Об использовании открытого и коммерческого программного обеспечения при разработке научных приложений: ограничения, задачи и опыт их решения

Куликов Игорь

д.ф.-м.н., в.н.с.

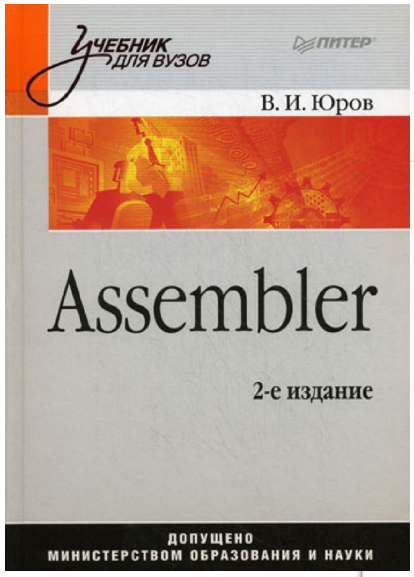
Лаборатория суперкомпьютерного моделирования ИВМиМГ СО РАН

28 апреля 2022 года

План доклада

1. О проблемах использования открытого и коммерческого программного обеспечения
2. О гарантированной точности линейной алгебры – основные положения академика С.К. Годунова
3. О разработке программного кода для решения задач астрофизики с требованиями Фортран 2008
4. Дискуссионные вопросы

Об интегрированных продуктах



Вместо заключения...

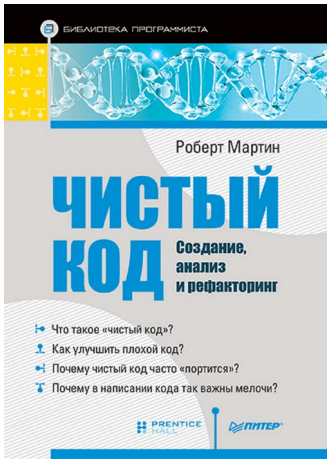
Ну вот и все! Прощаясь с читателем, позволю себе несколько мыслей о жизни вслух (не примите их в качестве наставлений).

В данной книге я попытался посмотреть на язык ассемблера как на обычный язык программирования. При этом хотелось показать, что этот язык ничем не хуже, а в чем-то даже лучше привычных читателю языков высокого уровня. Осознание этого особенно важно для начинающих программистов, которые зачастую с первых шагов в своей карьере попадают в «теплые, нежные, заботливые... (дальше придумайте сами)» объятия различных интегрированных оболочек, сред, студий... Опасно это тем, что такой молодой, неопытный, наивный программист постепенно всецело поддается нежному шелесту этих интегрированных продуктов и засыпает. Спать он может долго, может быть, даже на протяжении всей своей профессиональной карьеры, и, возможно, при этом ему будет сниться сон, суть которого в том, что:

- не нужно думать об оптимизации программы, ведь памяти и запаса быстродействия у современного процессора столько, что хватит на мгновенное псевдопараллельное выполнение десятка даже самых бездарных программ;

О библиотеках и фреймворках

Книги по эффективной разработке кодов



2010

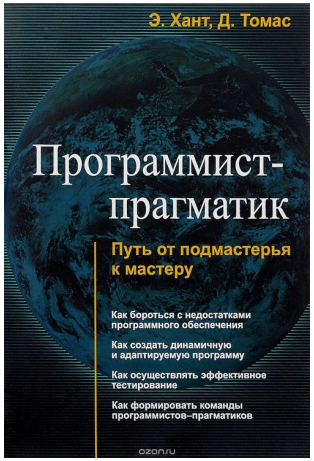
2002

1989

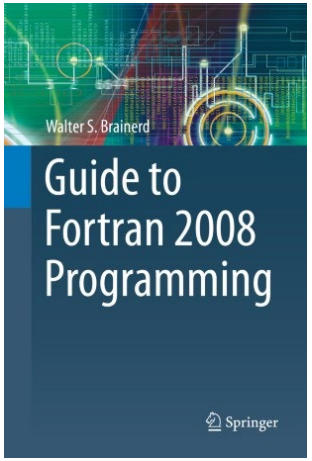
2007

2003

2010



2007



2015

Ограничения готовых решений

- Проблема черного ящика
- Проблема множества зависимостей
- Проблема ограничений

