

ივანე ჯავახიშვილის სახელობის თბილისის
სახელმწიფო უნივერსიტეტი

დავითი გოდერიძე

ჰიბრიდული გამოთვლითი არქიტექტურისათვის მაღალი
ენერგიებისას წარმოქმნილი ნაწილაკთა თანაკვეთის
დათვლისათვის პარალელური ალგორითმების შემუშავება

ზუსტ და საბუნებისმეტყველო მეცნიერებათა ფაკულტეტი

კომპიუტერული მეცნიერება

ნაშრომი შესრულებულია კომპიუტერული მეცნიერების მაგისტრის
აკადემიური ხარისხის მოსაპოვებლად

ხელმძღვანელები

ზურაბ მოდებაძე - ფიზიკა-მათემატიკის მეცნ. კანდიდატი, აკადემიური
დოქტორი, თსუ ასისტენტ პროფესორი

ოქსანა სტრელცოვა - ფიზიკა-მათემატიკის მეცნ. კანდიდატი, უფროსი
სამეცნიერო თანამშრომელი, გამოთვლითი ფიზიკის სამეცნიერო განყოფილება,
არაწრფივი სისტემების მეთოდების მოდელირების მე-2 სექტორი

იური კალინოვსკი - ფიზიკა-მათემატიკის მეცნიერების დოქტორი, უფროსი
სამეცნიერო თანამშრომელი, გამოთვლითი ფიზიკის განყოფილება, რთული
ფიზიკური სისტემების გამოთვლის მე-4 სექტორი

ბირთვული კლევების გაერთიანებული ინსტიტუტი, რუსეთი, დუბნა, 2018 წ.

სარჩევი

ანოტაცია	3
შესავალი	4
თავი 1. ამოცანის ჩამოყალიბება	6
თავი 2. მრავალგანზომილებიანი ინტეგრალის გამოთვლისთვის მონტე-კარლოს მეთოდის გამოყენება	9
2.1 მონტე-კარლოს მეთოდის აღწერა	9
2.2 მონტე-კარლოს მეთოდით ინტეგრირების ცდომილება	10
2.3 შემთხვევითი რიცხვების გენერატორის შერჩევა და მისი ანალიზი	12
2.4 გენერატორების აღწერა	14
2.4.1 TRNG	14
2.4.2 სტანდარტული ბიბლიოთეკა (Full Random)	16
2.4.3 ბადე (Grid)	17
2.5 გამოთვლებების გაპარალელება	17
2.5.1 Block Splitting	17
2.5.2 Leapfrog	18
3. მეთოდების გამოყენება და შედეგების ანალიზი	19
3.1 ამოცანის ალგორითმი	19
3.2 შემთხვევითი რიცხვების რაოდენობის შერჩევა	20
3.3 პროგრამის გაშვება და შედეგების ანალიზი	20
დასკვნა	26
ლიტერატურა	27

ანოტაცია

ექსპერიმენტული შედეგების ანალიზისა და პროგნოზირებისთვის საჭიროა თეორიული მოდელების შექმნა, რომლებიც აღწერენ ადრონების ქცევას მძიმე ბირთვების შეჯახებისას. თანამედროვე თეორიული გამოთვლები ხშირ შემთხვევაში საჭიროებენ სერიოზული კომპიუტერული უზრუნველყოფის შექმნას, ახალი ალგორითმების, გამოთვლების მეთოდებისა, დიდი სისტემების ინტეგრო-დიფერენციალური უტოლობების და მრავალგანზომილებიანი ინტეგრლების დონეზე. მოყვანილ ნამუშევარში მოყვანილია პიონის სიცოცხლის დროის გამოთვა, რომელიც დამოკიდებულია გარემოს ტემპერატურაზე. 5-განზომილებიანი ინტეგრალის გამოთვლისთვის იყო არჩეული მონტე-კარლოს მეთოდი. დაწერილი იყო კოდი ინტეგრალის გამოთვლისთვის, გაანალიზებული იყო დამოკიდებულება მეთოდის სიზუსტის არჩეული წერტილთა მიმართ, იყო შესწავლილი გაპარალელების მეთოდები.

For the analysis and prediction of experimental results, the creation of theoretical models describing the behavior of hadrons in the collision of heavy nuclei is required. Modern theoretical calculations often require the creation of a serious computer support at the level of creating new algorithms, calculation methods and software packages of large systems of integro-differential equations and multidimensional integrals. In this paper we calculated the lifetime of a pion depending on the ambient temperature. To calculate the 5-fold integrals, the Monte-Carlo method is chosen. The code for calculating the integral is written, the analysis of the dependence of the accuracy of the method on the number of selected points (convergence) is made, the methods of parallelization are studied.

შესავალი

ამჟამად მიდის დიდი სამუშაო მატერიის თვისებების შესწავლაზე, წარმოქმნილი მაღალი ენერგიების მძიმე იონების თანაკვეთისას. კვარკ-გლიონული პლაზმის პოვნა, ნაწილაკთა თვისებების აღწერა, კრიტიკულ პირობებში გაფანტვისა და დაშლის პროცესები რჩება თანამედროვე ფიზიკის აქტუალურ ამოცანად და შედის მძიმე იონების (NA49 , RHIC-BES, NICA [1], [2], [3], [4]) არსებული და მომავალი ექსპერიმენტების თემატიკურ გეგმებში. მიღებული ან მომავალი შედეგების წინასწარმეტყველების თეორიული აღწერა და ანალიზი ხშირად დაკავშირებულია შესახლო პროცესების მოდელირებასთან, მიმდინარე მატერიის ერთი ფაზური მდგომარეობიდან (ადრონული გაზი) მეორეში (კვარკ-გლუკონური პლაზმა) გადასვლისას. ფაზური გადასვლების მოდელირებისთვის იყენებენ ეფექტურ მოდელებს საბოლოო ტემპერატურისა და ბირთვული მატერიის სიმკვრივის დროს. ეს მოდელები უნდა გაითვალისწინონ ფიზიკური დამკვირვებადი სიდიდეების (მასა, დაშლის კონსტანტა, სიცოცხლის დრო) ქცევა ბირთვული მატერიის სიმკვრივის და ტემპერატურის ცვლილებებისას.

კომპიუტერული ტექნოლოგიების განვითარებით, მრავალი თეორია და თეორიული გამოთვლები ხდება ორიენტირებული კომპიუტერული გამოთვლის გამოყენებაზე და მოითხოვენ ახალი ალგორითმების, გამოთვლების მეთოდებისა და პროგრამული პაკეტების დიდი სისტემების ინტეგრო-დიფერენციალური უტოლობებისა და მრავალგანზომილებიანი ინტეგრალების შექმნას.

სამაგისტრო ნაშრომში დ.გოდერიძეს წინაშე იყო დასმული ამოცანა მეზონების მეზონურ გაზში სიცოცხლის დროის გამოთვლა, გამოთხვის დროის ოპტიმიზაცია (გაპარალელეობა). რიცხვითი მეთოდების თვალთახედვიდან ამოცანა დაიყვანება 5-განზომილებიანი ინტეგრალის გამოანგარიშებაზე.

დასმული ამოცანის გადასაჭრელად გადაწყდა მონტე-კარლოს მეთოდის გამოყენება [5], [6] მრავალგანზომილებიან ინტეგრალებისთვის. ამ მეთოდის არჩევის ძირითადი მიზეზი გახდა ინტეგრალის მრავალგანზომილებადობა და ინტეგრალური ფუნქციის თავისებურება (მრავალი პოლუსის არსებობა).

შემოთავაზებული და განხილული იყო შემთხვევითი რიცხვების რამდენიმე სახის გენერატორი (პსევდომშემთხვევითი რიცხვების გენერატორი, C++ ენის სტანდარტული ბიბლიოთეკის rand() გენერატორი და მისი სტატისტიკური მახასიათებლების

გაუმჯობესების მცდელობა). იყო აწონილი დადებითი და უარყოფითი მხარეები თითოეული მათგანის, ასევე იყო გაანალიზებული მიღებული შედეგები მათი გამოყენების შემდეგ რეალურ ამოცანაზე როგორც წრფივი, ასევე მრავალბირთვული (გაპარალელებული) ამოცანის რეალიზაციაში. ამისათვის იყო გამოყენებული გაპარალელების ისეთი ტექნოლოგიები, როგორცაა OpenMP და MPI [7], [8].

სადიპლომო ნაშრომი დაყოფილია რამდენიმე ნაწილად. პირველ თავში განხილულია ამოცანის თეორიული დადგმა. მეორე თავში განხილულია მონტე-კარლოს მეთოდების გამოყენება მრავალგანზომილებიან ინტეგრალებზე, შემთხვევითი რიცხვების გენერატორების შერჩევა და მათი ანალიზი. მესამე თავი მიძღვნილია არჩეული მეთოდების გამოყენებას მოყვანილი ამოცანის გადაწყვეტად და მიღებული შედეგების აღწერას.

თავი 1. ამოცანის ჩამოყალიბება

ვაკუუმში კონკრეტული ადრონული მდგომარეობის სიცოცხლის დრო განისაზღვრება მისი დაშლის სხვა ნაწილაკებზე ალბათობით. მკვრივ გარემოში ნაწილაკი შეიძლება ურთიერთქმედოს სხვა ნაწილაკებთან, რაც იწვევს მისი საწყისი მდგომარეობის ცვლილებას და კონკრეტული მდგომარეობის სიცოცხლის დრო განისაზღვრება შეჯახების სიჩქარით. არსებობს ასევე უკუპროცესებიც, რომლებსაც შეუძლიათ აღადგინონ დაშლილი მდგომარეობა გარემოს ნაწილაკთა ურთიერთქმედების ხარჯზე. ეს პროცესი ზრდის ნაწილაკის სიცოცხლის დროს. შედეგად კონკრეტული მდგომარეობის საშუალო სიცოცხლის დრო განისაზღვრება პირდაპირი და უკუპროცესების ურთიერთქმედებით.

