

Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Механико-математический факультет

УТВЕРЖДАЮ:  
Декан ММФ ТГУ  
Л. В. Гензе

Рабочая программа дисциплины

**Введение в методы параллельных вычислений**

по направлению подготовки

**01.03.01 Математика**

**02.03.01 Математика и компьютерные науки**

Направленность (профиль) подготовки:

**Основы научно-исследовательской деятельности в области математики  
Основы научно-исследовательской деятельности в области математики и  
компьютерных наук**

Форма обучения

**Очная**

Квалификация

**Бакалавр**

Год приема

**2023**

СОГЛАСОВАНО:  
Руководитель ОП  
Л.В. Гензе

Председатель УМК  
Е.А. Тарасов

Томск – 2023

## **1. Цель и планируемые результаты освоения дисциплины**

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-4 Способен проводить под научным руководством исследование на основе существующих методов в конкретной области профессиональной деятельности.

ПК-1 Способен проводить научно-исследовательские разработки по отдельным разделам выбранной темы.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК 4.1 Проводит поиск и обработку научной и научно-технической информации, необходимой для решения исследовательских задач

ИОПК 4.2 Оценивает полученные результаты и формулирует выводы по итогам проведенных исследований

ИПК 1.1 Проводит работы по обработке и анализу научно-технической информации и результатов исследований

ИПК 1.2 Подготавливает планы и программы проведения отдельных этапов научно-исследовательской работы

ИПК 1.3 Проводит отдельные этапы научно-исследовательской работы

## **2. Задачи освоения дисциплины**

- Освоить аппарат параллельной реализации задач вычислительной математики различного уровня сложности - от вычисления частных сумм или рекуррентных соотношений до решения уравнений в частных производных методом конечных разностей.
- Научиться проведению теоретических оценок эффективности известных или вновь созданных параллельных алгоритмов.

## **3. Место дисциплины в структуре образовательной программы**

Дисциплина относится к Блоку 1 «Дисциплина (модули)».

Дисциплина относится к части образовательной программы, формируемой участниками образовательных отношений, предлагается обучающимся на выбор.

## **4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине**

Восьмой семестр, зачет

## **5. Входные требования для освоения дисциплины**

Для успешного освоения дисциплины требуются результаты обучения по следующим дисциплинам: программирование, алгебра, численные методы.

## **6. Язык реализации**

Русский

## **7. Объем дисциплины**

Общая трудоемкость дисциплины составляет 2 з.е., 72 часов, из которых:  
-лекции: 32 ч.

в том числе практическая подготовка: 32 ч.

Объем самостоятельной работы студента определен учебным планом.

## **8. Содержание дисциплины, структурированное по темам**

Тема 1. Введение.

Суперкомпьютеры мира вчера и сегодня. TOP500 и TOP50. Флопс. Тест Linpack. Архитектура МВС. Критерии оценки производительности параллельного алгоритма. Ускорение и эффективность. Закон Амдала. Основные этапы разработки параллельных программ.

**Тема 2. Основные понятия параллельных вычислительных систем**  
Последовательные, конвейерные и матричные вычисления. 9. Текущий контроль по дисциплине. **Топологии многопроцессорных вычислительных систем.**

**Тема 3. Распараллеливание рекуррентных формул**  
Вычисление частных сумм последовательности числовых значений. Циклическая редукция. Параллельное программирование каскадной схемы суммирования, алгоритма сдваивания и модифицированной каскадной схемы суммирования.

**Тема 4. Параллельное вычисление интегралов**  
Формулы прямоугольников, трапеций и Симпсона для вычисления определенных интегралов. Их распараллеливание. Формулы ячеек, повторного интегрирования, Монте-Карло для вычисления кратных интегралов. Их параллельная реализация.

**Тема 5. Параллельная реализация базовых алгоритмов линейной алгебры**  
Векторные операции. Умножение матрицы на вектор. Подбор топологии МВС. Умножение матриц. Увеличенный объем ОЗУ МВС. Простой блочный алгоритм. Алгоритм Кэннона. Алгоритм Фокса.

**Тема 6. Распараллеливание прямых методов решения СЛАУ**  
LU-разложение матрицы. Одномерные и двумерная декомпозиция матрицы на МВС. Выбор главного элемента. Решение треугольных систем линейных уравнений. Решение трехдиагональных систем линейных уравнений. Метод прогонки и метод циклической редукции.