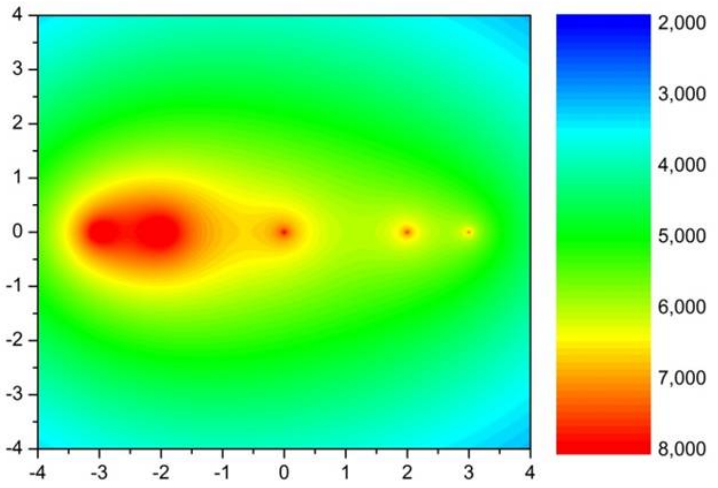
Ограничения готовых решений

- **Проблема черного ящика**
  - **Математика внутри черного ящика**
  - **Точность и корректность результата**
  - **Эффективность реализации**

Ограничения готовых решений

- **Проблема черного ящика**
  - **Математика внутри черного ящика**
  - **Точность и корректность результата**
  - **Эффективность реализации**

Математический аппарат спектральных портретов как ответ на некорректную задачу о численном нахождении собственных чисел несимметричной матрицы



Ограничения готовых решений

- **Проблема черного ящика**
  - **Математика внутри черного ящика**
  - **Точность и корректность результата**
  - **Эффективность реализации**

This necessarily implies positive pressure and $v < 1$. Whenever a non-physical conserved state occurs, we floor the density to $\rho = 10^{-10}$ and the pressure to $P = 10^{-20}$. This occurs roughly once every 10^6 updates in our gamma-ray binary simulations. It occurs in the unshocked pulsar wind, where the density and pressure are lowest and the Lorentz factor the highest. It mostly occurs

Lamberts et al, A&A, 2013

Искусственная «ошибка» в массе и внутренней энергии в одной из миллиона ячеек в релятивистских джетах

Ограничения готовых решений

- **Проблема черного ящика**
  - Математика внутри черного ящика
  - Точность и корректность результата
  - Эффективность реализации

Lamberts et al, A&A, 2013

Искусственная «ошибка» в массе и внутренней энергии в одной из миллиона ячеек в релятивистских джетах

- ✓ **16 «ошибочных» ячеек на одном временном шаге при сетке 256^3**
- ✓ **100 шагов по времени «отравит» всю расчетную область (за 1 % расчетного времени)**

Ограничения готовых решений

- **Проблема черного ящика**
  - Математика внутри черного ящика
  - Точность и корректность результата
  - Эффективность реализации

Lamberts et al, A&A, 2013

Искусственная «ошибка» в массе и внутренней энергии в одной из миллиона ячеек в релятивистских джетах

Помнит ли результат о начальных данных?

Ограничения готовых решений

- **Проблема черного ящика**
  - Математика внутри черного ящика
  - Точность и корректность результата
  - Эффективность реализации

Доклады на Российских суперкомпьютерных днях:

- Petrushov. Advanced Genetic Algorithm in the Problem of Linear Solver Parameters Optimization (2021)
- Chernykh. The Impact of Compiler Level Optimization on the Performance of Iterative Poisson Solver for Numerical Modeling of Protostellar Disks (2021)
- Pershin. GPU Implementation of a Stencil Code with More Than 90% of the Peak Theoretical Performance (2019)
- Andreev. Solving of Eigenvalue and Singular Value Problems via Modified Householder Transformations on Shared Memory Parallel Computing Systems (2019)
- Zamarashkin. Block Lanczos-Montgomery Method over Large Prime Fields with GPU Accelerated Dense Operations (2018)
- Kropotina. Maximus: A Hybrid Particle-in-Cell Code for Microscopic Modeling of Collisionless Plasmas (2018)
- Chernykh. Advanced Vectorization of PPML Method for Intel® Xeon® Scalable Processors (2018)
- Stegailov. Deploying Elbrus VLIW CPU Ecosystem for Materials Science Calculations: Performance and Problems (2018)

