left.\frac{\partial N_{\text{channel}}}{\partial t}\right|_{\text{right}}$$

$$= \frac{2e}{h}\int T(E)\,(f_L(E) - f_R(E))\,dE$$

$$\approx \frac{2e}{h}\int T(E)\frac{\partial f}{\partial \mu}\,\underbrace{(\mu_L(E) - \mu_R(E))}_{\propto V_{\text{bias}}}\,dE$$

(*cf.* Ohm's law: $I = V/R$)

- **Transmission coefficient**

$$T(E) = \text{Tr}\left[\Gamma_L(E)G(E)\Gamma_R(E)G^{\dagger}(E)\right]$$

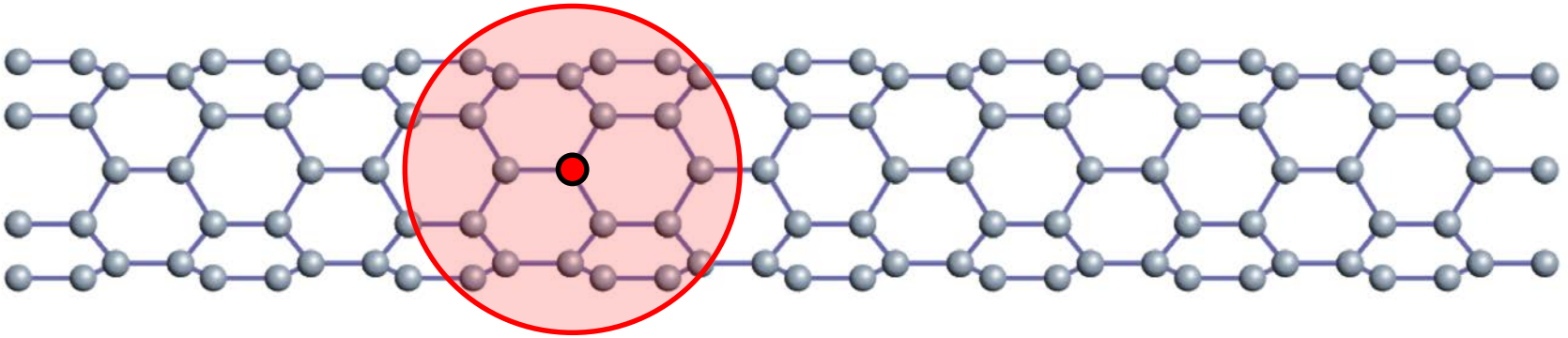
Divide-conquer method

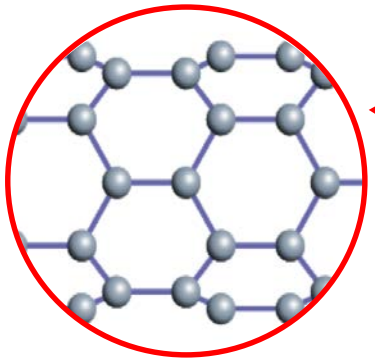
O(N) × O(N) matrix

N: System size
M: Cluster size
M<<N

$$G(E) = [ES - H(\rho) - \Sigma_L(E) - \Sigma_R(E)]^{-1}$$

: Computational cost ~ O(N^3)





$$G^{(i)}(E) = \left[ES^{(i)} - H^{(i)}(\rho) - \Sigma_L^{(i)}(E) - \Sigma_R^{(i)}(E) - \sum_j \Sigma_j^{(i)}(E) \right]^{-1}$$

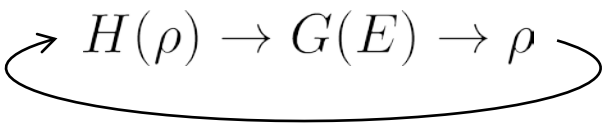
O(M) × O(M) matrix

$$G^{\mathrm{DC}}(E) = \sum_i^N G^{(i)}(E)$$: Computational cost ~ O(N)

In the divide-conquer method, we truncate a cluster and calculate its local Green's function for each cluster. Finally, all the local Green's functions are put together.

Algorisms for NEGF and DC-NEGF

ordinary NEGF



do SCF cycle
  solve Poisson equation
  do E
    construct InvG=ES-H-Σ_L-Σ_R
    calculate G(E)
    calculate ρ^{eq}(E)
    calculate $G^{<}$(E)
    calculate $\Delta\rho$(E)
  end do
end do

MPI parallelization about E

DC-NEGF

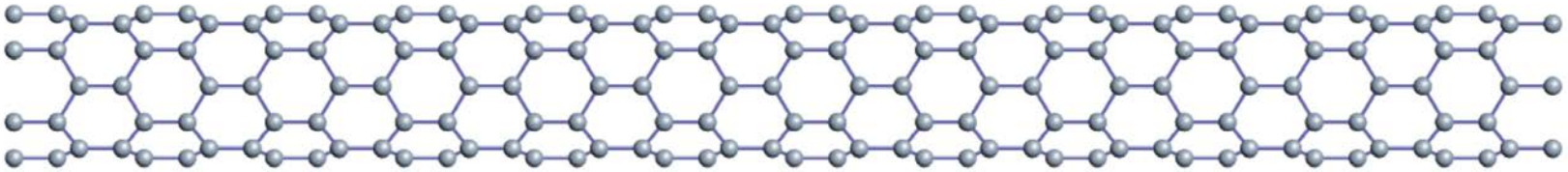
- The computational cost scales only linearly.
- Massively parallel computing is available.