გაზრდის პროცესების გათვლისთვის თეორიულ ფიზიკაში გამოიყენება ისეთი გაგება, როგორცაა „გაზრდის ამპლიტუდა“. ინვარიანტული გაზრდის ამპლიტუდა შეიძლება ჩაიწეროს შემდეგი სახით [2]

$$\langle cp_c; dp_d | T | ap_a; bp_b \rangle = \delta_{ab} \delta_{cd} A(s, t, u) + \delta_{ac} \delta_{bd} B(s, t, u) + \delta_{ad} \delta_{bc} C(s, t, u) \quad (1)$$

სადაც s, t, u - მანდელშტამის ინვარიანტებია და $s + t + u = 4M_\pi^2$. ასეთი სახით მას აქვს კროსინგ-სიმეტრიის მახასიათებელი, ე.ი. არ იცვლება გარე ნაწილაკთა გადაადგილებით (ან ჩანაცვლებით $s \leftrightarrow t, s \leftrightarrow u, t \leftrightarrow u$). კონკრეტული პროცესების განფანტვის გამოთვლის მოხერხებულობისათვის შეიყვანება განფანტვის ამპლიტუდები სრული იზოსპინით $I (I = 0, 1, 2)$, რომლებიც შესაბამისად ტოლია

$$A_0 = 3A + B + C \quad (2)$$

$$A_1 = B - C$$

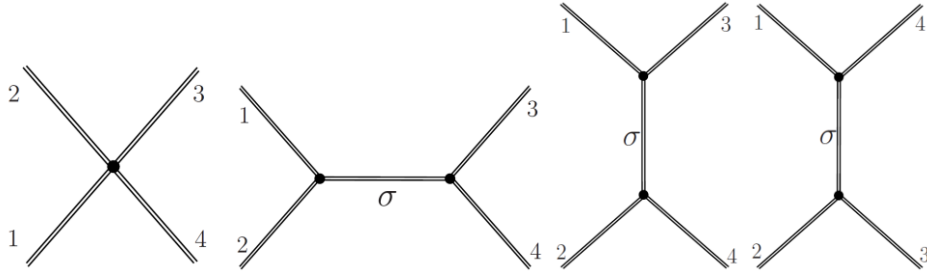
$$A_2 = B + C$$

თუმცა პრაქტიკულ ინტერესს წარმოადგენს განფანტვის სიგრძეები $a_i = \frac{1}{32\pi} A_i$, რომელთა განსაზღვრა ექსპერიმენტალურად შეიძლება.

გაზრდის ამპლიტუდის გამოანგარიშებლად აუცილებელია აირჩიოს მოდელი, რომლის ფარგლებში აიგება გაზრდის პროცესის აღმწერი ფეინმანის დიაგრამები და მოხდება ამპლიტუდების გაანალიზება. მოცემულ ნაშრომში გამოიყენება წრფივი სიგმა მოდელი ლაგრანჟიანით [3]

$$\mathbf{L} = g_{4\pi}(\vec{\phi}_\pi \vec{\phi}_\pi)^2 + g_{\sigma\pi\pi} \phi_\sigma \vec{\phi}_\pi \vec{\phi}_\pi \quad (3)$$

პიონის გაბნევის პროცესები, რომელიც აღიწერება დიაგრამებით (სურ. 1), რომელთათვისაც მიიღება შემდეგი იზოსპინური



სურ. 1: $\pi\pi$ -გაბნევის წრფივი სიგმა მოდელის დიაგრამა.

ამპლიტუდების მნიშვნელობები

$$A_0 = 5C_{4\pi} + 4g_{\sigma\pi\pi}^2 \left(\frac{3}{M_\sigma^2 - s} + \frac{1}{M_\sigma^2 - t} + \frac{1}{M_\sigma^2 - u} \right), \quad (4)$$

$$A_1 = 4g_{\sigma\pi\pi}^2 \left(\frac{1}{M_\sigma^2 - t} - \frac{1}{M_\sigma^2 - u} \right), \quad (5)$$

$$A_2 = 2C_{4\pi} + 4g_{\sigma\pi\pi}^2 \left(\frac{1}{M_\sigma^2 - t} + \frac{1}{M_\sigma^2 - u} \right), \quad (6)$$

აქ $C_{4\pi} = 2g_{4\pi}$ არის 4 პიონის ურთიერთქმედების კონსტანტა. წრფივი სიგმა მოდელის ფარგლებში: $m_\sigma^2 = 0,66 \text{ GeV}^2$, $g_{\sigma\pi\pi} = 2.7 \text{ GeV}$, $C_{4\pi\pi} = -34.81$

როგორც ნაჩვენებია (1)-ზე, იმის ალბათობა, რომ ნაწილაკის მოთავსებისას იმპულსით \mathbf{p} გარემოში დროის t მომენტში და ნაწილაკთა დაშორებით იგივე იმპულსით დროის t მომენტში, შესაძლებელია ვიპოვოთ გაზი იგივე მდგომარეობაში, რაც დასაწყისში, რომლის დაშლა ხდება კანონით $e^{-\Gamma(t-t)}$, სადაც Γ – კონსტანტაა, რომელიც აღწერს დაშლის სიჩქარეს (ხშირად „საგანეს“ ემახიან). თუ ამპლიტუდად ავირჩევთ სპექტრალურ ფუნქციას ბრეიტ-ვიგნერის ფორმით:

$$A(s) = \frac{M\Gamma}{(s-M^2)^2 + M^2\Gamma^2} \quad (7)$$

სადაც M – მდგომარეობის მასა, მაშინ დაშლის ალბათობა განისაზღვრება ფორმულით $e^{-\Gamma(t-t)}$, სადაც Γ [1].

ძირითად შემთხვევაში, სიგანე წარმოადგენს ენერჯისა და იმპულსის ფუნქციას, და საბოლოო შედეგად ვიღებთ სიგანისთვის განტოლებათა სისტემას, რომელიც ცუდად მიეწოდება რიცხვით გამოთვლას. რეალისტიკური აღწერისთვის ვივარაუდოთ, რომ სიგანე დამოკიდებულია ტემპერატურისა და სიმკვრივეზე. T პიონის სიცოცხლის დრო განისაზღვრება ფორმულით [1]

$$\Gamma(\mathbf{p}) = \tau^{-1}(\mathbf{p}) = \Sigma^>(\mathbf{p}) - \Sigma^<(\mathbf{p}), \quad (8)$$

$$\Gamma = \frac{1}{2p_0}(\Sigma^>(\mathbf{p}_0) - \Sigma^<(\mathbf{p}_0)) \quad (9)$$

სადაც

$$\Sigma^<(\mathbf{p}) = \frac{1}{(2\pi)^3} \int ds_1 A(s_1) ds_3 A(s_3) ds_4 A(s_4) \int \frac{d^3\vec{p}_1}{(2\pi)^3 2E_1} (\mathbf{1} + \mathbf{n}_1) \quad (10)$$

$$\frac{1}{16\pi} \int_{-1}^1 d(\cos\alpha) \frac{|\vec{p}_3^*|^2}{|\vec{p}_3^*|^2 (\sqrt{s_2} + m_1) - |\vec{p}_1| E_3^* \cos\alpha} |A|^2 n_3 n_4$$

$$\Sigma^>(\mathbf{p}) = \frac{1}{(2\pi)^3} \int ds_1 A(s_1) ds_3 A(s_3) ds_4 A(s_4) \int \frac{d^3\vec{p}_1}{(2\pi)^3 2E_1} \mathbf{n}_1 \quad (11)$$

$$\frac{1}{16\pi} \int_{-1}^1 d(\cos\alpha) \frac{|\vec{p}_3^*|^2}{|\vec{p}_3^*|^2 (\sqrt{s_2} + m_1) - |\vec{p}_1| E_3^* \cos\alpha} |A|^2 (\mathbf{1} + \mathbf{n}_3)(\mathbf{1} + \mathbf{n}_4)$$

სადაც $E_1 = \sqrt{\vec{p}_1^2 + s_1}$, $E_3^* = \sqrt{(\vec{p}_3^*)^2 + s_3}$, $E_4^* = \sqrt{s_2} + E_1 - E_3^*$ - პიონების ენერჯია,

$\mathbf{n}_i = \frac{\mathbf{E}_i}{E_i} - \frac{1}{e^{\tau} - 1}$ - პიონის განაწილების ფუნქცია, $p_0 = m_\pi = 0.14$ გეგ-პიონის მასა.

$$\int \frac{d\vec{p}_1}{(2\pi)^3 2E_1} = \frac{1}{4\pi^2} \int \frac{|\vec{p}_1|^2 d|\vec{p}_1|}{E_1},$$

$$|\vec{p}_3^*|_{1,2} = \frac{|\vec{p}_1| \cos\alpha \pm \sqrt{|\vec{p}_1|^2 \cos^2\alpha a^2 + ((\sqrt{s_2} + E_1)^2 - |\vec{p}_1|^2 \cos^2\alpha)(a^2 - 4s_3(\sqrt{s_2} + E_1)^2)}}{2((\sqrt{s_2} + E_1)^2 - |\vec{p}_1|^2 \cos^2\alpha)},$$

$$\alpha = (\sqrt{s_2} + s_1)^2 + s_3 - s_4 + s_1 - E_1^2.$$

თავი 2. მრავალგანზომილებიანი ინტეგრალის გამოთვლისთვის მონტე-კარლოს მეთოდის გამოყენება

2.1 მონტე-კარლოს მეთოდის აღწერა

არსებობს ინტეგრალების გამოთვლის უამრავი რიცხვითი მეთოდები, მაგრამ მოყვანილი ამოცანის სპეციფიკიდან გამომდინარე იყო არჩეული მონტე-კარლოს მეთოდი [5], [6].

როგორ მუშაობს მონტე-კარლოს მეთოდი და რატომ იგი იყო არჩეული?

მონტე-კარლოს მეთოდების საერთო განმარტება დღეს-დღეობით არ არსებობს, მაგრამ შეიძლება ითქვას, რომ მონტე-კარლოს მეთოდები ეს არის მათემატიკური ამოცანების შემთხვევითი რიცხვების მოდელირებისას გამოთვლის რიცხვითი მეთოდები (პროცესი მოდელირდება შემთხვევითი რიცხვების გენერატორის საშუალებით. მიღებული შემთხვევითი მონაცემების საფუძველზე გამოითვლება პრობლემის ალბათობის მახასიათებლები).

დავუშვათ, საჭიროა რომელიმე ფუნქციის ინტეგრალის გამოთვლა. გეომეტრიული წერტილის თვალთახედვიდან ამოცანა წარმოადგენს მოცემული ფუნქციის გრაფიკისთვის ფართობის გამოთვლას. კლასიკური ინტეგრირების უმარტივესი მეთოდების გამოყენებისას (მაგ, მართკუთხედების მეთოდი), საჭიროა სეგმენტის თანაბარი დანაწილება, ყოველი ქვესეგმენტისთვის ფუნქციის მნიშვნელობის გამოთვლა და შემდგომ ამ პასუხების აჯამვა. ორგანზომილებიან სივრცეში ამოცანის გამოთვლა ძალიან მარტივად და სწრაფად გამოიყურება. მაგალითად, საერთო სეგმენტის 100 ტოლ ქვესეგმენდად დანაწილებისას საჭიროა მარტო 100 ლოკალური პასუხის გამოთვლა და მათი აჯამვა საერთო პასუხის გასაგებად (ინტეგრალის პასუხი). N-განზომილებიან სივრცეში კი თუ ასევე დავყოფთ ყოველ სეგმენტს 100 ტოლ ნაწილად, მაშინ საჭიროა 100^N გამოთვლის შესრულება. მრავალგანზომილებიან ამოცანებში გამოთვლების რაოდენობა იზრდება კატასტროფულად სწრაფად.