**Тема 7. Параллельная реализация итерационных методов решения СЛАУ**  
Методы простой итерации – Якоби, Зейделя и релаксации. Метод сопряженных градиентов.

## **9. Текущий контроль по дисциплине**

Текущий контроль по дисциплине проводится путем контроля посещаемости, проведения контрольных работ, тестов по лекционному материалу, и фиксируется в форме контрольной точки не менее одного раза в семестр.

## **10. Порядок проведения и критерии оценивания промежуточной аттестации**

Зачет в восьмом семестре проводится в письменной форме по билетам. Билет содержит десять теоретических вопросов и три задачи. Продолжительность зачета 1,5 часа.

Примерный перечень теоретических вопросов:

1. В каких единицах измеряется производительность компьютера?
2. Что такое степень параллелизма компьютера? Чему она равна для конкретных компьютеров?
3. Что такое степень параллелизма алгоритма?
4. Как определяется ускорение параллельной программы? В каких пределах оно может меняться?
5. Можно ли распространить определение ускорения для параллельного алгоритма? Дайте комментарий.
6. На каких математических принципах базируется алгоритм сдваивания? В чем его преимущество перед обычным алгоритмом суммирования?
7. Сформулируйте закон Амдала. Каким будет максимальное ускорение параллельной программы, в которой 1% арифметических операций выполняется только одним процессором?
8. Пользуясь формальной моделью ускорения, определите условия, при которых будет иметь место максимальный параллелизм, параллелизм среднего уровня и отсутствие ускорения параллельной программы.
9. Перечислите основные этапы разработки параллельной программы. Какие из них зависят от архитектуры ЭВМ, какие не зависят?
10. Каковы требования к проведению первого этапа построения параллельного алгоритма?
11. Чем отличается мелкозернистый параллелизм алгоритма от крупнозернистого?

12. В чем отличие функциональной декомпозиции от декомпозиции по данным?
13. Перечислите типы коммуникаций. Дайте им краткую характеристику.
14. Приведите примеры использования рекуррентных соотношений в вычислительной математике.
15. В чем заключается проблема распараллеливания рекуррентных соотношений?
16. Что собой представляет диаграмма маршрутизации алгоритма и для чего она используется?
17. Опишите алгоритм каскадной схемы суммирования.
18. В чем заключается параллелизм каскадной схемы суммирования?
19. За счет чего удастся обеспечить равномерную работу ПЭ при выполнении каскадной схемы суммирования?
20. Опишите алгоритм сдваивания для вычисления суммы элементов последовательности.
21. Оцените количество операций сложения в алгоритме сдваивания при вычислении суммы  $n=2^q$  чисел.
22. Опишите алгоритм модифицированной каскадной схемы суммирования элементов последовательности (МКСС).
23. Каким ограничениям должно соответствовать число элементов суммируемой последовательности в МКСС?
24. Дайте оценку ускорения и эффективности МКСС.
25. В чем заключается особенность МКСС по сравнению с алгоритмом сдваивания? За счет чего эта особенность проявляется?
26. В чем заключается основная идея метода циклической редукции?
27. Как применяется метод циклической редукции для распараллеливания линейной рекуррентной формулы первого порядка?
28. После применения циклической редукции к линейным рекуррентным формулам удастся ли получить явные формулы?
29. Какова степень параллелизма алгоритма циклической редукции?
30. В чем заключается преимущество обобщенной формулы средних прямоугольников перед формулой трапеций?
31. Докажите, что метод средних прямоугольников имеет больший объем вычислений по сравнению с формулой трапеций.
32. Для чего нужны адаптивные алгоритмы при вычислении интегралов? Опишите принципы их работы.
33. Как построить параллельный алгоритм расчета по квадратурной формуле? Перечислите основные этапы.
34. Чем опасен блочный способ распределения вычислительной нагрузки между процессорами при вычислении интегралов?
35. Как лучше распределять подобласти вычисления интегралов между процессорами при использовании адаптивного алгоритма?
36. Перечислите основные способы вычисления кратных интегралов. Дайте их сравнительный анализ.
37. Какие методы параллельного программирования используются при вычислении кратных интегралов? Функциональная декомпозиция или декомпозиция по данным?
38. Опишите как с использованием общей схемы построения параллельных алгоритмов получить параллельную программу скалярного умножения векторов.
39. Перечислите возможные подходы распараллеливания матрично-векторного умножения.
40. Дайте сравнительный анализ достоинств и недостатков одномерного и двумерного укрупнения при построении параллельного алгоритма умножения матрицы на вектор.
41. Опишите параллельный алгоритм матрично-векторного умножения при максимально необходимом для решения этой задачи числе процессоров.
42. Чем объяснить преимущество топологии «полный граф» по сравнению с топологией «линейка» для случая  $p=n^2$ ?
43. Опишите как работает конвейерная схема умножения матрицы на вектор при  $p=2n$ .
44. Оцените время работы параллельного алгоритма конвейерной схемы умножения матрицы на вектор при  $p=2n$ .
45. В чем заключается преимущество конвейерного способа решения задачи перед декомпозицией по данным?
46. Какие способы декомпозиции задачи матрично-векторного умножения возможны при ограниченном числе используемых процессоров? Дайте сравнительный анализ.
47. Какая топологическая структура наиболее подходит для параллельного матрично-векторного умножения при ограниченном числе используемых процессоров  $p \ll n$ ?
48. Как обеспечить баланс загрузки процессоров при  $n \bmod p \neq 0$ ?
49. Какие способы повышения производительности умножения матриц известны Вам?
50. Какие способы декомпозиции задачи матричного умножения возможны?

