Report on changes made to the program [2]

Juris Vencels

juris.vencels@gmail.com

`https://github.com/jvencels/bari-montecarlo/tree/linux-build`

August 31, 2013

Contents

1 Changes made to the program

- Particles are stored as linked list, it allows to store variable number of particles.
- Structure of the input file. Example will be provided in Section 3.
- Program handles two cases: calculation of the stationary electron energy distribution function *eedf* and dynamic change of the *eedf* function during a time. For details concerning initialization of each case look in the input file example in Section 3.
- Implemented dynamic particle initialization function. For details look in Section 5.1.

2 Electron energy distribution and probability functions

2.1 Normalization of the *eedf*

Normalization of the *eedf* function $f(\epsilon)$, where ϵ is particle energy, is done using equation (3) in [1]. In the final form

$$\int f(\epsilon) \cdot \epsilon^{\frac{1}{2}} \; d\epsilon = 1. \tag{1}$$

In the program normalization of discrete $f(\epsilon_i)$ implemented as

$$f(\epsilon_i) = \frac{f(\epsilon_i)}{\sum\limits_i \left(f(\epsilon_i) \cdot \epsilon_i^{\frac{1}{2}} \cdot \Delta\epsilon \right)}. \tag{2}$$

2.2 Normalization of the electron energy probability function

Function represents the distribution of electrons over energy bins. In the program it is calculated as

$$p(\epsilon_i) = \frac{p(\epsilon_i)}{\sum\limits_i p(\epsilon_i)}. \tag{3}$$

3 Input data example

```
0                      | Simulation mode (0 = stationary, 1 = non-stationary)
1e6                    | Particle initialization expression (e.g., 10^11*t+1e7)
                         In stationary mode skips time dependent part of expression,
                         initializes only constant (e.g., 1e7)
50e-9                  | Time of simulation [s]
                         In stationary mode affects iteration number
.1e-9                  | Time step [s]
1e3                    | Initial energy of electrons [eV]
50e-3                  | Gas pressure [Torr]
7 2                    | Number of collision processes, particle species
1.53e7 2.74e7          | A(b), A(c)
8.84e-10 3.67e-11      | Q(b), Q(c)
"data/N2+B.dat"
"data/N2C-2.dat"
"data/N2.dat"
"data/N2A3.dat"
"data/N2B3.dat"
"data/N2Bprime3.dat"
"data/N2W3u.dat"
```

4 Stationary case

4.1 Convergence criteria

Electron energy probability function is normalized $\sum_{i=1}^{N} p(\epsilon_i) = 1$. Convergence criteria for time step t is

$$C^t = \sum_{i=1}^{N} |p^t(\epsilon_i) - p^{t-\Delta t}(\epsilon_i)|, \tag{4}$$

where $t - \Delta t$ is previous time step. C may vary in range $[0, 1]$, where less value is better.

4.2 Results

All Figures except Figure 5 were got with the input shown in Section 3. In Figure 5 number of particles was changed to $1e7$. Average electron energy, total number of electrons, number of electrons with energy below 10 eV and number of electrons with initial energy (1 keV) shown in Figure 1. Normalized electron energy distribution function $f(\epsilon)$ over time shown in Figure 2. Normalized electron energy probability function $f(\epsilon)$ over time shown in Figure 3. Plots for different number of initialized particles that proofs convergence of the solution shown in Figures 4 and 5.