მონტე-კარლოს მეთოდი კი თავის მხვრივ თავისუფალია ამგვარი საჭირო გამოთვლების ზრდისგან. საჭიროა მხოლოდ საკმაოდ კარგი შემთხვევითი/პსევდოშემთხვევითი გენერატორის შეჩვენა, რომელიც მოგვცემს რიცხვების თანაბარ გადანაწილებას ფუნქციის განსაზღვრის არეზე. პასუხის სიზუსტე ამ შემთხვევაში დამოკიდებულია მხოლოდ გენერატორზე და გამოთვლებში მონაწილე შემთხვევითი

რიცხვის რაოდენობაზე (თუ ფუნქცია საკმაოდ გლუვია).

ასევე მონტე-კარლოს მეთოდს იყენებენ იმ შემთხვევებში, როცა ცნობილია ინტეგრალური ანალიტიკური წარმოდგენა, მაგრამ მისი პირველადი ვერ აისახება ანალიტიკური ფუნქციით, ან პირველადის სახე იმდენად რთულია, რომ უფრო სწრაფი იქნება ინტეგრალის მნიშვნელობის გამოთვლა რიცხვითი მეთოდებით. ზემოთთქმულს ასევე ემატება ის ფაქტი, რომ მონტე-კარლოს მეთოდი საშუალებას გვაძლევს შემოვიაროთ ის კრიტიკული წერტილები, რომლებშიც შეიძლება მივიღოთ „უცნაური“ შედეგები (მაგ. უსასრულობა, ნული და ა.შ.). ეს შემოვლა მსგავს წერტილებში კომპენსირდება დანარჩენი ვარიანტების ჯამის განახევრებით.

განსაზღვრული ერთგანზომილებიანი ინტეგრალის მონტე-კარლოს მეთოდით გამოთვლის ფორმულას შემდეგი სახე აქვს [5]:

$$\int_a^b f(x) dx \approx \frac{(b-a)}{N} \sum_{i=1}^N f(x_i), \quad (12)$$

სადაც x_i შემთხვევითი რიცხვია მონაკვეთიდან $[a; b]$ (განსაზღვრული ინტეგრალის ზღვარები), $f(x_i)$ – ფუნქციის მნიშვნელობა მოცემულ წერტილში x_i , N – გამოთვლებში მონაწილე შემთხვევითი რიცხვების რაოდენობა.

მრავალგანზომილებიანი ინტეგრალისათვის ფორმულა (12) შემდეგნაირად გამოიყურება:

$$\int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} f(x_1, \dots, x_n) dx_1 \dots dx_n = \frac{(b_1 - a_1) \dots (b_n - a_n)}{N} \sum_{i=1}^N \bar{f}(x_1, \dots, x_n) \quad (13)$$

მონტე-კარლოს მეთოდი გამოყენებადია მხოლოდ იმ შემთხვევებში, როდესაც ინტეგრალის ზღვრები არ არიან დამოკიდებულები ერთმანეთზე. ინტეგრალის (10), (11) თავდაპირველი ფორმა არ აკმაყოფილებდა ამ პირობას, იყო ჩატარებული განუსაზღვრელობა.

2.2 მონტე-კარლოს მეთოდით ინტეგრირების ცდომილება

უზუსტოების (ინტეგრირების ცდომილების) გასაგებად იყენებენ შემდეგ ფორმულას:

$$\tilde{I} = V \cdot \tilde{F}_{cp} \cdot \tilde{\Omega} \quad (14)$$

სადაც V – მოცულობა, ინტეგრირების არის შემოსაზღვრა;

$\tilde{\Omega}$ - ინტეგრირების ახალი არის მოცულობა Ω , მიღებული ცვლადების ცვლილების შემდეგ (ინტეგრალის განუსაზღვრელობის შემდეგ);

\tilde{F}_{cp} – ინტეგრალური ფუნქციის საშუალო მნიშვნელობის შეფასება, მიღებული განუსაზღვრელობის შემდეგ.

ცდომილებათა თეორიიდან ცნობილია ორი სიდიდის ნამრავლის ცდომილების ფორმულა:

$$\Delta I = \Delta(V \cdot \tilde{F}_{cp} \cdot \tilde{\Omega}) = V \cdot (\Delta F_{cp} \cdot \tilde{\Omega} + \Delta \Omega \cdot |\tilde{F}_{cp}|). \quad (15)$$

ცენტრალური ზღრვული თეორემით:

$$P \left\{ |F_{cp} - \tilde{F}_{cp}| \leq t_{\beta} \sqrt{\frac{\sigma_1^2}{n}} \right\} \approx \beta;$$

$$P \left\{ |\Omega - \tilde{\Omega}| \leq t_{\beta} \sqrt{\frac{\sigma_1^2}{N}} \right\} \approx \beta;$$

ე.ი. ალბათობით β

$$\Delta F_{cp} = t_{\beta} \sqrt{\frac{\sigma_1^2}{n}}, \Delta \Omega = t_{\beta} \sqrt{\frac{\sigma_1^2}{N}};$$

ამასთანავე

$$\tilde{F}_{cp} = \frac{1}{n} \sum_{i=1}^n F(M_i), \tilde{\Omega} = \frac{n}{N}.$$

ჩავსვათ ცდომილების მნიშვნელობები ფორმულაში (14)

$$\Delta I = V \cdot \left(t_{\beta} \sqrt{\frac{\sigma_1^2}{n}} \cdot \tilde{\Omega} + t_{\beta} \sqrt{\frac{\sigma_1^2}{N}} \cdot |\tilde{F}_{cp}| \right). \quad (16)$$

უმნიშვნელო გარდაქმნებით ცნობილი σ_1 და σ_2 , ცდომილების შეფასებისათვის მივიღებთ:

$$\Delta I = V \cdot t_{\beta} \left(\sigma_1 \frac{\tilde{\Omega}}{\sqrt{n}} + \sigma_2 \frac{|\tilde{F}_{cp}|}{\sqrt{N}} \right).$$

შემთხვევებში, როცა σ_1^2 და σ_2^2 უცნობია – მათი მიახლოებული გამოითვლება ფორმულებით:

$$S_1^2 = (\tilde{F}^2)_{cp} - (|\tilde{F}_{cp}|)^2, \quad (17)$$

$$S_2^2 = \tilde{N} \cdot (1 - \tilde{N})$$

სადაც

$$\tilde{F}_{cp} = \frac{1}{n} \sum_{i=1}^n F(M_i) \quad (18)$$

$$(\tilde{F}^2)_{cp} = \frac{1}{n} \sum_{i=1}^n F^2(M_i)$$

$$\tilde{N} = \frac{n}{N} \quad (19)$$

მაშინ ცდომილების შეფასებისთვის ვიღებთ ფორმულას:

$$\Delta I = V \cdot t_\beta \left(S_1 \cdot \frac{\tilde{N}}{n} + S_2 \cdot \frac{|\tilde{F}_{cp}|}{\sqrt{N}} \right) \quad (20)$$

N – შემთხვევით რიცხვთა საერთო რაოდენობა

n – გამოთვლებში გამოყენებული წერტილთა რაოდენობა

t_β – პარამეტრი, β ნდობის კოეფიციენტის შესაბამისი (შეფასებაში უდრის 3)

2.3 შემთხვევითი რიცხვების გენერატორის შერჩევა და მისი ანალიზი

ინტეგრალის გამოთვლისთვის მონტე-კარლოს მეთოდის გამოყენებისას საჭიროა შესასვლელი შემთხვევითი რიცხვების უსასრულო ნაკადი ფუნქციის ინტეგრირების არიდან. ამისათვის საჭიროა რიცხვითი ფუნქციის არსებობა, რომელიც ყოველი იტერაციის დროს დააგენერირებს დამოუკიდებელ წერტილს, რომელშიც მოხდება ინტეგრალის მნიშვნელობის გამოთვლა. ამ ამოცანის შესასრულებლად შეიძლება გამოყენებული იქნას ფსევდოშემთხვევითი რიცხვების მზა გენერატორი, სტანდარტული ბიბლიოთეკა C++ ენაში, ან და ახლიდან შექმნილი შემთხვევითი რიცხვების გენერატორი.

მთავარ ფაქტორად, რომელიც გავლენას ახდენს ინტეგრალის გამოთვლის სიზუსტეზე, არის არჩეული შემთხვევითი რიცხვების გენერატორი. ნაკლებად ზეგავლენას ახდენს გამოთვლებში მონაწილე შემთხვევითი რიცხვების რაოდენობა (N -ჯერ შემთხვევითი რიცხვების ზრდა ამცირებს გამოთვლის ცდომილებას $\frac{1}{\sqrt{N}}$ - ზე). ეს წესი

მოქმედებს მარტო გლუვ ფუნქციებზე პოლუსების არ არსებობის შემთხვევაში, როდესაც ფუნქციის მნიშვნელობა არ ისწრაფვის უსასრულო რეზულტატისკენ.

მოყვანილი ამოცანის გადასაჭრელად იყო არჩეული შემთხვევითი რიცხვების გენერაციის რამდენიმე ვარიანტი:

1. ფსევდოშემთხვევითი რიცხვების გენერატორის გამოყენება („შემთხვევითი“ რიცხვების მიღების რიგი წინასწარ ცნობილია, მაგრამ სამაგიეროდ ვიღებთ კარგად გადანაწილებას განსაზღვრების არეზე, რითაც თავისუფლდებით სრულიად შემთხვევითი რიცხვების მიღებისაგან, რითაც შეგვიძლია მივიღოთ არათანაბარი გადანაწილება განსაძღვრების არიდან. ეს ფაქტორი შეუძლებელს გახდის რეალურ პასუხამდე მიახლოებას.

2. სრულიად შემთხვევითი რიცხვების მიღება. სტანდარტული ბიბლიოთეკა C++-ში გვაძლევს ამის საშუალებას (`srand(time(NULL))`); ბრძანების შესრულებით). იტერაციის ყოველ ბიჯზე მიიღება სრულიად შემთხვევითი რიცხვი (მონტე-კარლოს მეთოდის ძირითადი იდეა რიცხვთა შერჩევისას ამაში მდგომარეობს), მაგრამ თანაბარი გადანაწილების შეფასება ამ შემთხვევაში შეუძლებელია. შესაბამისად, შესაძლებელი შედეგის ვარაუდიც შეუძლებელია.

3. ჰიბრიდი მონტე-კარლოს მეთოდის, მარტივი რიცხვითი მეთოდების და C++ სტანდარტული ბიბლიოთეკის სრულიად შემთხვევითი რიცხვების გენერაციისათვის სეგმენტებზე. ამ ვარიანტში რეალიზებადია შემდეგი ორი მიდგომა:

1. სეგმენტის დაყოფა ტოლ ქვესეგმენტებად და 1 შემთხვევითი რიცხვის არჩევა თითო სეგმენტიდან (საჭიროა დაყოფის საკმაოდ დიდი რაოდენობა წერტილთა დიდი რაოდენობის მისაღებად, რაც ძალიან ბევრი რაოდენობის დაყოფისას დაიყვანება ინტეგრალის გამოთვლაზე მართკუთხედების მეთოდზე).