51. Можно ли рассматривать параллельные вычисления не с точки зрения быстродействия получения результата, а как средство для решения сверхбольших задач, которые не помещаются в память обычного компьютера?
52. Какие блочные алгоритмы умножения матриц известны? Преимущества и недостатки.
53. Как можно сократить затраты на пересылку данных в алгоритме Фокса?
54. Для чего нужна LU-факторизация матрицы  $A$  в методе исключения Гаусса?
55. Укажите алгоритмическую сложность LU-факторизации.
56. Что может быть выбрано в качестве мелкозернистой фундаментальной подзадачи в алгоритме LU-факторизации? Как связаны они между собой?
57. Какие стратегии укрупнения могут использоваться при построении параллельного алгоритма LU-факторизации? Их особенности.
58. В чем заключаются недостатки 1D строкового укрупнения в параллельном алгоритме LU-факторизации, в котором строки распределяются по порядковым номерам по укрупненным подзадачам?
59. Что такое циклическая схема распределения подзадач?
60. В чем заключаются особенности слоистой строчной схемы с отражениями?
61. Для чего используется опережающая рассылка? В чем заключается ее преимущество?
62. Где меньше коммуникационные затраты в параллельном алгоритме LU-факторизации: в 1D строковом укрупнении или в 2D укрупнении?
63. Как реализуется частичный выбор главного элемента в параллельном алгоритме LU-факторизации?
64. Как можно минимизировать коммуникационные затраты в параллельном алгоритме LU-факторизации?
65. Какова алгоритмическая сложность решения систем с треугольными матрицами?
66. Что можно выбрать в качестве мелкозернистой фундаментальной подзадачи при построении параллельного алгоритма? Как они связаны между собой?
67. Какие стратегии укрупнения можно использовать при построении параллельного алгоритма решения систем с треугольными матрицами? Укажите особенности их реализации.
68. В чем заключаются недостатки 1D строковой схемы укрупнения в параллельном алгоритме решения треугольных систем?
69. Как можно повысить эффективность 1D строковой схемы укрупнения в параллельном алгоритме решения треугольных систем?
70. В чем заключается идея итерационных методов решения СЛАУ и преимущество их применения перед прямыми методами?
71. Какие базовые операции линейной алгебры обычно включают итерационные методы и какими способами они распараллеливаются?
72. Оцените временные затраты на выполнение одной итерации в методе Якоби.
73. Дайте словесное описание параллельного алгоритма метода Якоби.
74. Как следует организовать проверку условия завершения итерационного процесса в параллельном алгоритме метода Якоби?
75. Нарисуйте схему распределения данных по процессорным элементам в параллельном алгоритме метода Якоби.
76. Выполните оценку временных затрат в параллельном алгоритме метода Якоби.
77. Оцените ускорение параллельного алгоритма метода Якоби.
78. В чем заключается проблема распараллеливания метода Зейделя? Прокомментируйте с использованием формул метода.
79. Что собой представляет асинхронная или хаотическая релаксация?
80. В чем заключается проблема применения асинхронной релаксации?
81. Дайте современную классификацию суперкомпьютеров.
82. Каких суперкомпьютеров больше: кластеров или MPP-машин?
83. Что такое тест Linpack? Каков его результат?
84. Что такое флопс?
85. Какая операционная система чаще используется на кластерах?
86. Какая страна производит самые производительные суперкомпьютеры в мире?
87. Какова производительность самого мощного суперкомпьютера в мире?
88. Какой самый производительный суперкомпьютер в России?
89. Что такое top500? Сколько электроэнергии (Вт) потребляет самый мощный суперкомпьютер?
90. Какова производительность кластера ТГУ Cyberia?
91. Самый мощный суперкомпьютер в мире имеет производительность: 1 Exaflops или 1 Petaflops? Дайте правильный ответ.
92. Нарисуйте архитектуру суперкомпьютера с общей памятью.
93. Нарисуйте архитектуру суперкомпьютера с распределенной памятью.
94. Почему конвейерные вычисления имеют преимущества над последовательными?
95. Когда конвейерные вычисления имеют преимущество перед параллельными?
96. Что такое Message Passing Interface? Сколько функций в MPI?

