# Calculation of ground states of few-body nuclei using NVIDIA CUDA technology

## M. A. Naumenko[1,a], V. V. Samarin[1,2]

[1] Flerov Laboratory of Nuclear Reactions, Joint Institute for Nuclear Research, 6 Joliot-Curie st., Moscow reg., Dubna, 141980, Russian Federation

[2] Dubna State University, 19 Universitetskaya st., Moscow reg., Dubna, 141982, Russian Federation

E-mail: [a] anaumenko@jinr.ru

The possibility of application of modern parallel computing solutions to speed up the calculations of ground states of few-body nuclei by Feynman's continual integrals method has been investigated. These calculations may require large computational time, particularly in the case of systems with many degrees of freedom. The results of application of general-purpose computing on graphics processing units (GPGPU) using NVIDIA CUDA technology are presented. The algorithm allowing us to perform calculations directly on GPU was developed and implemented in C++ programming language. Calculations were performed on the NVIDIA Tesla K40 accelerator installed within the heterogeneous cluster of the Laboratory of Information Technologies, Joint Institute for Nuclear Research, Dubna. The energy and the square modulus of the wave function of the ground states of several few-body nuclei have been calculated. The results show that the use of GPGPU significantly increases the speed of calculations.

Keywords: NVIDIA CUDA, Feynman's continual integrals method, few-body nuclei

# 1. Introduction

In this work an attempt is made to use modern parallel computing solutions to speed up the calculations of ground states of few-body nuclei by Feynman's continual integrals method [Samarin, 2015; Samarin, Naumenko, 2016; Naumenko, Samarin, 2016]. The algorithm allowing us to perform calculations directly on GPU was developed and implemented in C++ programming language. The energy and the square modulus of the wave function of the ground states of several few-body nuclei have been calculated using NVIDIA CUDA technology [NVIDIA; Sanders J., Kandrot E., 2011]. The results show that the use of GPU is very effective for these calculations.

# 2. Theory and computing

The energy $E_0$ and the square modulus of the wave function $|\Psi_0|^2$ of the ground state of a system of few particles with coordinates $q$ may be calculated by Feynman's continual integrals method using the propagator $K_E(q,\tau;q,0)$ in Euclidian time $\tau$ [Shuryak, Zhirov, 1984]

$$K_E(q,\tau;q,0) = \sum_n |\Psi_n(q)|^2 \exp\left(-\frac{E_n\tau}{\hbar}\right) + \int_{E_{cont}}^{\infty} |\Psi_E(q)|^2 \exp\left(-\frac{E\tau}{\hbar}\right) g(E)dE . \tag{1}$$

Here $g(E)$ is the density of states with the continuous spectrum $E \geq E_{cont}$ . For the system with a discrete spectrum and finite motion of particles the square modulus of the wave function of the ground state may be found in the limit $\tau \to \infty$ together with the energy $E_0$

$$K_E(q,\tau;q,0) \to |\Psi_0(q)|^2 \exp\left(-\frac{E_0\tau}{\hbar}\right), \tau \to \infty . \tag{2}$$

The theoretical approach is described in detail in [Naumenko, Samarin, 2016]. The calculation of $K_E(q,\tau;q,0)$ for the fixed $\tau$ was performed by parallel calculation of exponentials $F$

$$F = \exp\left[-b_0\Delta\tau \sum_{k=1}^{N} V(q_k)\right] \tag{3}$$

for every random trajectory $q_k = f(q,k\Delta\tau)$, where $N = \tau/\Delta\tau$ . The same nucleon-nucleon interaction potentials $V(q_k)$ were used for all the studied nuclei.

The Monte Carlo algorithm for numerical calculations was developed and implemented in C++ programming language using NVIDIA CUDA technology. The integration method does not require the use of any additional integration libraries. The calculation included 3 steps:

1) $K_E(q,\tau;q,0)$ was calculated in a set of multidimensional points $q$ and the maximum of $K_E(q,\tau;q,0)$ (i.e. $|\Psi_0|^2$) was found.

2) The $q_0$ corresponding to the obtained maximum was fixed, $K_E(q_0,\tau;q_0,0)$ was calculated for several increasing values of $\tau$ and the linear region of $K_E(q_0,\tau;q_0,0)$ was found.

3) The time $\tau_{lin}$ corresponding to the beginning of the obtained linear region was fixed and $K_E(q_0,\tau_{lin};q_0,0)$ (i.e. $|\Psi_0|^2$) was calculated in all points of the necessary region.

