PARALLEL ALGORITHMS FOR STUDYING THE SYSTEM OF LONG JOSEPHSON JUNCTIONS

M. Bashashin[1,3], A. Nechaevskiy[1], D. Podgainy[1], I. Rahmonov[2], Yu. Shukrinov[2,3], O. Streltsova[1,3], E. Zemlyanaya[1,3], M. Zuev[1]

[1] *Laboratory of Information Technologies, JINR, 6 Joliot-Curie St., Dubna, 141980, Russia*

[2] *Bogoliubov Laboratory of Theoretical Physics, JINR, 6 Joliot-Curie St., Dubna, 141980, Russia*

[3] *Dubna State University, 19 Universitetskaya St., Dubna, 141980, Russia*

E-mail: zuevmax@jinr.ru

The results on studying the efficiency of parallel implementations of the computing scheme for calculating the current-voltage characteristics of the system of long Josephson junctions are presented in the paper [1-4]. The following parallel implementations were developed: the OpenMP implementation for computing on systems with shared memory, the CUDA implementation for computing on Nvidia graphics processors. The development, debugging and profiling of parallel applications were performed on the education and testing polygon of the HybriLIT heterogeneous computing platform, while computations were carried out on the "Govorun" supercomputer [2].

Keywords: long Josephson junctions, parallel computations, HPC

Maxim Bashashin, Andrey Nechaevskiy, Dmitry Podgainy, Ilhom Rahmonov, Yury Shukrinov, Oksana Streltsova, Elena Zemlyanaya, Maxim Zuev

1. Formulation of the problem

A generalized model that takes into account the inductive and capacitive coupling between long Josephson junctions (LJJs) is considered [1]. The system of *N* coupled LJJs is supposed to consist of superconducting (*S*) and intermediate dielectric (*I*) layers with a length *L* (Fig. 1).

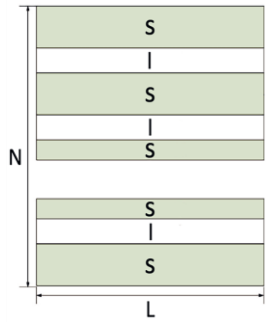

Figure 1. System of coupled long Josephson junctions

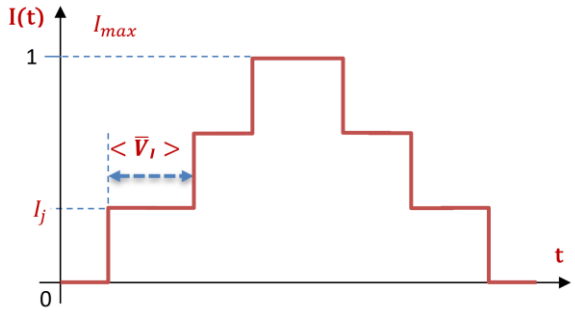

Figure 2. Schematic representation of the dependence of the current on time

Taking into account the capacitive and inductive coupling between the contacts, the phase dynamics of the system *N* LJJs is described by the initial-boundary problem for the system of differential equations relative to the difference of phases $\varphi_l(x,t)$ and voltage $V_l(x,t)$ at each *l* contact $(l=1,2,\dots N)$. In a dimensionless form, the system of equations has the form [1]:

$$\begin{cases} \dfrac{\partial \varphi}{\partial t} = C \cdot V, \\ \dfrac{\partial V}{\partial t} = \Lambda^{-1} \dfrac{\partial^2 \varphi}{\partial x^2} - \beta V + I(t); \end{cases} \quad \varphi = \begin{pmatrix} \varphi_1 \\ \varphi_2 \\ \cdots \\ \varphi_N \end{pmatrix}, \quad V = \begin{pmatrix} V_1 \\ V_2 \\ \cdots \\ V_N \end{pmatrix}, \quad 0 < x < L, \; t > 0,$$

where Λ is the inductive coupling matrix, *C* is the capacitive coupling matrix:

$$\Lambda = \begin{pmatrix} 1 & S & 0 & \cdots & \cdots & 0 & S \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & 0 & S & 1 & S & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ S & 0 & \cdots & \cdots & 0 & S & 1 \end{pmatrix}, \quad C = \begin{pmatrix} D_c & s_c & 0 & \cdots & \cdots & 0 & s_c \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & 0 & s_c & D_c & s_c & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ s_c & 0 & \cdots & \cdots & 0 & s_c & D_c \end{pmatrix},$$

β is the dissipation parameter, the inductive coupling parameter *S* takes a value in the interval $0<|S|<0.5$; D_c is the effective electric JJ thickness normalized by the thickness of the dielectric layer; s_c is the capacitive coupling parameter, $I(t)$ is the external current.