Ограничения готовых решений

- **Проблема черного ящика**
  - **Математика внутри черного ящика**
  - **Точность и корректность результата**
  - **Эффективность реализации**

Доклады на Российских суперкомпьютерных днях:

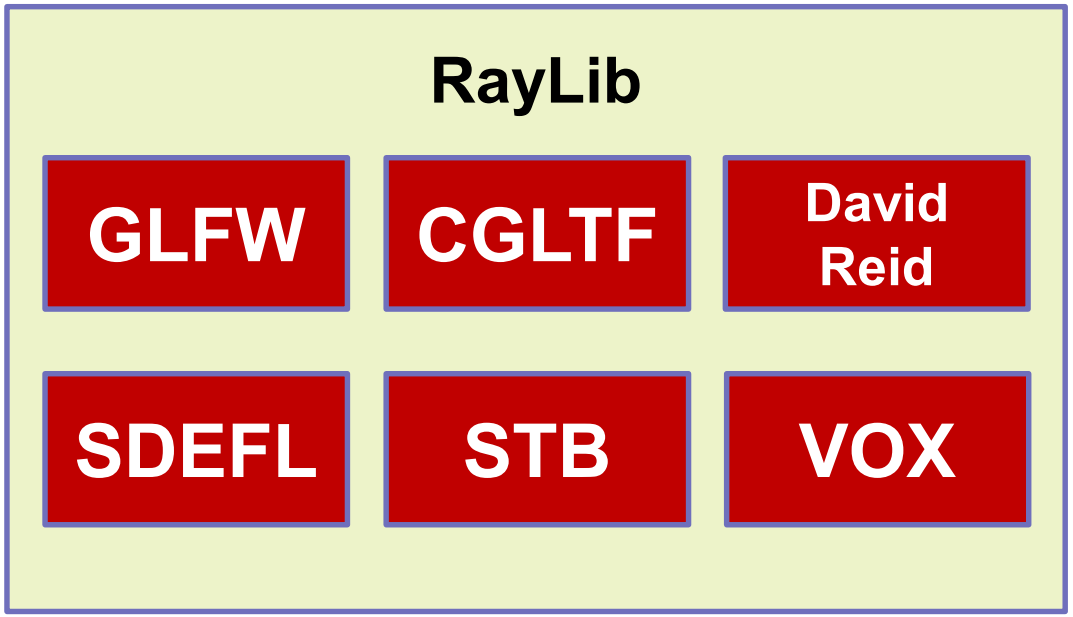
- Petrushov. Advanced Genetic Algorithm in the Problem of Linear Solver Parameters Optimization (2021)
- Chernykh. The Impact of Compiler Level Optimization on the Performance of Iterative Poisson Solver for Numerical Modeling of Protostellar Disks (2021)
- Pershin. GPU Implementation of a Stencil Code with More Than 90% of the Peak Theoretical Performance (2019)
- Andreev. Solving of Eigenvalue and Singular Value Problems via Modified Householder Transformations on Shared Memory Parallel Computing Systems (2019)
- Zamarashkin. Block Lanczos-Montgomery Method over Large Prime Fields with GPU Accelerated Dense Operations (2018)
- Kropotina. Maximus: A Hybrid Particle-in-Cell Code for Microscopic Modeling of Collisionless Plasmas (2018)
- Chernykh. Advanced Vectorization of PPML Method for Intel® Xeon® Scalable Processors (2018)
- Stegailov. Deploying Elbrus VLIW CPU Ecosystem for Materials Science Calculations: Performance and Problems (2018)

Ограничения готовых решений

- **Проблема множества зависимостей**
  - **Все библиотеки в составе кода (RayLib)**
  - **Все библиотеки подключаются снаружи (XQuartz)**

Ограничения готовых решений

- **Проблема множества зависимостей**
  - **Все библиотеки в составе кода (RayLib)**
  - **Все библиотеки подключаются снаружи (XQuartz)**