97. Перечислите основных 6 функций MPI.
98. Как нумеруются процессы в MPI?
99. Для чего используется функция MPI\_Finalize?
100. Для чего используется функция MPI\_Wtime?
101. Где больше формальных параметров: в MPI\_Send или в MPI\_Recv?
102. Что такое коммуникатор, процесс и группа процессов в MPI?
103. Что такое MPI\_COMM\_WORLD?
104. Для чего используется функций MPI MPI\_Comm\_Rank?
105. Для чего используется функций MPI MPI\_Comm\_Size?
106. Что происходит после вызова функции MPI\_Init()?
107. Как откомпилировать C++ программу, вызывающую библиотеку MPI?
108. Как запустить MPI-программу на счет?
109. Что такое putty?
110. Зачем нужна программа WinSCP?

#### Примеры задач:

1. Дана последовательность числовых значений целого типа (например,  $x(i)=i$ ,  $i=1, \dots, n$ ). Необходимо написать MPI-программу, реализующую вычисление суммы (частичных сумм) значений заданной последовательности, применив (каскадную схему суммирования, алгоритм сдваивания, модифицированную схему суммирования). Протестировать программу для различных  $n$ .

$$\int_a^b f(x)dx$$

2. Написать MPI-программу приближенного вычисления определенного интеграла с точностью  $\epsilon=10^{-10}$ , используя обобщенную формулу (левых, средних, правых прямоугольников, трапеций, Симпсона, формулу Ньютона «3/8»). Для контроля точности использовать результат расчета на MathCad. Оценить ускорение и эффективность параллельной программы.

3. Написать MPI-программу вычисления двойного интеграла с точностью  $\epsilon=10^{-5}$ , используя метод (Монте-Карло, повторного интегрирования, ячеек). Для контроля точности использовать результат расчета на MathCad. Оценить ускорение и эффективность параллельной программы.

Результаты зачета определяются оценками «зачтено», «не зачтено».

*Чтобы получить «зачтено», нужно ответить на 2/3 теоретических вопросов из билета правильно и решить три задачи.*

### 11. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Moodle» - <https://moodle.tsu.ru/course/view.php?id=6117>

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

в) Методические указания по организации самостоятельной работы студентов.

### 12. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

1. Гергель В. П. Высокопроизводительные вычисления для многопроцессорных многоядерных систем : [учебник] / В. П. Гергель ; Б-ка Нижегород. гос. ун-та им. Н. И. Лобачевского ; Суперкомпьютерный консорциум ун-тов России. – М. : Физматлит [и др.], 2010. – 539 с.