2. სეგმენტის დაყოფა მცირე რაოდენობის ქვესეგმენტად (მაგ, მონაკვეთი $[0; 1]$ დაყოფით 10 ტოლ მონაკვეთად, რის შემდეგ ყოველ მონაკვეთში გენერირდება $N/10$ (N – წერტილთა საერთო რაოდენობა განსაზღვრების არიდან მოცემულ სიბრტყეზე) შემთხვევითი წერტილი. ამ ვარიანტში იზრდება მიღებული მნიშვნელობის შემთხვევითობა, მაგრამ თანაბარი გადანაწილების შემთხვევითობა შეიძლება დაზიანდეს).

2.4 გენერატორების აღწერა

2.4.1 TRNG

TRNG (Tina random number generator) [7], [8] არის პსევდოშემთხვევითი რიცხვების გენერატორი. მისაღები რიცხვები წინასწარგანსაზღვრულია, რაც გამორიცხავს სრულიად შემთხვევითი რიცხვების მიღებას. საკმაოდ დიდი ნაბიჯის მქონე ალგორითმები დაგვიბრუნებს სხვადასვა მონაცემებს დუბლირების გარეშე და რომლებიც თანაბრად იქნებიან გადანაწილებულები ფუნქციის განზღვრის არეზე.

ბიბლიოთეკა TRNG საკმაოდ დიდია, მაგრამ ჩვენ გამოვიყენებთ პსევდოშემთხვევითი რიცხვების გენერატორს YARN (Yet Another Random Number), ასევე გადანაწილებას Uniform01.

გენერატორი YARN დაფუძნებულია მრავალ რეკურსიულ გენერატორზე n ცალ მოპირდაპირე ნიშანზე, რისთვისაც გადასვლის ალგორითმი კითხულებს

$$r_i = a_i r_{i-1} + a_2 r_{i-2} + \dots + a_n r_{i-n} \bmod m \quad (18)$$

გენერატორის მდგომარეობა i -ურ წერტილში მოცემულია მნიშვნელობებით $(r_{i-1}, r_{i-2}, \dots, r_{i-n})$.

YARN გენერატორის მახასიათებლები წარმოდგენელია შემდეგ ცხრილებში:

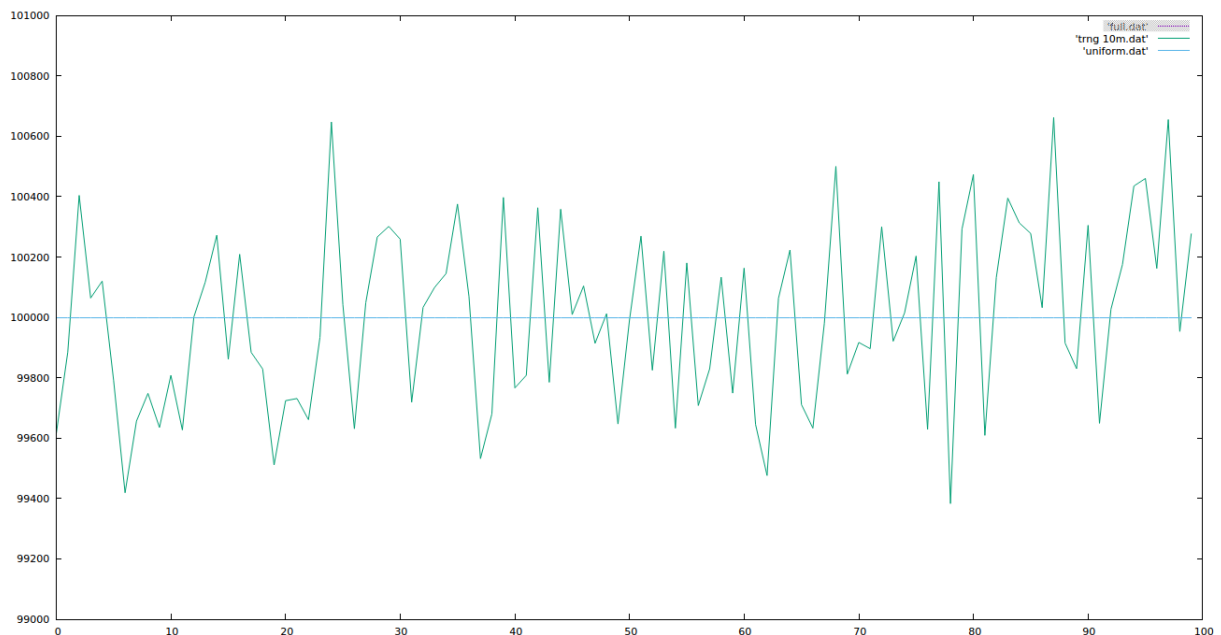
class	header file	feedback taps n	prime field \mathbb{F}_m	period	return value of name()
trng::yarn2	trng/yarn2.hpp	2	$\mathbb{F}_{2^{31}-1}$	$m^2 - 1 \approx 2^{62} \approx 4.61 \cdot 10^{18}$	yarn2
trng::yarn3	trng/yarn3.hpp	3	$\mathbb{F}_{2^{31}-1}$	$m^3 - 1 \approx 2^{93} \approx 9.90 \cdot 10^{27}$	yarn3
trng::yarn3s	trng/yarn3s.hpp	3	$\mathbb{F}_{2^{31}-21069}$	$m^3 - 1 \approx 2^{93} \approx 9.90 \cdot 10^{27}$	yarn4s
trng::yarn4	trng/yarn4.hpp	4	$\mathbb{F}_{2^{31}-1}$	$m^4 - 1 \approx 2^{124} \approx 2.13 \cdot 10^{37}$	yarn4
trng::yarn5	trng/yarn5.hpp	5	$\mathbb{F}_{2^{31}-1}$	$m^5 - 1 \approx 2^{155} \approx 4.57 \cdot 10^{46}$	yarn5
trng::yarn5s	trng/yarn5s.hpp	5	$\mathbb{F}_{2^{31}-22641}$	$m^5 - 1 \approx 2^{155} \approx 4.57 \cdot 10^{46}$	yarn5s

ცხრილი 1. YARN კლასის გენერატორების ფუნქციების გასაღებები(18)

parameter set	a_1	a_2	a_3	a_4	a_5	b
trng::yarn2::LEcuyer1	1 498 809 829	1 160 990 996				123 567 893
trng::yarn2::LEcuyer2	46 325	1 084 587				123 567 893
trng::yarn3::LEcuyer1	2 021 422 057	1 826 992 351	1 977 753 457			123 567 893
trng::yarn3::LEcuyer2	1 476 728 729	0	1 155 643 113			123 567 893
trng::yarn3::LEcuyer3	65 338	0	64 636			123 567 893
trng::yarn3s::trng0	2 025 213 985	1 112 953 677	2 038 969 601			1 616 076 847
trng::yarn3s::trng1	1 287 767 370	1 045 931 779	58 150 106			1 616 076 847
trng::yarn4::LEcuyer1	2 001 982 722	1 412 284 257	1 155 380 217	1 668 339 922		123 567 893
trng::yarn4::LEcuyer2	64 886	0	0	64 322		123 567 893
trng::yarn5::LEcuyer1	107 374 182	0	0	0 104 480	123 567 893	
trng::yarn5s::trng0	1 053 223 373	1 530 818 118	1 612 122 482	133 497 989	573 245 311	889 744 251
trng::yarn5s::trng1	2 068 619 238	2 138 332 912	671 754 166	1 442 240 992	1 526 958 817	889 744 251

ცხრილი 2. YARN გენერატორების პარამეტრები(18)

სურ. 2-ზე ასახულია გადანაწილების სიმჭიდროვე 10 მილიონი შემთხვევითი რიცხვის გათამაშებისას [0;1] მონაკვეთზე. სეგმენტი დაყოფილია 100 ტოლ მონაკვეთად. როგორც ჩანს, გადანაწილება საკმაოდ თანაბარია და ყოველ მონაკვეთზე მოსალოდნელ 100 000 მნიშვნელობიდან გადახრა წარმოადგენს სულ მხოლოდ ± 600 შემთხვევითი რიცხვს.

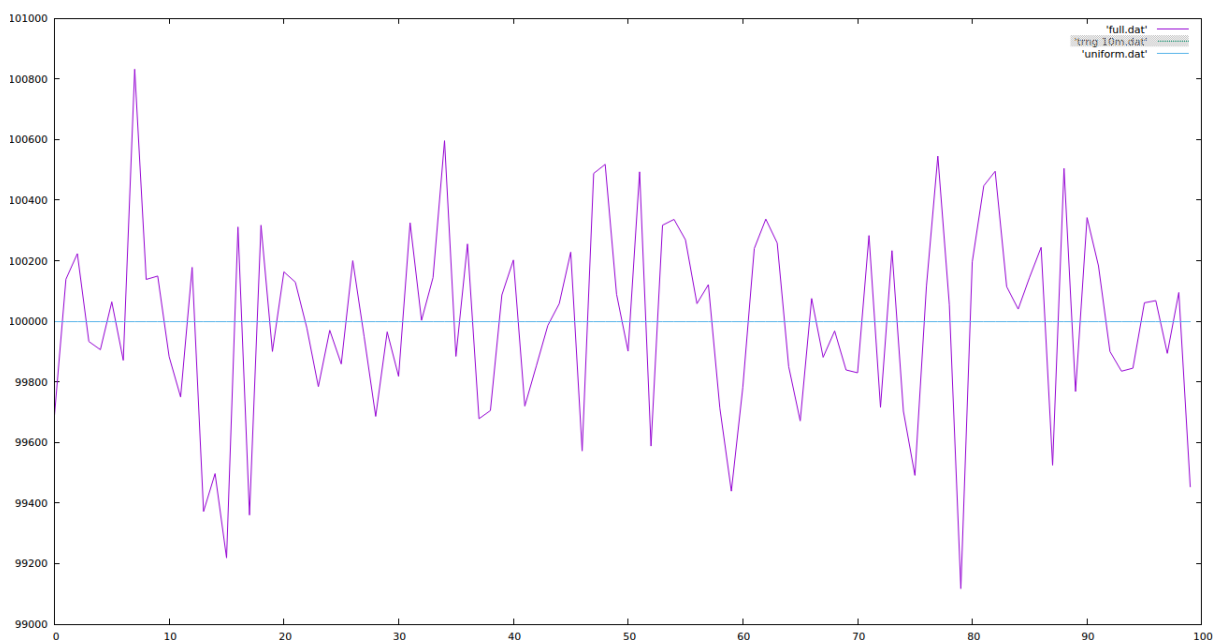


სურ 2. 10 მილიონი შემთხვევითი რიცხვის გადანაწილება TRNG (YARN + Uniform01) ბიბლიოთეკის გამოყენებისას მონაკვეთზე [0;1], რომელიც, თავის მხვრივ დაყვავით 100 ტოლ მონაკვეთად. (აბსოლუტურად თანაბარი გადანაწილებისას ყოველ მონაკვეთში უნდა ყოფილიყო 100 000 მნიშვნელობა).