Ограничения готовых решений

- **Проблема множества зависимостей**
  - **Все библиотеки в составе кода (RayLib)**
  - **Все библиотеки подключаются снаружи (XQuartz)**

XQuartz

Sparkle **Cairo** **FreeGLUT**

libpng **Mesa** **X Keyboard**

Ограничения готовых решений

- **Проблема ограничений**
  - **Испорченный программный код**
  - **Экспортные ограничения**

Ограничения готовых решений

- **Проблема ограничений**
  - **Испорченный программный код**
  - **Экспортные ограничения**

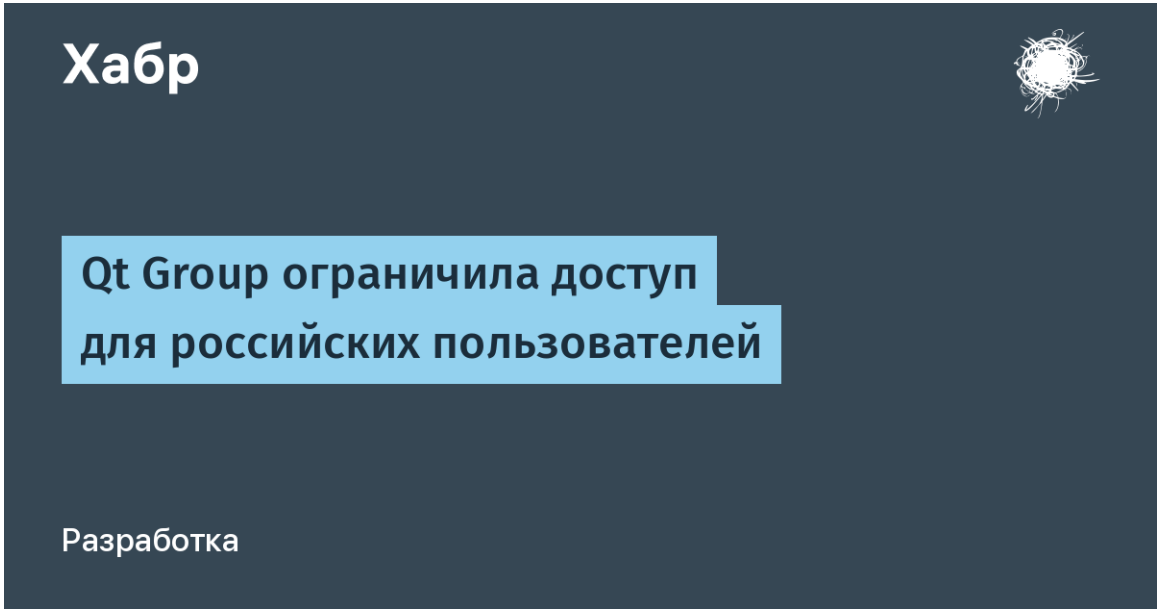
Хабр

Автор faker.js и colors.js намеренно сломал свои пакеты

Разработка

Ограничения готовых решений

- **Проблема ограничений**
  - **Испорченный программный код**
  - **Экспортные ограничения**



Решения ограничений готовых решений

- **Проблема черного ящика**
  - Новые подходы к верификации результата
  - Собственные разработки (из-за надежности)
- **Проблема множества зависимостей**
  - Использование минимального числа библиотек
  - Собственные разработки (из-за переносимости)
- **Проблема ограничений**
  - Параллельный импорт
  - Собственные разработки (из-за токсичности)

ПАРАДОКСЫ ВЫЧИСЛИТЕЛЬНОЙ ЛИНЕЙНОЙ АЛГЕБРЫ И СПЕКТРАЛЬНЫЕ ПОРТРЕТЫ МАТРИЦ

Годунов С.К.