2. Старченко А. В. Методы параллельных вычислений : [учебник] / А. В. Старченко, В. Н. Берцун ; Том. гос. ун-т. – Томск : Изд-во Томского ун-та, 2013. – 224 с.

<http://math.tsu.ru/sites/default/files/mmf2/e-resources/parallel%20comp%20meth.pdf>

3. Линева А. В. Технологии параллельного программирования для процессоров новых архитектур : [учебник] / А. В. Линева, Д. К. Боголепов, С. И. Бахраков ; под ред. В. П. Гергеля ; Нижегородский гос. ун-т им. Н. И. Лобачевского ; [Суперкомпьютерный консорциум университетов России]. – М. : Изд-во Московского университета, 2010. – 148 с.

4. Практикум по методам параллельных вычислений : [учебник] / А. В. Старченко, Е. А. Данилкин, В. И. Лаева, С. А. Проханов ; под ред. А. В. Старченко ; Томский гос. ун-т ; Суперкомпьютерный консорциум ун-тов России. – М. : Изд-во Московского университета, 2010. – 199 с. – URL:  
<http://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000421177>
5. Высокопроизводительные вычисления на кластерах. Томск: Изд-во Том. Ун-та, 2008. – 198 с. <http://math.tsu.ru/sites/default/files/mmf2/e-resources/parallel.pdf>

б) дополнительная литература:

1. Хокни, Джесхоуп. Параллельные ЭВМ. М.: Радио и связь, 1986.
2. Ортега Дж. Введение в параллельные и векторные методы решения линейных систем. М.: Мир, 1991.
3. Фадеева В.Н., Фадеев Д.К. Параллельные вычисления в линейной алгебре.
4. Воеводин В.В. Математические модели и методы в параллельных процессах.
5. Гергель В.П., Стронгин Р.Г. Основы параллельных вычислений для многопроцессорных вычислительных систем. Нижний Новгород: ННГУ, 2002. 122с.
6. Воеводин В.В., Воеводин Вл. В. Параллельные вычисления. -СПб: БХВ -Петербург, 2002. -608 с.

в) ресурсы сети Интернет:

- Официальный сайт [parallel.ru](http://parallel.ru)
- <http://www.netlib.org/blas/>
- <https://software.intel.com/en-us/intel-mkl>
- Массовый открытый онлайн-курс «Введение в параллельное программирование с использованием OpenMP и MPI»  
<https://www.coursera.org/learn/parallelnoye-programirovaniye>
- <http://top500.org>
- <http://top50.supercomputers.ru>

### 13. Перечень информационных технологий

а) лицензионное и свободно распространяемое программное обеспечение:

- Microsoft Office Standart 2013 Russian: пакет программ. Включает приложения: MS Office Word, MS Office Excel, MS Office PowerPoint, MS Office OneNote, MS Office Publisher, MS Outlook, MS Office Web Apps (Word Excel MS PowerPoint Outlook);
- putty (дистрибутив [putty](https://www.putty.org/)) <https://www.putty.org/>
- winscp (дистрибутив [winscp](https://winscp.net/eng/download.php)) <https://winscp.net/eng/download.php>
- публично доступные облачные технологии (Google Docs, Яндекс диск и т.п.).

б) информационные справочные системы:

- Электронный каталог Научной библиотеки ТГУ –  
<http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system>
- Электронная библиотека (репозиторий) ТГУ –  
<http://vital.lib.tsu.ru/vital/access/manager/Index>
- ЭБС Лань – <http://e.lanbook.com/>
- ЭБС Консультант студента – <http://www.studentlibrary.ru/>
- Образовательная платформа Юрайт – <https://urait.ru/>
- ЭБС ZNANIUM.com – <https://znanium.com/>
- ЭБС IPRbooks – <http://www.iprbookshop.ru/>

#### **14. Материально-техническое обеспечение**

Аудитории для проведения занятий лекционного типа.

Аудитории для проведения занятий семинарского типа, индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы, оснащенные компьютерной техникой и доступом к сети Интернет, в электронную информационно-образовательную среду и к информационным справочным системам.

#### **15. Информация о разработчиках**

Старченко Александр Васильевич, профессор, доктор физико-математических наук, механико-математический факультет, зав. каф.

Данилкин Евгений Александрович, кандидат физико-математических наук, механико-математический факультет, доцент.