2.4.2 სტანდარტული ბიბლიოთეკა (Full Random)

ეს ალგორითმი ანალოგიურია როგორც TRNG-ს მაგალითში, მაგრამ შემთხვევითი რიცხვების შერჩევისას გამოიყენება C++ ენის სტანდარტული ბიბლიოთეკა rand(). ამ შემთხვევაში შენარჩუნებულია მონტე-კარლოს მეთოდის ძირითადი იდეა შემთხვევითობის მხრიდან, მაგრამ შეუძლებელია ვაკონტროლოთ შემთხვევითი რიცხვების თანაბარი გადანაწილება ინტეგრირების არეზე და ასევე რჩება სრულიად შემთხვევითი.

მაგალითად, C++ ენის სტანდარტული ბიბლიოთეკის rand()-ის გამოყენებისას ჩვენ ვიღებთ შემდეგ სურათს:



სურ 3. C++ სტანდარტული ბიბლიოთეკის (rand()) გამოყენებისას 10 მილიონი შემთხვევითი რიცხვის გადანაწილება [0;1] მონაკვეთზე, რომელიც, თავის მხრივ დავყავით 100 ტოლ მონაკვეთად. (აბსოლუტურად თანაბარი გადანაწილებისას ყოველ მონაკვეთში უნდა ყოფილიყო 100 000 მნიშვნელობა).

როგორც ჩანს სურ 3.-დან, ზოგიერთ მონაკვეთზე ცდომილება აღემატება 800 მნიშვნელობას და ეს გადახრა არამუდმივია. შესაბამისად, შეგვიძლია დავასკვნათ, რომ TRNG პსევდოშემთხვევითი რიცხვების გენერატორის გამოყენება იგებს მნიშვნელობების

გადანაწილების სიმჭიდროვეში, მაგრამ ამ შემთხვევაში იკარგება თვით „შემთხვევითი“ რიცხვის გაგება, რადგან მნიშვნელობათა ნაკადი რჩება უცვლელი.

ამასთანავე TRNG ბიბლიოთეკაში წარმოდგენილია ფუნქციონალი გაპარალელელებულ პროგრამებში გამოყენებისთვის. ორი მეთოდი (Block Splitting და Leap Frog) საშუალებას გვაძლევს გადავანაწილოდ ფსევდოშემთხვევით რიცხვთა სტრიქონი ნაკადების შორის შიშის გარეშე, რომ გენერატორით დაბრუნებული მნიშვნელობა იქნება დუბლირებული.

2.4.3 ბადე (Grid)

მარტივი რიცხვითი მეთოდების გამოყენება მრავალგანზომილებიან ინტეგრალებზე დაკავშირებულია გამოთხვებისთვის საჭირო მნიშვნელობათა სწრაფ ზრდასთან. თუმცა, გარკვეული მანიპულაციების საშუალებით შეგვიძლია შევქმნათ ჰიბრიდული მეთოდი.

2.4.3.1 ბადე შემთხვევითი რიცხვებით (Grid Random)

Grid Random – ეს არის შემთხვევითი რიცხვების სტანდარტული ბიბლიოთეკით გენერაციის სტატისტიკული შედეგების გაუმჯობესების მცდელობა. ამ მეთოდის ლოგიკა შემდეგია:

1. დავყოთ განსაზღვრების არე n ტოლ მონაკვეთად
2. ყოველი ქვემონაკვეთიდან ავირჩიოთ m ცალი შემთხვევითი რიცხვი
3. გამოვიყენოთ ყოველი მიღებული რიცხვი გამოთვლებში

ამ სახის მიდგომა გარანტიას გვაძლევს შემთხვევითი რიცხვების უფრო კარგ გადანაწილებაზე (რადგან ყოველ სეგმენტში გვექნება თანაბარი რაოდენობის რიცხვი, რაც, თავის მხრივ, აისახება გადანაწილების მთლიან სურათზე).

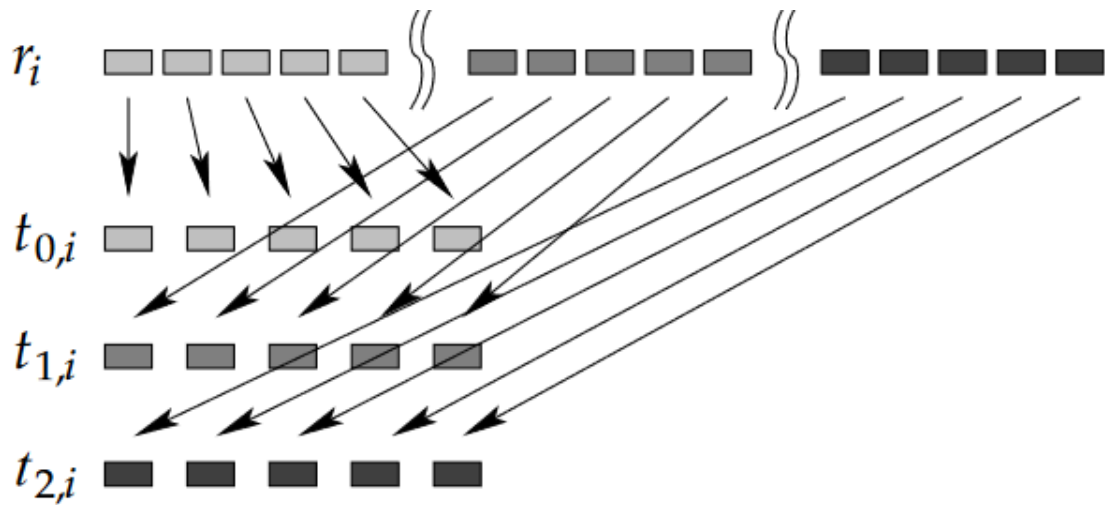
2.5 გამოთვლების გაპარალელება

გამოთვლათა დაჩქარების მიზნით, იყო დაწერილი პროგრამის ვერსია, რომელიც იყენებს გაპარალელების ტექნოლოგიებს.

2.5.1 Block Splitting

N ცალი შემთხვევითი რიცხვის გათამაშებისას p ცალ პროცესორზე (რიცხვთა ნუმერაცია $[0 - N-1]$, პროცესორთა ნუმერაცია $[0 - p-1]$, იქმნება N/p ადგილის მქონე p ცალი დამოუკიდებელი ბლოკი და ამ ბლოკთა შორის ნაწილდება ყველა გათამაშებადი წერტილი გადანაწილების მიმდევრობიდან. ახლა ყოველი პროცესორი ფლობს თავისი

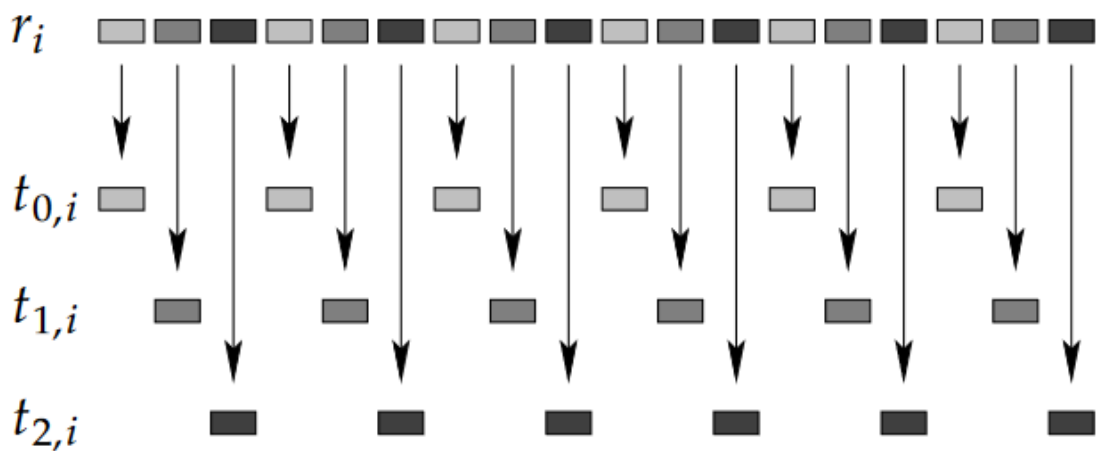
დამოუკიდებელი ფსევდომემთხვევითი რიცხვების ნაკადით (ბლოკით) და შეუძლია სხვა პროცესორების პარალელურად მოახდინოს გამოთვლები.



სურ 4. Block-Splitting მეთოდით ფსევდომემთხვევითი რიცხვების ნაკადის გადანაწილება პროცესორების შორის

2.5.2 Leapfrog

ეს არის მარტივი მიდგომა p -რაოდენობის დამოუკიდებელი ფსევდომემთხვევითი რიცხვების ნაკადების მიღებისათვის. წინა ვარიანტისგან განსხვავებით მშობელი ნაკადი არ ნაწილდება დამოუკიდებელ ბლოკებად. ამ მეთოდის გამოყენებისას ყოველი პროცესორი იღებს რიცხვს საერთო ნაკადიდან ისე, რომ ამოღებული რიცხვი ვერ მოხვდება სხვა პროცესორში.