The system of equations is supplemented with zero initial and boundary conditions:

$$V_l(0,t) = V_l(L,t) = \frac{\partial \varphi_l(0,t)}{\partial x} = \frac{\partial \varphi_l(L,t)}{\partial x} = 0, \quad l = 1,2\dots N.$$

The problem when boundary conditions in the direction *x* were defined by the external magnetic field was also considered [3].

When calculating the current-voltage characteristics (CVC), the dependence of the current on time was selected in the form of steps (a schematic representation is given in Figure 2), i.e. the

problem is solved at the constant current $\left(I = I_j\right)$, while the found functions φ_l, $V_l\left(l = 1,2\ldots N\right)$ are taken as initial conditions to solve the problem for the current I_{j+1}.

2. Computing scheme

A uniform grid by the spatial variable (the number of grid nodes *NX*) was built for the numerical solution of the initial-boundary problem. In the system of equations (1), the second-order derivative by the coordinate *x* is approximated using three-point finite-difference formulas on the discrete grid with the uniform step Δx. The obtained system of differential equations relative to the values φ_l, $V_l\left(l = 1,2\ldots N\right)$ in nodes of the discrete grid by *x* is solved by the fourth-order Runge-Kutta method.

To calculate CVC, averaging $V_l\left(x,t\right)$ over the coordinate and time is performed. To do this, at each time step, the integration of voltage over the coordinate using the Simpson method and the averaging are carried out

$$\bar{V}_l = \frac{1}{L}\int_0^L V_l\left(x,t\right)dx,$$

then the voltage is averaged over time using the formula

$$\left\langle \bar{V}_l \right\rangle = \frac{1}{T_{\max} - T_{\min}} \int_{T_{\min}}^{T_{\max}} \bar{V}_l\left(t\right)dt$$

and summed by all JJs. To integrate over time, the rectangle method is used.

3. Parallel scheme

When numerically solving the initial-boundary problem by the fourth-order Runge-Kutta method over the time variable, at each time layer the Runge-Kutta coefficients $\left(K_i\right)$ can be found independently (in parallel) for all *NX* nodes of the spatial grid and for all *N* Josephson junctions. Meanwhile, the coefficients $K_i\left(i = 1,2,3,4\right)$ are defined one by one (sequentially). Thus, the parallelization is efficiently performed on the $NX \cdot N$ points. When carrying out averaging in CVC computing, the calculation of integrals can be performed in parallel as well.

4. Parallel implementations

To speed up CVC computing, parallel implementations of the computing scheme described above were developed. The results on studying the efficiency of parallel implementations performed at the values of the following parameters: $L = 10$, $I_{\min} = 0$, $I_{\max} = 1.1$, $\beta = 0.2$, $N = 1$, $T_{\max} = 200$, $\Delta t = 0.04$ – are presented below; the number of nodes by the spatial variable is $NX = 20048$.

4.1 OpenMP implementation

The computations were performed:

- on computing nodes with processors Intel Xeon Phi 7290 (KNL: 16GB, 1.50 GHz, 72 cores, 4 threads per core supported – total 288 logical cores) and the Intel compiler (Intel Cluster Studio 18.0.1 20171018);
- on dual-processor computing nodes with processors Intel Xeon E5-2695 (Broadwell; 45 MB Cache, 2.1 GHz, 18 cores, 2 threads per core supported – total 72 logical cores per node);

- on dual-processor computing nodes with processors Intel Xeon Gold 6154 (Skylake; 24.75 MB Cache, 3.00 GHz, 18 cores, 2 threads per core supported – total 72 logical cores per node);
- on dual-processor computing nodes with processors Intel Xeon Platinum 8268 (Cascade Lake; 35.75 MB Cache, 2.9 GHz, 24 cores, 2 threads per core supported – total 96 logical cores per node).

The graphs of the dependence of the calculation **speedup** obtained using the parallel algorithm:

$$S = \frac{T_1}{T_n},$$

(where T_1 is the computation time using one core, T_n is the time of computations on *n*-logical cores) on the number of threads, the number of which is equal to the number of logical cores, and the graph of the dependence of **efficiency** of using computing cores by the parallel algorithm:

$$E = \frac{T_1}{nT_n} \cdot 100\%,$$

characterizing the scalability of the parallel algorithm, are presented below.

Figure 3 shows the dependence of speedup on the number of threads when performing computations on nodes with KNL without instructions AVX-512 and using instructions AVX-512, while Figure 4 illustrates the dependence of efficiency of these computations. It is noteworthy that the use of this instruction allowed us to reduce the computation time in 1.8 times.