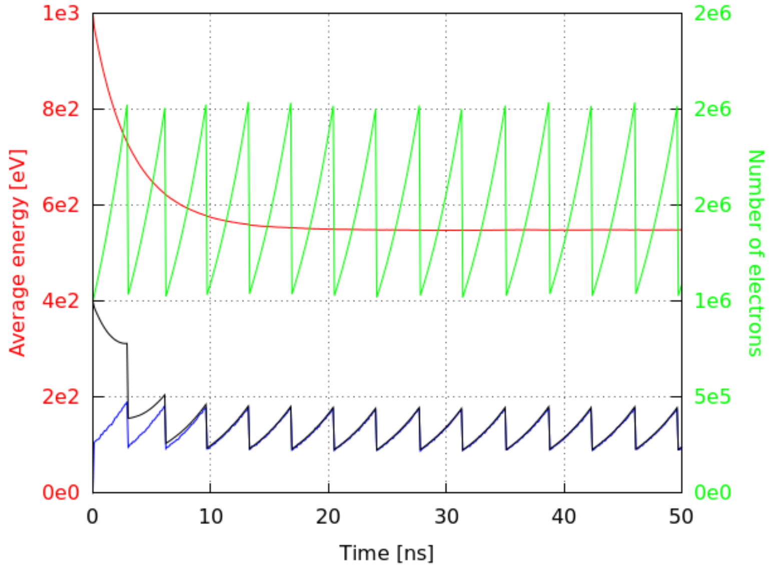

Figure 1: Average electron kinetic energy (red), total number of electrons (green), number of electrons with energy below 10 eV multiplied by 20 (blue) and number of electrons with initial energy (black).

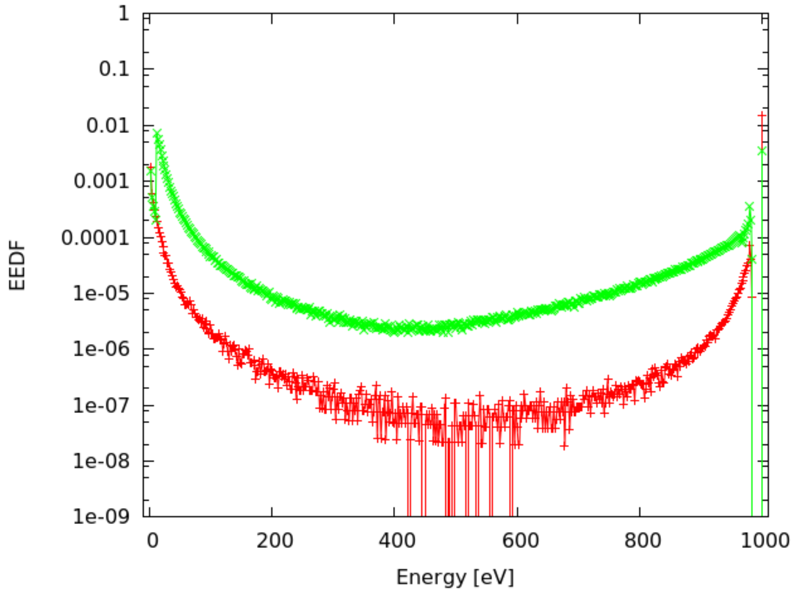

Figure 2: Evolution of normalized *eedf* after first (red) and last (green) time steps.

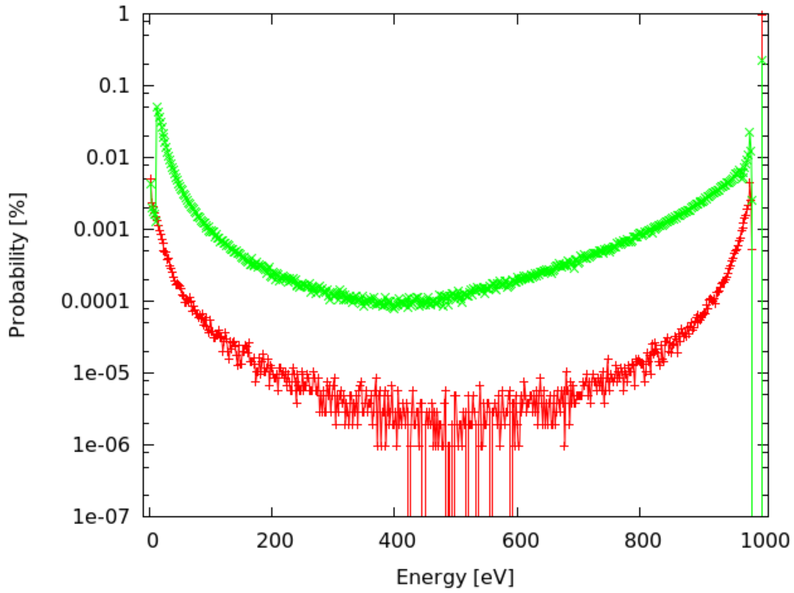

Figure 3: Evolution of the normalized electron probability after first (red) and last (green) time steps. Plot shows distribution of electrons in energy bins in the program.