სურ 5. Leapfrog მეთოდით ფსევდომემთხვევითი რიცხვების ნაკადის გადანაწილება პროცესორების შორის

3. მეთოდების გამოყენება და შედეგების ანალიზი

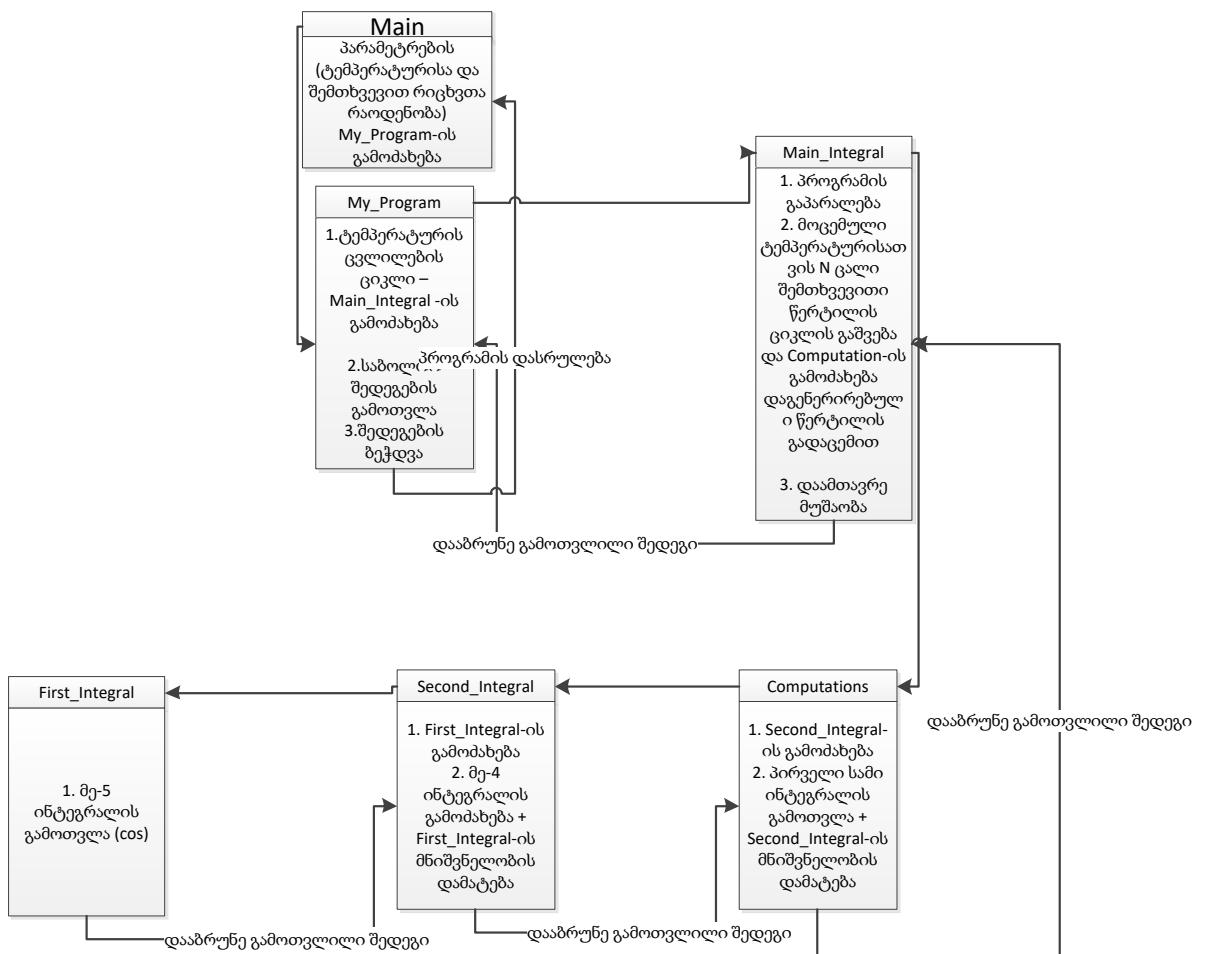
ყველა წარმოდგენილი მეთოდი განსხვავდება მხოლოდ შემთხვევითი რიცხვების მიღების გზით და მათი გადანაწილებით პროცესორთა შორის. თვით ინტეგრალების გამოთვლის ფუნქციები უცვლელი რჩება.

3.1 ამოცანის ალგორითმი

1. ამოცანის ჩამოყალიბება:

საჭიროა პიონის სიგანის გამოთვლა ტემპერატურის ცვლილების დროს, ვიპოვოთ ამოცანის ამონახსნი თეორიული ვარაუდებიდან გამომდინარე, და შედეგი წარმოვიდგინოთ გრაფიკის სახით.

2. მეთოდის განსაზღვრენა და გამოთვლისთვის შემთხვევით წერტილთა რაოდენობის შერჩევა.
3. ინტეგრალების (10), (11) გამოთვლის ალგორითმის სქემა



სურ. 6. ნაჩვენებია პროგრამის ალგორითმის სქემა. გვაქვს მარეგულირებელი ფუნქცია (Main_Integral), რომელიშიც დაწერილია გაპარალელების კოდი და შემთხვევითი

რიცხვების გენერატორი. 5-განზომილებიანი ინტეგრალის გამოთვლა დაყოფილია 3 ნაწილად (1-2-3 ინტეგრალები საჭიროებენ მსგავსი დამატებითი გამოთვლების განხორციელებას). ყოველი შემთხვევითი წერტილი ხდება სამივე ნაწილის გამოთვლის ფუნქციაში (ჯერ Computation-ფუნქციაში, აქედან Second_Integral-ში და ბოლოს First_Integral-ში. მონაცემების დაბრუნება ხდება იგივე გზით. და ყოველ ფუნქციის გავლისას მათი შედეგები ერთიანდება. ბოლოს საერთო შედეგი Main_Integral-ში ბრუნდება სადაც იგი ინახება სხვა წერტილებში გამოთხვლილი შედეგების ერთად.

4.გამოთვლების შემდეგ მიღებული შედეგების ანალიზი.

3.2 შემთხვევითი რიცხვების რაოდენობის შერჩევა

ამოცანის გადაჭრისთვის საჭირო შემთხვევით რიცხვთა რაოდენობის შერჩევისთვის იყო ჩატარებული სტატისტიკური ანალიზები TRNG ბიბლიოთეკის გამოყენებით. ყოველ ექსპერიმენტში ტემპერატურა იყო ფიქსირებული. იყო ჩატარებული მონტე-კარლოს მეთოდით ინტეგრირების ცდომილების გამოთვლა პარამეტრებით: $m_p=0.14$; $T=0.19$;

- N წერტილთა საერთო რაოდენობა
- n „კარგი“ წერტილების რაოდენობა
- I_Gr $\Sigma^>$ ინტეგრალის მნიშვნელობა
- I_Les $\Sigma^<$ ინტეგრალის მნიშვნელობა
- ΔGr $\Sigma^>$ ინტეგრალის ცდომილება
- ΔLes $\Sigma^<$ ინტეგრალის ცდომილება
- Γ პიონის სიცოცხლის დრო

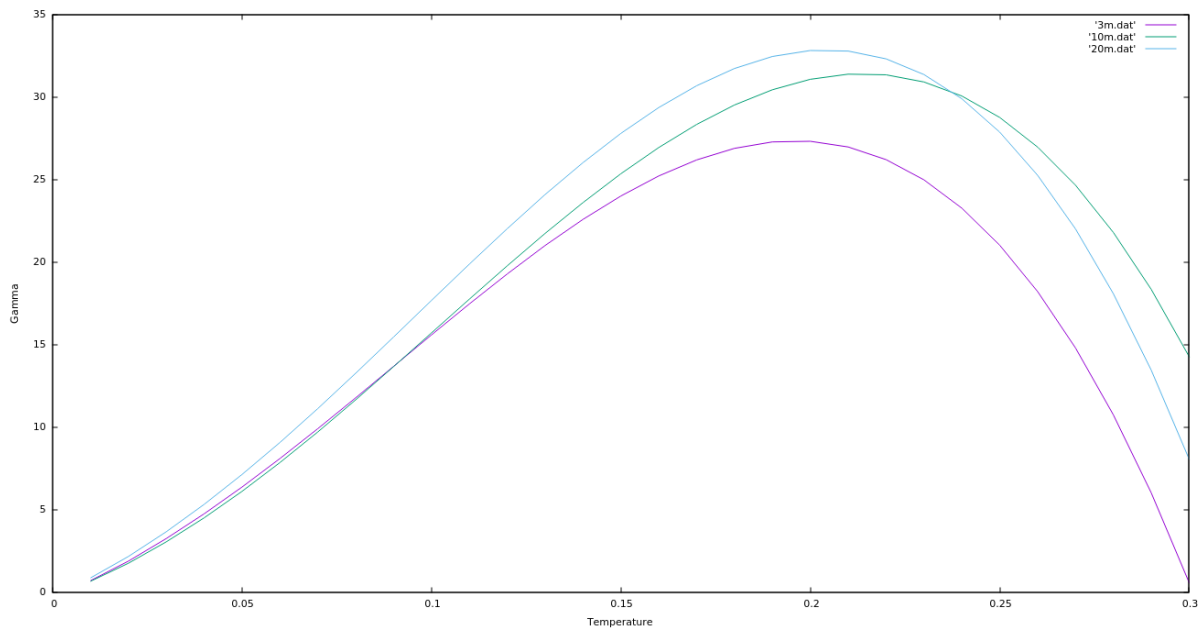
N	n	I_Gr	I_Les	ΔGr	ΔLes	Γ
$3 \cdot 10^6$	74 470	24.7215	17.0849	0.00737965	0.0104805	27.2735
10^7	248 038	25.4871	16.9658	0.00402328	0.0110849	30.4333
$2 \cdot 10^7$	496 318	28.3509	19.2653	0.00326851	0.00514189	32.4486

ცხრილი 3. ინტეგრირების ცდომილების ანალიზი

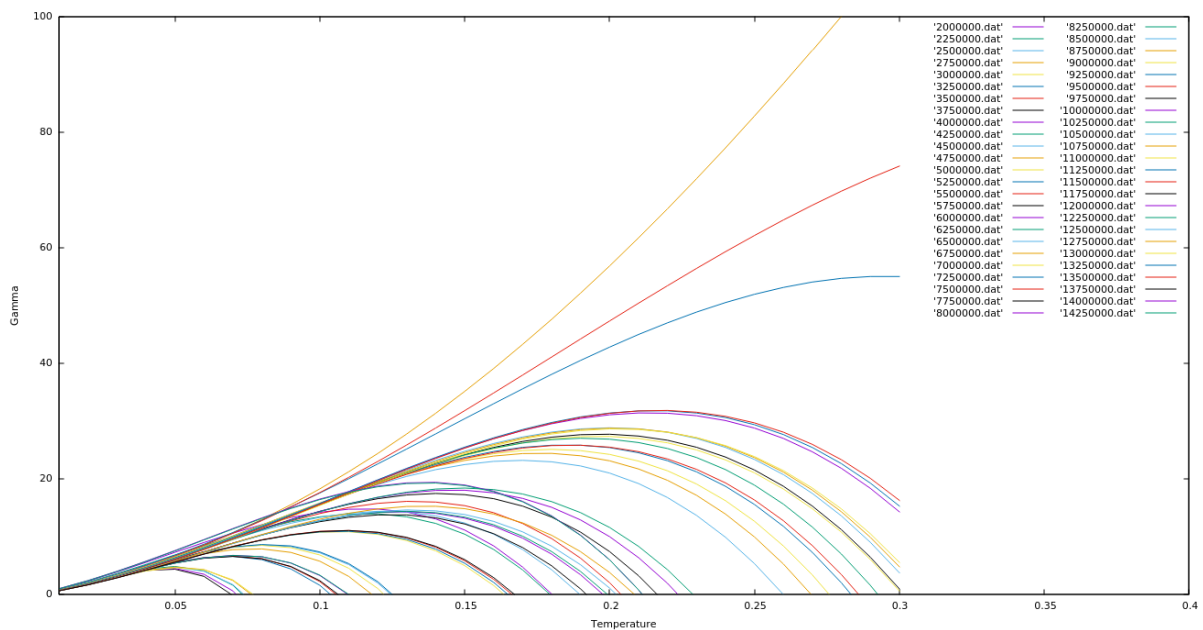
3.3 პროგრამის გაშვება და შედეგების ანალიზი

*ყველა გამოთვლა იყო ჩატარებული HybriLIT ჰეტეროგენურ პლატფორმაზე [11]

ცხრილი 3.-ში მოყვანილია შემთხვევით რიცხვთა რაოდენობის 3 ვარიანტი, რომელთა გათამაშებისას ინტეგრალები (10), (11) სტაბილურობას ინარჩუნებდნენ. (სტაბილურობა ნარჩუნდებოდა მოყვანილი წერტილთა რაოდენობებიდან $\pm 100\,000 - 200\,000$ წერტილების რაოდენობის შეცვლით, უფრო დიდი ნაბიჯით გადასვლა ამ წერტილებიდან იწვევდა ინტეგრალის რიცხვითი პასუხების განსხვავებას (უყ. სურ 7). ზუსტად ამ რაოდენობის გათამაშებით ოყო აშენებული პიონის სიცოცხლის ტემპერატურაზე დამოკიდებულების ცხრილები.