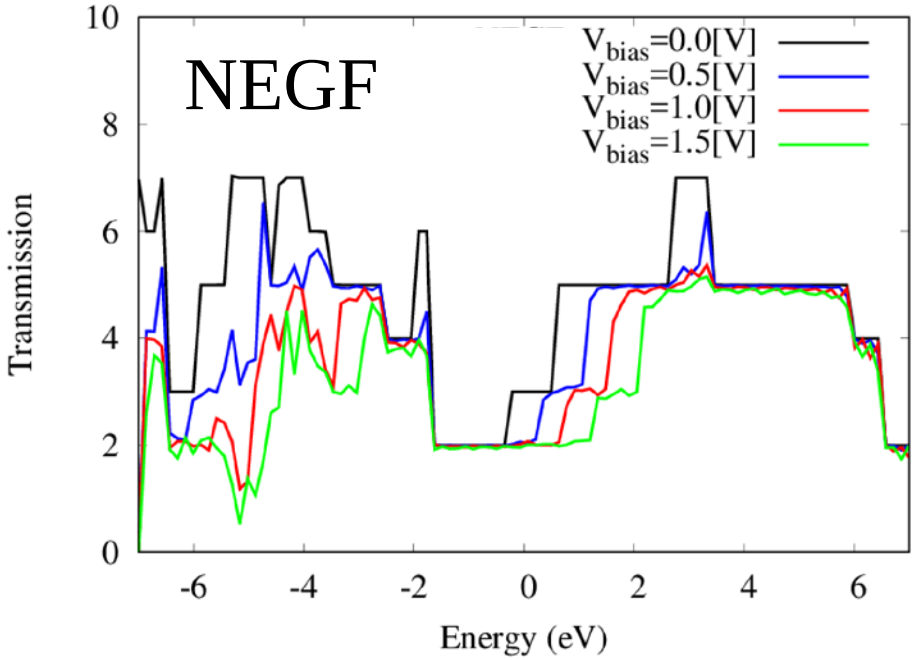
do SCF cycle
  solve Poisson equation
  do E
    construct InvG=ES-H-Σ_L-Σ_R
    do site i
      construct $InvG^{(i)}$
      construct $InvG^{(i)}$-$\Sigma^{(i)}$
      calculate $G^{(i)}$(E)
      calculate $\rho^{eq(i)}$(E)
    end do
    construct G(E) and ρ^{eq}(E)
    calculate $G^{<}$(E)
    calculate $\Delta\rho$(E)
  end do
end do

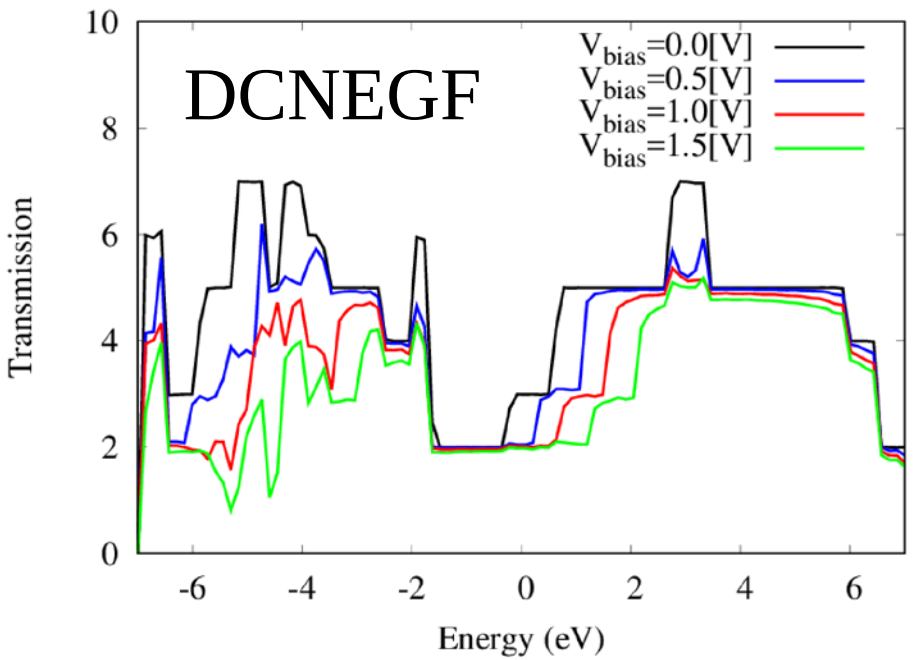
MPI parallelization about E and clusters

Transmission of carbon nanotube

program code : OpenMX (http://www.openmx-square.org/)







NEGF: 60sec/SCF by 200 processes
(Max parallelization : 200)

DCNEGF: 5sec/SCF by 3600 processes
(Max parallelization : 57600)