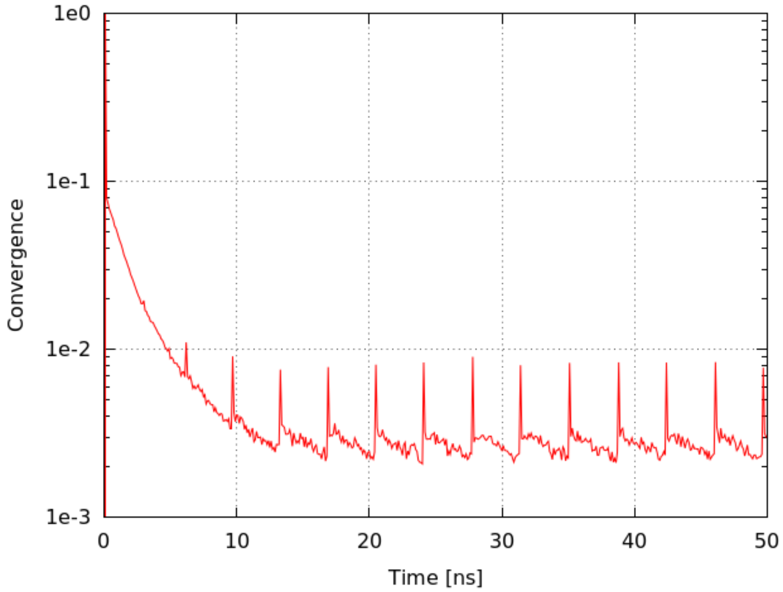

Figure 4: Convergence during time for 10^6 particles.

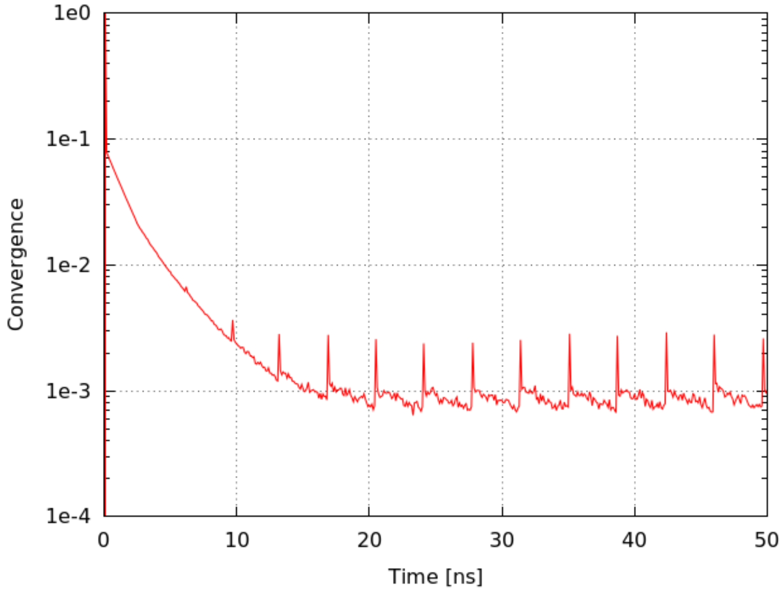

Figure 5: Convergence during time for 10^7 particles.

4.3 Discussion

Note that in Figure 1 proportion between number of electrons with initial energy and number of electrons with energy below 10 eV becomes 20 when solution converges.

Additional simulation with initial energy 500 eV was done, results shown in Figure 7. Again, the same proportion between electrons with higher and lower energies (blue and black plots).