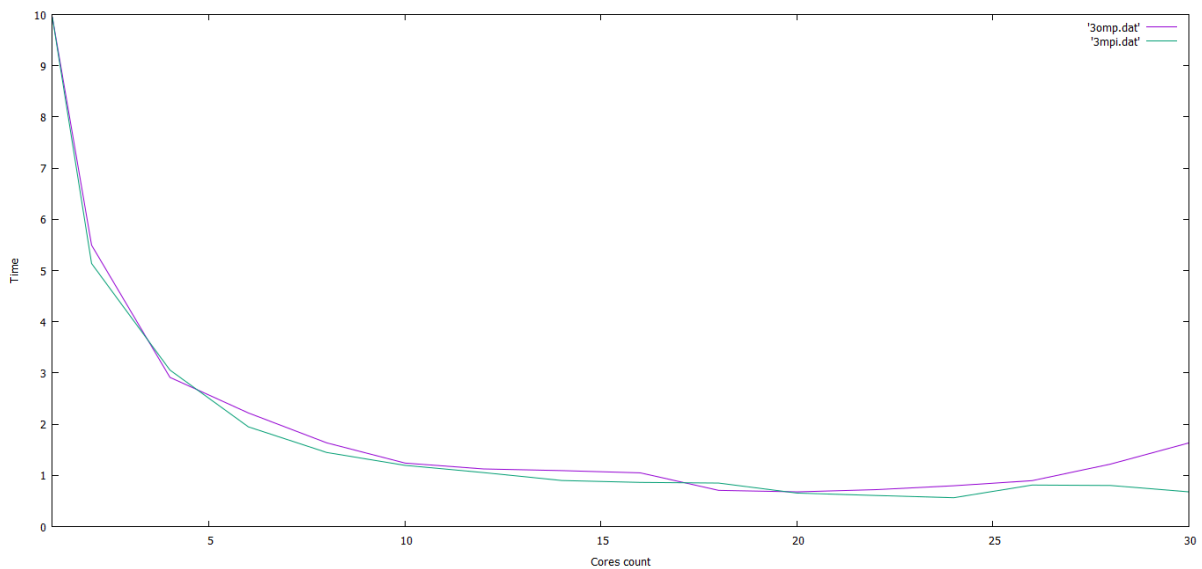


სურ 7. პიონის სიგანის მნიშვნელობის ცვლილება ტემპერატურის ცვლილების შედეგად. გამათვლავში მონაწილე წერტილთა საერთო რაოდენობა: 3, 10 და 20 მილიონი წერტილი.

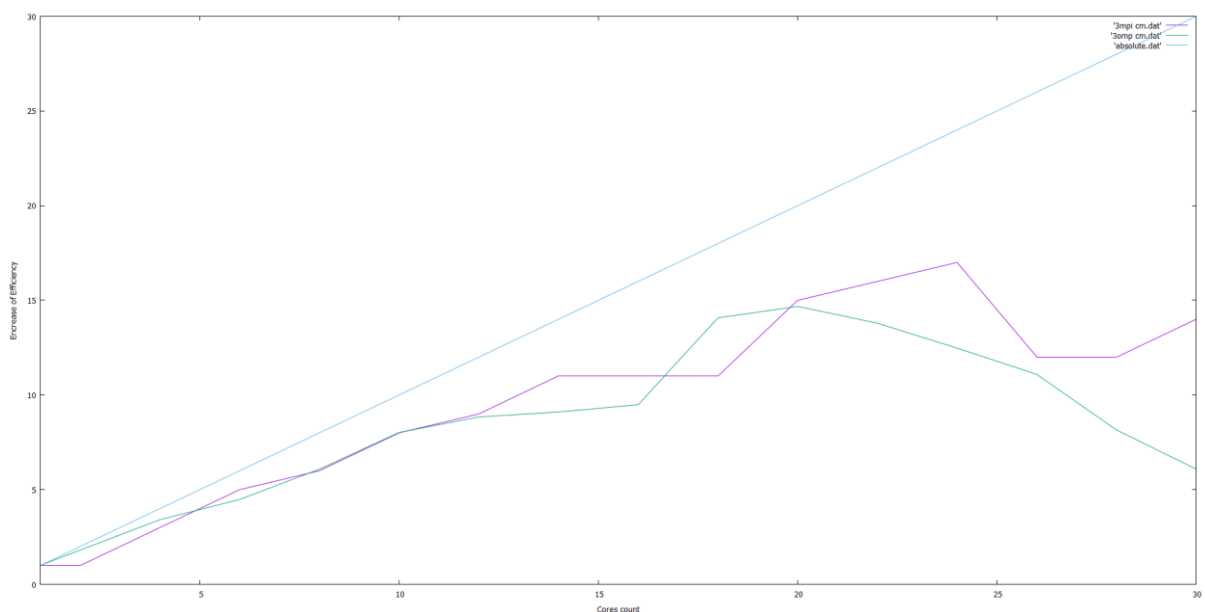


სურ. 8. პიონის სიგანის მნიშვნელობის ცვლილება ტემპერატურის ცვლილების შედეგად. გამოთვლებში მონაწილეობდნენ წერტილების რაოდენობა დაწყებული 2 მილიონიდან და დამთავრებული 14.25 მილიონით, ნაბიჯით 250 ათასი წერტილი.

ყველა ასახულ ცდებში იყო ჩატარებული დიდი რაოდენობი გამოთვლა, რის შედეგად დადგა საკითხი გამოთვლების დაჩქარებაზე გაპარალელების მეთოდების გამოყენების გზით (უყ. პ. 2.5). სურათებზე 9-14 ასახულია გამოთვლების დრო სხვადასხვა მეთოდებისა და სხვადასხვა ბირთვთა რაოდენობის გამოყენებისას.

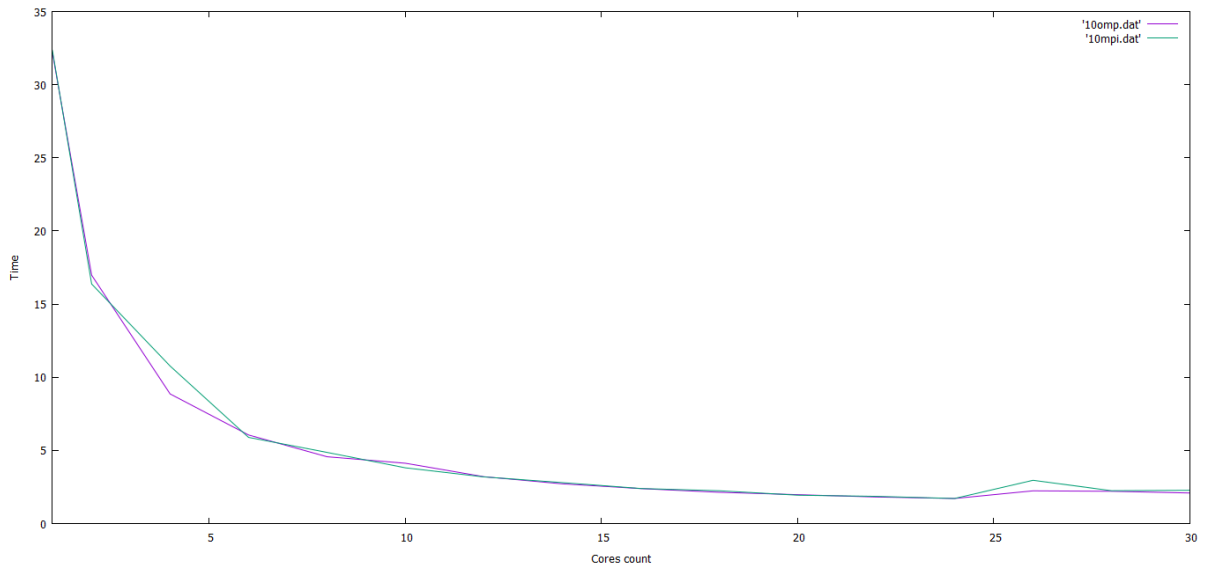


სურ 9. გამოთვლების დროის ცვლილება ბირთვთა რიცხვის ზრდის შედეგად. OpenMP და MPI ტექნოლოგია. 3 მილიონი წერტილი.

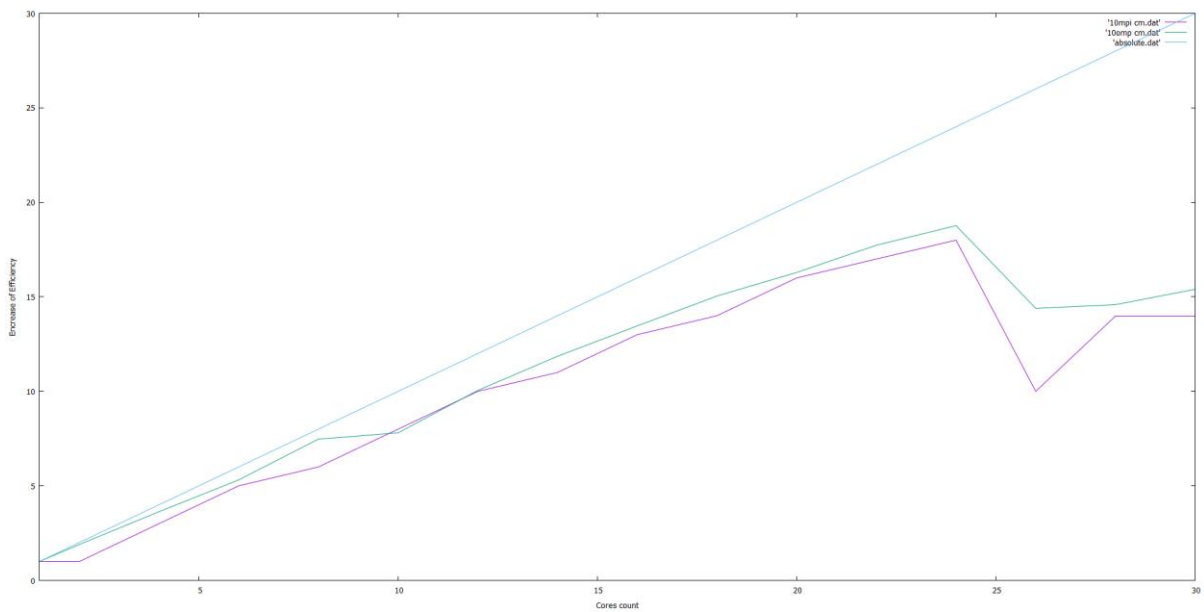


სურ. 10 გამოთვლების ეფექტურობის ცვლილება ბირთვების რაოდენობის ზრდის შემთხვევაში.