The principal scheme of the calculation of the ground state energy is shown in Fig. 1. The calculation of the propagator is performed using $L$ sequential launches of the kernel. Each kernel launch simulates $n$ random trajectories in the space evolving from the Euclidean time $\tau = 0$ to $\tau_j$, where

$j = \overline{1, L}$. All trajectories with $N_j = \tau_j / \Delta\tau$ time steps start at the same point $q^{(0)}$ in the space and in the moment $\tau_j$ return to the same point $q^{(0)}$ according to the chosen probability distribution.
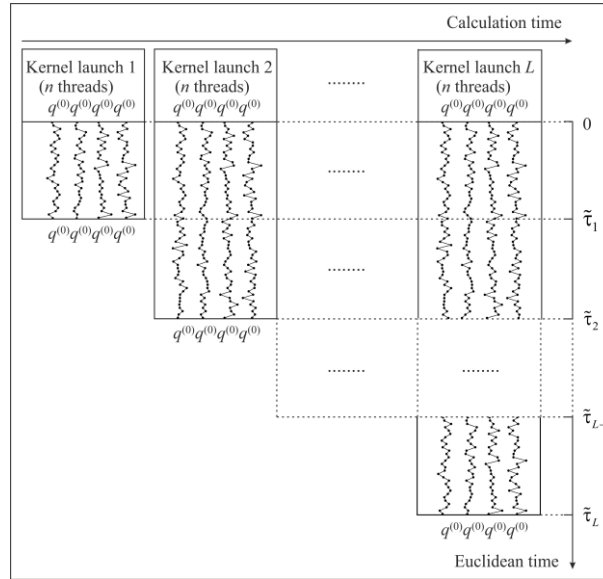


Fig. 1. The scheme of calculation of the ground state energy $E_0$

All threads in a given kernel launch finish at approximately the same time, which makes the scheme quite effective in spite of the possible delays associated with the kernel launch overhead. Besides, the typical number of kernel launches $L$ required for the calculation of the ground state energy usually does not exceed 100.

Starting from the certain time $\tau_{lin}$ the obtained values of the logarithm of the propagator $b_0^{-1} \ln K_E$ tend to lie on the straight line, the slope of which gives the value of the ground state energy. The time $\tau_{lin}$ is then used in the calculation of the square modulus of the wave function.

The principal scheme of the calculation of the square modulus of the wave function is shown in Fig. 2. Similarly, the calculation is performed using $M$ sequential launches of the kernel. Each kernel launch simulates $n$ random trajectories in the space evolving from the Euclidean time $\tau = 0$ to the time $\tau_{lin}$ determined in the calculation of the ground state energy. All trajectories start at the same point $q^{(s)}$ in the space and in the moment $\tau_{lin}$ return back to the same point $q^{(s)}$ according to the chosen probability distribution. Here $s = \overline{1, M}$, where $M$ is the total number of points in the space in which the square modulus of the wave function must be calculated.

One of the benefits of the approach is that the calculation may be easily resumed later. For example, initially the square modulus of the wave function may be calculated with a large space step to obtain the general features of the probability distribution, and later new intermediate points are calculated and combined with those calculated previously. This may be very useful because the calculation of the square modulus of the wave function is generally much more time-consuming since it requires calculation in many points in the multidimensional space.

An important feature of the algorithm allowing effective use of graphic processors is low consumption of memory during the calculation because it is not necessary to prepare a grid of values and store it in the memory.

To obtain normally distributed random numbers the cuRAND random number generator was used. According to the recommendations of the cuRAND developers, each experiment was assigned a unique seed. Within the experiment, each thread of computation was assigned a unique sequence

number. All threads between kernel launches were given the same seed, and the sequence numbers were assigned in a monotonically increasing way.
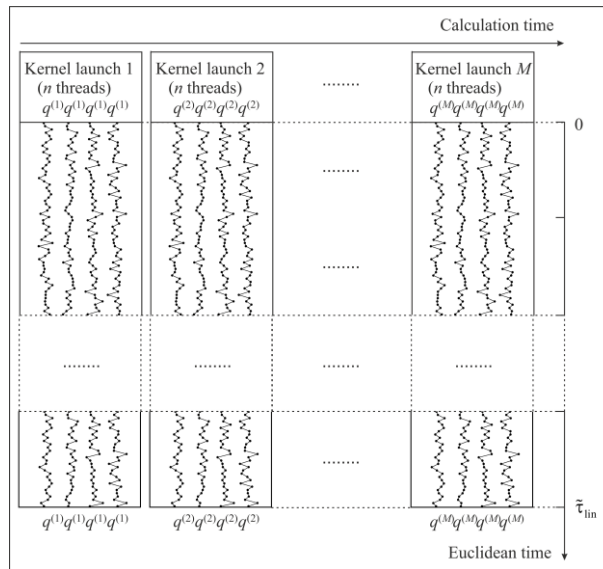


Fig. 2. The scheme of calculation of the square modulus of the wave function

Calculations were performed on the NVIDIA Tesla K40 accelerator installed within the heterogeneous cluster [Heterogeneous Cluster] of the Laboratory of Information Technologies, Joint Institute for Nuclear Research, Dubna. The code was compiled with NVIDIA CUDA version 7.5 for architecture version 3.5. Calculations were performed with single precision.

The energy of the ground state is negative and therefore only the first term in formula (1) increases with the increase of $\tau$, whereas the energies of the excited states are positive and hence the other terms in decrease with the increase of $\tau$. The slope of the linear regression equals the energy of the ground state $E_0$.