Проблема нахождения собственных чисел

Выберем для исследования следующую матрицу:

$$C=\begin{bmatrix} 289 & 2048 & 336 & 128 & 80 & 32 & 16 \\ 1152 & 30 & 1312 & 512 & 288 & 128 & 32 \\ -29 & -1980 & 756 & 384 & 1008 & 224 & 48 \\ 512 & 128 & 640 & 0 & 640 & 512 & 128 \\ 1053 & 2136 & -604 & -384 & -856 & 800 & 108 \\ -287 & 4 & 1712 & -128 & 1968 & -30 & 2032 \\ -2176 & -187 & -1465 & -512 & -441 & -1152 & -189 \end{bmatrix}$$

Для исследования были выбраны программные системы MATLAB, MAPLE, Scilab и пакет IMSL

Проблема нахождения собственных чисел

На самом деле матрица получена следующим образом:

$$C = L^{-1}RL$$

$$R=\begin{bmatrix} 1 & 2028 & 256 & 128 & 64 & 32 & 16 \\ 0 & -2 & 1024 & 512 & 256 & 128 & 32 \\ 0 & 0 & 4 & 512 & 1024 & 256 & 64 \\ 0 & 0 & 0 & 0 & 512 & 512 & 128 \\ 0 & 0 & 0 & 0 & -4 & 1024 & 156 \\ 0 & 0 & 0 & 0 & 0 & 2 & 2048 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \quad L=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Проблема нахождения собственных чисел

Точное решение	MATLAB	MAPLE	SCILAB	IMSL
$\lambda_1(C) = 0$	$\lambda_1(C) = -6.18$	$\lambda_1(C) = -8.41$	$\lambda_1(C) = -7.14$	$\lambda_1(C) = 0$
$\lambda_2(C) = 1$	$\lambda_2(C) = 5.59 + 2.21i$	$\lambda_2(C) = -7.49 + 3.24i$	$\lambda_2(C) = 6.48 + 2.73i$	$\lambda_2(C) = -5.91 + 2.54i$
$\lambda_3(C) = -1$	$\lambda_3(C) = 5.59 - 2.21i$	$\lambda_3(C) = -7.49 - 3.24i$	$\lambda_3(C) = 6.48 - 2.73i$	$\lambda_3(C) = -5.91 - 2.54i$
$\lambda_4(C) = 2$	$\lambda_4(C) = 1.33 + 5.02i$	$\lambda_4(C) = -1.89 + 7.39i$	$\lambda_4(C) = 1.54 + 6.14i$	$\lambda_4(C) = 5.91 + 2.54i$
$\lambda_5(C) = 4$	$\lambda_5(C) = 1.33 - 5.02i$	$\lambda_5(C) = -1.89 - 7.39i$	$\lambda_5(C) = 1.54 - 6.14i$	$\lambda_5(C) = 5.91 - 2.54i$
$\lambda_6(C) = -4$	$\lambda_6(C) = -3.83 + 3.96i$	$\lambda_6(C) = 5.18 + 5.98i$	$\lambda_6(C) = -4.45 + 4.85i$	$\lambda_6(C) = 3.77i$
$\lambda_7(C) = -2$	$\lambda_7(C) = -3.83 - 3.96i$	$\lambda_7(C) = 5.18 - 5.98i$	$\lambda_7(C) = -4.45 - 4.85i$	$\lambda_7(C) = 3.77i$
Точное решение	MATLAB	MAPLE	SCILAB	IMSL
$\lambda_1(C^T) = 0$	$\lambda_1(C^T) = 8.04$	$\lambda_1(C^T) = -9.14$	$\lambda_1(C^T) = 5.74$	$\lambda_1(C^T) = 0$
$\lambda_2(C^T) = 1$	$\lambda_2(C^T) = 4.95 + 5.66i$	$\lambda_2(C^T) = 8.33 + 3.74i$	$\lambda_2(C^T) = 3.48 + 3.51i$	$\lambda_2(C^T) = 10.89$
$\lambda_3(C^T) = -1$	$\lambda_3(C^T) = 4.95 - 5.66i$	$\lambda_3(C^T) = 8.33 - 3.74i$	$\lambda_3(C^T) = 3.48 - 3.51i$	$\lambda_3(C^T) = 10.89$
$\lambda_4(C^T) = 2$	$\lambda_4(C^T) = -1.81 + 7.01i$	$\lambda_4(C^T) = 1.98 + 8.34i$	$\lambda_4(C^T) = -1.26 + 4.39i$	$\lambda_4(C^T) = -0.61 + 0.42i$
$\lambda_5(C^T) = -2$	$\lambda_5(C^T) = -1.81 - 7.01i$	$\lambda_5(C^T) = 1.98 - 8.34i$	$\lambda_5(C^T) = -1.26 - 4.39i$	$\lambda_5(C^T) = -0.61 - 0.42i$
$\lambda_6(C^T) = 4$	$\lambda_6(C^T) = -7.16 + 3.07i$	$\lambda_6(C^T) = -5.74 + 6.59i$	$\lambda_6(C^T) = -5.09 + 1.87i$	$\lambda_6(C^T) = 0.61 + 0.42i$
$\lambda_7(C^T) = -4$	$\lambda_7(C^T) = -7.16 - 3.07i$	$\lambda_7(C^T) = -5.74 - 6.59i$	$\lambda_7(C^T) = -5.09 - 1.87i$	$\lambda_7(C^T) = 0.61 - 0.42i$