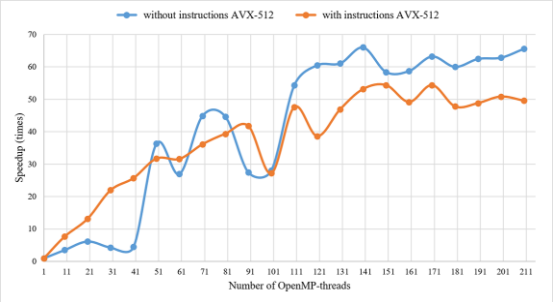

Figure 3. Graph of the dependence of speedup of parallel computing on the number of threads

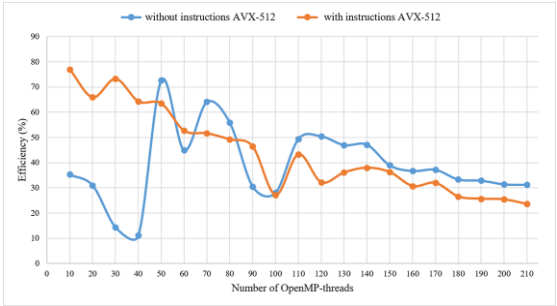

Figure 4. Graph of the dependence of efficiency of parallel computing on the number of threads

The computation time on CPU Intel Xeon E5-2695, Intel Xeon Gold 6154 and Intel Xeon Platinum 8268 is presented in Figure 5.

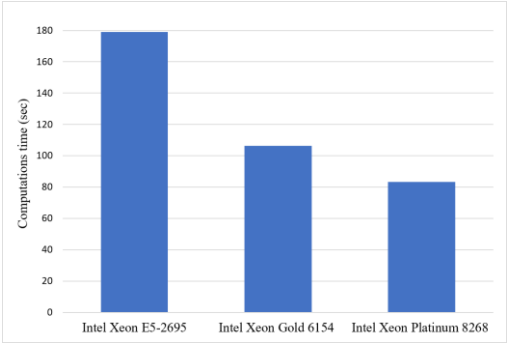

Figure 5. Computation time on CPU

4.2 CUDA implementation

A CUDA implementation of the parallel algorithm was developed for computing on Nvidia graphics accelerators. The parallel reduction algorithm using shared memory was applied for

calculating integrals. The computation time on the graphics accelerators Nvidia Tesla K40 and Nvidia Tesla K80 is presented in Figure 6.

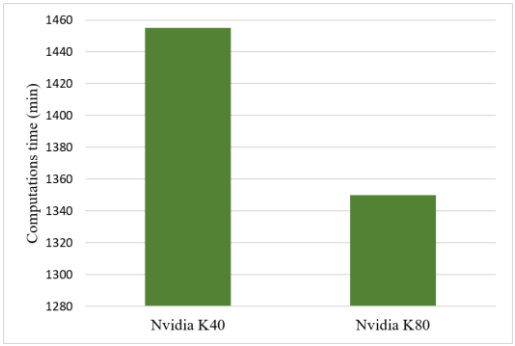

Figure 6. Computation time on GPU

5. Comparative analysis of parallel implementations

For the above parameter values to calculate CVC of LJJs, the best computation time on nodes with the KNL processor is 2108.911 minutes on 150 OpenMP threads with instructions AVX-512; using Nvidia K80 instead of Nvidia K40 reduced the computation time by 1.08 times or the above parameter values for calculating CVC of LJJs.

When comparing the processors Intel Xeon E5-2695, Intel Xeon Gold 6154 and Intel Xeon Platinum 8268, the minimal computation time is 83.23 seconds on Intel Xeon Platinum 8268, and the speedup of computing reached 2.15 times in comparison with Intel Xeon E5-2695.

Acknowledgements

The study was supported by the Russian Science Foundation (the project № 18-71-10095).

References

[1] Atanasova P.H., Bashashin M.V., Rahmonov I.R., Shukrinov Yu.M., Zemlyanaya E.V. Influence of the inductive and capacitive coupling on the current-voltage characteristic and electromagnetic radiation of the system of long Josephson junctions // Journal of Experimental and Theoretical Physics. 2017. Vol. 151. No 1. Pp. 151-159 (in Russian).

[2] Adam Gh., Bashashin M., Belyakov D., Kirakosyan M., Matveev M., Podgainy D., Sapozhnikova T., Streltsova O., Torosyan Sh., Vala M., Valova L., Vorontsov A., Zaikina T., Zemlyanaya E., Zuev M. IT-ecosystem of the HybriLIT heterogeneous platform for high-performance computing and training of IT-specialists // CEUR Workshop Proceedings. 2018. Vol. 2267. Pp.638-644.

[3] Bashashin M.V., Zemlyanaya E.V., Rahmonov I.R., Shukrinov J.M., Atanasova P.C., Volokhova A.V. Numerical approach and parallel implementation for computer simulation of stacked long Josephson Junctions // Computer Research and Modeling. 2016. Vol. 8. No. 4. Pp. 593-604.

[4] Zemlyanaya E.V., Bashashin M.V., Rahmonov I.R., Shukrinov Yu.M., Atanasova P.Kh., Volokhova A. V.Model of stacked long Josephson junctions: Parallel algorithm and numerical results in case of weak coupling // AIP Conference Proceedings. 2016. Vol 1773. 110018.