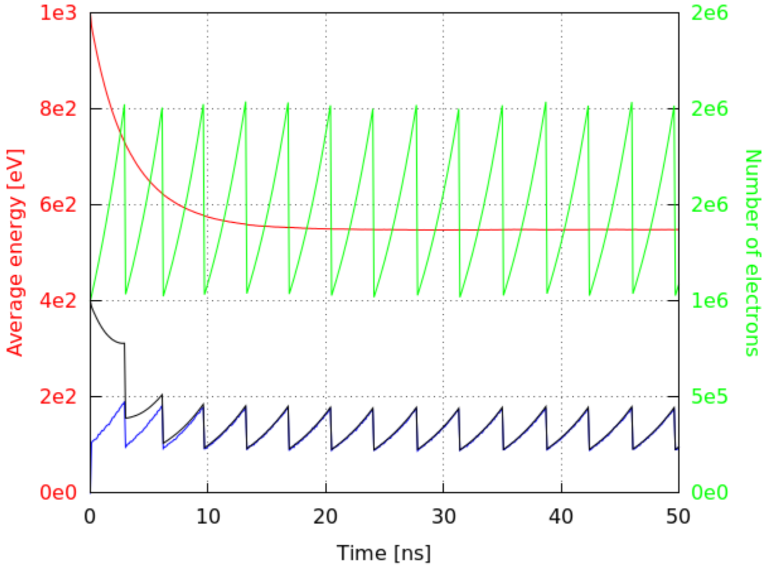

Figure 6: Average electron kinetic energy (red), total number of electrons (green), number of electrons with energy below 10 eV multiplied by 20 (blue) and number of electrons with initial energy (black).

5 Non-stationary case

5.1 Particle initialization function

Particle initialization function $g(t)$ corresponds to the total number of initialized particles at time t from the beginning of the simulation, i.e. derivative of the function $g(t)$ is the rate of particle initialization. Function $g(t)$ must be chose that its derivative is non-negative. Time in the program is not continuous; every time step t number of initialized particles in the program is

$$\Delta N_{init}^{t} = g(t) - g(t - \Delta t), \tag{5}$$

where Δt is the time sampling step.

5.2 Input data example for modified program

```
1                  | Simulation mode (0 = stationary, 1 = non-stationary)
1e4+10^20*t^2      | Particle initialization expression (e.g., 10^11*t+1e7)
                     In stationary mode skips time dependent part of expression,
                     initializes only constant (e.g., 1e7)
20e-9              | Time of simulation [s]
                     In stationary mode affects iteration number
.1e-9              | Time step [s]
```

5.3 Results

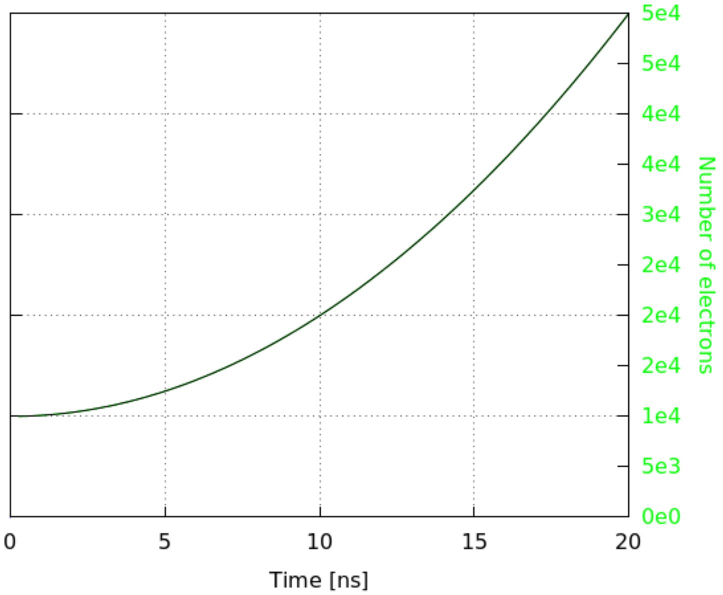

Figure 7: Total number of electrons. This graph is special case with modified program when collisions doesn't happen, it shows the particle initialization function in work. Note that this plot is the same as expression "$10^4 + 10^{20} \cdot t^2$" in the input data example.

References

[1] Savino Longo. Monte Carlo models of electron and ion transport in non-equilibrium plasmas. 2000.

[2] Tomas Lycken. Monte Carlo Simulation of Electron Beam in Nitrogen. https://github.com/tlycken/bari-montecarlo/tree/linux-build, 2013.