Ни у одного из пакетов результат с точным не совпал!

Представление вещественных чисел

The three fields in an IEEE 754 float

$$A = (-1)^S \times M \times 2^E$$

$$1 \le M < 2$$

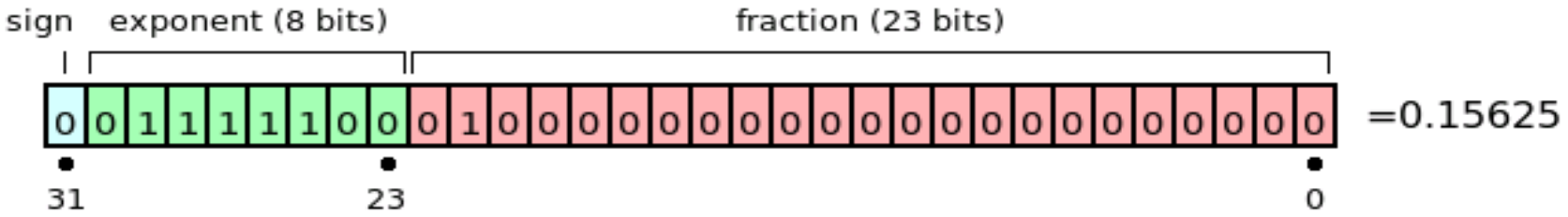
Представление вещественных чисел

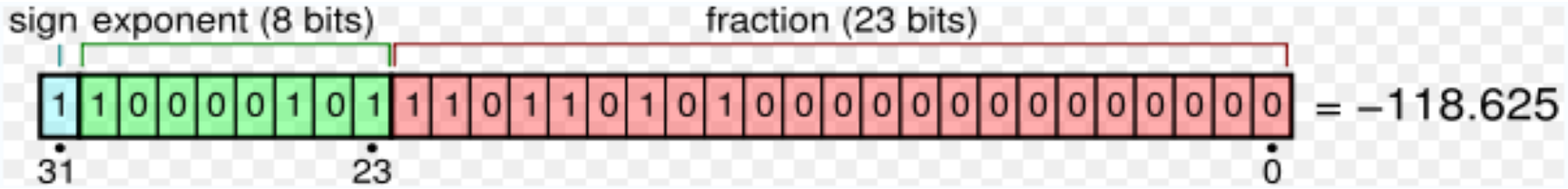
$$0.15625 = 1.25 \times 2^{-3}$$

$$M = 1.01_2 \rightarrow \; fraction = 01_2$$

$$exponent = E + 127_{10} = -3_{10} + 127_{10} = 124_{10} = 01111100_2$$

$$sign = 0$$





Модель округлений

По стандарту IEEE 754 вещественное число представляется в виде:

$$A = \pm m(z)\gamma^{p(z)}, \gamma = 2$$

$$p_- \le p(z) \le p_+$$

$$m(z) = \frac{a_1}{\gamma} + \frac{a_2}{\gamma^2} + \frac{a_3}{\gamma^3} + \ldots + \frac{a_k}{\gamma^k}$$