The obtained theoretical binding energies $E_b = -E_0$ are listed in Tab. 1 together with the experimental values taken from the knowledge base [Zagrebaev, Denikin, Karpov, Alekseev, Naumenko, Rachkov, Samarin, Saiko]. It is clear that the theoretical values are close enough to the experimental ones, though obtaining good agreement was not the goal. The observed difference between the calculated binding energies of $^3H$ and $^3He$ is also in agreement with the experimental values.

Tab. 1. Comparison of theoretical and experimental binding energies for the ground states of the studied nuclei

| Atomic nucleus | Theoretical value, MeV | Experimental value, MeV |
|---|---|---|
| $^2H$ | $1.17 \pm 1$ | 2.225 |
| $^3H$ | $9.29 \pm 1$ | 8.482 |
| $^3He$ | $6.86 \pm 1$ | 7.718 |
| $^4He$ | $26.95 \pm 1$ | 28.296 |

The code implementing Feynman's continual integrals method was initially written for CPU. The comparison of the calculation time of the ground state energy for $^3He$ using Intel Core i5 3470 (double precision) and NVIDIA Tesla K40 (single precision) with different statistics is shown in Tab. 2. Even taking into account that the code for CPU used only one thread, double precision and a different random number generator, the time difference is impressive. This fact allows us to increase the statistics and the accuracy of calculations in the case of using NVIDIA CUDA technology.

The comparison of the calculation time of the square modulus of the wave function for the ground state of $^3He$ using Intel Core i5 3470 and NVIDIA Tesla K40 with the statistics $10^6$ for every point in the space and the total number of points 43200 is shown in Tab. 3. The value ~ 177 days for

CPU is an estimation based on the performance gain in the calculation of the ground state energy. It is evident that beside the performance gain the use of NVIDIA CUDA technology in certain cases may enable calculations impossible before.

Tab. 2. Comparison of the calculation time of the ground state energy for $^3$He nucleus

| Statistics, $n$ | Intel Core i5 3470 (1 thread, double precision), sec | NVIDIA Tesla K40, (single precision), sec | Performance gain, times |
|---|---|---|---|
| $10^5$ | ~ 1854 | ~ 8 | ~ 232 |
| $10^6$ | ~ 18377 | ~ 47 | ~ 391 |
| $5 \cdot 10^6$ | – | ~ 221 | – |
| $10^7$ | – | ~ 439 | – |

Tab. 3. Comparison of the calculation time of the square modulus of the wave function for the ground state of $^3$He nucleus

| Statistics, $n$ | Intel Core i5 3470 (1 thread, double precision, estimation) | NVIDIA Tesla K40, (single precision) |
|---|---|---|
| $10^6$ | ~ 177 days | ~ 11 hours |

## 3. Conclusion

In this work an attempt is made to use modern parallel computing solutions to speed up the calculations of ground states of few-body nuclei by Feynman's continual integrals method. The algorithm allowing us to perform calculations directly on GPU was developed and implemented in C++ programming language. The method was applied to the nuclei consisting of nucleons, but it may also be applied to the calculation of cluster nuclei. The results show that the use of GPGPU significantly increases the speed of calculations. This allows us to increase the statistics and the accuracy of calculations as well as reduce the space step in calculations of wave functions. It also greatly simplifies the process of debugging and testing. In certain cases, the use of NVIDIA CUDA enables calculations impossible before.

## References

Heterogeneous Cluster of LIT, JINR [Electronic resource]: http://hybrilit.jinr.ru/.

*Naumenko M. A., Samarin V. V.* Application of CUDA technology to calculation of ground states of few-body nuclei by Feynman's continual integrals method // Supercomputing frontiers and innovations. — 2016. — Vol. 3, No. 2. — P. 80–95.

NVIDIA CUDA [Electronic resource]: http://developer.nvidia.com/cuda-zone/.

*Samarin V. V.* Quantum Description of Coupling to Neutron-Rearrangement Channels in Fusion Reactions near the Coulomb Barrier // Phys. At. Nucl. — 2015. — Vol. 78, No. 7 — P. 861–872.

*Samarin V. V., Naumenko M. A.* Study of Ground States of $^{3,4,6}$He Nuclides by Feynman's Continual Integrals Method // Bull. Russ. Acad. Sci. Phys. — 2016. — Vol. 80, No. 3 — P. 283–289.

*Sanders J., Kandrot E.* CUDA by Example: An Introduction to General-Purpose GPU Programming. — New York: Addison-Wesley, 2011.

*Shuryak E. V., Zhirov O. V.* Testing Monte Carlo Methods for Path Integrals in Some Quantum Mechanical Problems // Nucl. Phys. B. — 1984. — Vol. 242. — P. 393–406.

*Zagrebaev V. I., Denikin A. S., Karpov A. V., Alekseev A. P., Naumenko M. A., Rachkov V. A., Samarin V. V., Saiko V. V.* NRV web knowledge base on low-energy nuclear physics [Electronic resource]: http://nrv.jinr.ru/