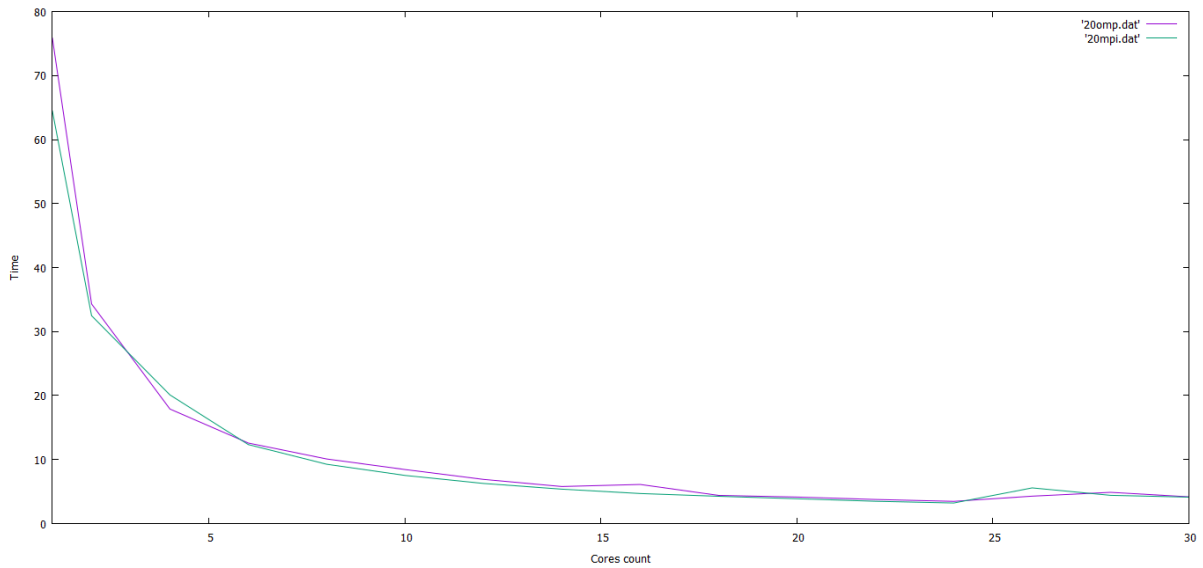
OpenMP და MPI ტექნოლოგია. 3 მილიონი წერტილი.



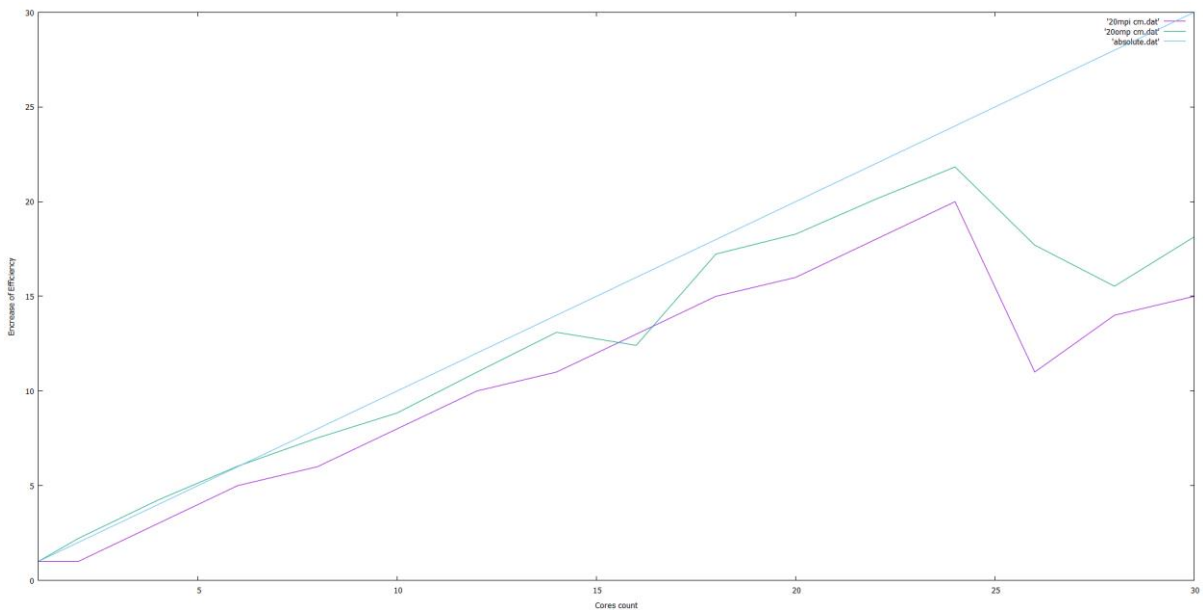
სურ 11. გამოთვლების დროის ცვლილება ბირთვთა რიცხვის ზრდის შედეგად. OpenMP და MPI ტექნოლოგია. 10 მილიონი წერტილი.



სურ. 12 გამოთვლების ეფექტურობის ცვლილება ბირთვების რაოდენობის ზრდის შემთხვევაში. OpenMP და MPI ტექნოლოგია. 10 მილიონი წერტილი.



სურ. 13. გამოთვლების დროის ცვლილება ბირთვთა რიცხვის ზრდის შედეგად. OpenMP და MPI ტექნოლოგია. 20 მილიონი წერტილი.



სურ. 14 გამოთვლების ეფექტურობის ცვლილება ბირთვების რაოდენობის ზრდის შემთხვევაში. OpenMP და MPI ტექნოლოგია. 20 მილიონი წერტილი.

ზემოთ წარმოდგენილ სურათებიდან არ არის რთული იმის შენიშვნა, რომ ბირთვთა რაოდენობის ზრდას, რომლებიც მონაწილეობენ გამოთვლებში, აზრი აქვს მხოლოდ გარკვეულ ფარგლებში, რომელიც ძირითადად დამოკიდებული საჭირო

გამოთვლების რაოდენობებაზე. შემდგომი მომატება იწვევს გამოაგარიშების სიჩქარის შენელებას.

შემთხვევითი რიცხვების მიღების სტანდარტული ბიბლიოთეკა აჩვენა თავისი სრული უვარგისობა მოცემული ამოცანის გადასაჭრელად, რადგან უფრო ცუდი გადანაწილება დიდი ზეგავლენა მოახდინა მიღებულ შედეგზე (10), (11) ინტეგრალებში პოლუსების არსებობის გამო.

დასკვნა

იყო აგებული დამოკიდებულება პიონის სიგანის გარემოს ტემპერატურის მიმართ. იყო ნაჩვენები, რომ ტემპერატურის ზრდასთან ეს მნიშვნელობა აღწევს მაქსიმუმს, როცა $T \approx 0.2$, რიც შემდეგ იწყებს შემცირებას (ფენომენს ჰქვია - dumping). 5-განზომილებიანი ინტეგრალების (10), (11) გამოთვლისათვის იყო არჩეული მონტე-კარლოს მეთოდი. იყო შესწავლილი მეთოდის თეორიული საფუძვლები და დაწერილი პროგრამა მისი პრაქტიკული რეალიზაციისთვის. გამოთვლის პროცესის დაჩქარების მიზნით იყო დაწერილი პროგრამის ვერსია, რომელიც იყენებს გაპარალელების ტექნოლოგიებს (OpenMP, MPI). იყო დადგენილი ბირთვთა ოპტიმალური რაოდენობა, რომელთა რაოდენობის გამოყენებისას შედეგს ვიღებთ უმოკლეს დროში.

მიღებული შედეგების ანალიზისას იყო დადგენილი, რომ ინტეგრალები (10), (11) ფლობენ განსაკუთრებლობას მნიშვნელში და დამოკიდებულები არიან შესასვლელ მონაცემთა რაოდენობაზე. შემთხვევით რიცხვთა გენერატორი რიცხვთა უფრო თანაბრი გადანაწილებით გამოყენებადია დასმული ამოცანის გადაჭრაში, რაც შეუძლებელია ითქვას C++ ენის შემთხვევით რიცხვთა სტანდარტულ ბიბლიოთეკაზე.

ლიტერატურა

- [1] S.V. Afanasiev et.al. [NA49 collaboration] Phys. Rev. C66, 054902 (2002)), RHIC-BES (STAR BES White paper, SN0598), NICA (D. Blaschke et.al, Eur. Phys. J.A (2016) 52: 267)).
- [2] L .P. Kadanoff and G. Baym, Quantum Statistical Mechanics, (W. A. Benjamin, Inc., New York, 1962).
- [3] J. D. Bjorken and S. Drell Relativistic Quantum Mechanics (1964, New York: McGraw-Hill).
- [4] S. R. Colanich, P. Maris, Phys. Rev. D 66, 116010 (2002).
- [5] И. М. Соболев Численные Методы Монте-Карло (1973, Москва)
- [6] Ермаков С.М. Метод Монте-Карло и смежные вопросы. – М. Наука, 1975г.
- [7] Антонов А.С. Технологии параллельного программирования MPI и OpenMP: Учебное пособие для вузов. МГУ им.М.В.Ломоносова; Суперкомпьютерный консорциум университетов России. - М.: Издательство Московского университета, 2012. - 344с.
- [8] Гергель В.П. Современные языки и технологии параллельного программирования. Суперкомпьютерный консорциум университетов России. - М.: Издательство Московского университета, 2012. - 408с.
- [9] Heiko Bauke Tina's Random Number Generator Library Version 4.20, January 29, 2018
- [10] Walter E. Brown, Mark Fischler, Jim Kowalkowski, Marc Paterno, Random Number Generation in C++0X: A Comprehensive Proposal, version 2 , CEPA Dept., Computing Division Fermi National Accelerator Laboratory Batavia, IL 60510-0500 U.S.A.
- [11] <http://hlit.jinr.ru/>