$$A = (-1)^S \times M \times 2^E$$

$$1 \le M < 2$$

$$\varepsilon_0 = \gamma^{p-}\frac{1}{\gamma}$$

$$\varepsilon_1 = \gamma^1\frac{1}{\gamma^k}$$

$$x^{[c]} \oplus y^{[c]} = (x+y)(1+\alpha)+\beta$$

$$|\alpha| \le \varepsilon_1 \qquad |\beta| \le \varepsilon_0$$

Проблема нахождения собственных чисел

Точное решение	16 знаков	17 знаков	18 знаков	19 знаков
$\lambda_1(C) = 0$	$\lambda_1(C) = -8.41$	$\lambda_1(C) = -6.81$	$\lambda_1(C) = 4.71$	$\lambda_1(C) = 4.18$
$\lambda_2(C) = 1$	$\lambda_2(C) = -7.49 + 3.24i$	$\lambda_2(C) = 6.17 + 2.55i$	$\lambda_2(C) = 2.74 + 2.33i$	$\lambda_2(C) = 2.17 + 1.36i$
$\lambda_3(C) = -1$	$\lambda_3(C) = -7.49 - 3.24i$	$\lambda_3(C) = 6.17 - 2.55i$	$\lambda_3(C) = 2.74 - 2.33i$	$\lambda_3(C) = 2.17 - 1.36i$
$\lambda_4(C) = 2$	$\lambda_4(C) = -1.89 + 7.39i$	$\lambda_4(C) = 1.47 + 5.77i$	$\lambda_4(C) = -0.98 + 2.98i$	$\lambda_4(C) = -0.77 + 1.84i$
$\lambda_5(C) = 4$	$\lambda_5(C) = -1.89 - 7.39i$	$\lambda_5(C) = 1.47 - 5.77i$	$\lambda_5(C) = -0.98 - 2.98i$	$\lambda_5(C) = -0.77 - 1.84i$
$\lambda_6(C) = -4$	$\lambda_6(C) = 5.18 + 5.98i$	$\lambda_6(C) = -4.24 + 4.55i$	$\lambda_6(C) = -4.12 + 1.15i$	$\lambda_6(C) = -3.49 + 0.18i$
$\lambda_7(C) = -2$	$\lambda_7(C) = 5.18 - 5.98i$	$\lambda_7(C) = -4.24 - 4.55i$	$\lambda_7(C) = -4.12 - 1.15i$	$\lambda_7(C) = -3.49 - 0.18i$
Точное решение	16 знаков	17 знаков	18 знаков	19 знаков
$\lambda_1(C^T) = 0$	$\lambda_1(C^T) = -9.14$	$\lambda_1(C^T) = 5.66$	$\lambda_1(C^T) = -5.38$	$\lambda_1(C^T) = -4.24$
$\lambda_2(C^T) = 1$	$\lambda_2(C^T) = 8.33 + 3.74i$	$\lambda_2(C^T) = 3.42 + 3.44i$	$\lambda_2(C^T) = 4.83 + 1.71i$	$\lambda_2(C^T) = 3.61 + 0.51i$
$\lambda_3(C^T) = -1$	$\lambda_3(C^T) = 8.33 - 3.74i$	$\lambda_3(C^T) = 3.42 - 3.44i$	$\lambda_3(C^T) = 4.83 - 1.71i$	$\lambda_3(C^T) = 3.61 - 0.51i$
$\lambda_4(C^T) = 2$	$\lambda_4(C^T) = 1.98 + 8.34i$	$\lambda_4(C^T) = -1.24 + 4.28i$	$\lambda_4(C^T) = 1.14 + 4.01i$	$\lambda_4(C^T) = 0.79 + 2.06i$
$\lambda_5(C^T) = -2$	$\lambda_5(C^T) = 1.98 - 8.34i$	$\lambda_5(C^T) = -1.24 - 4.28i$	$\lambda_5(C^T) = 1.14 - 4.01i$	$\lambda_5(C^T) = 0.79 - 2.06i$
$\lambda_6(C^T) = 4$	$\lambda_6(C^T) = -5.74 + 6.59i$	$\lambda_6(C^T) = -5.01 + 1.81i$	$\lambda_6(C^T) = -3.27 + 3.14i$	$\lambda_6(C^T) = -2.27 + 1.51i$
$\lambda_7(C^T) = -4$	$\lambda_7(C^T) = -5.74 - 6.59i$	$\lambda_7(C^T) = -5.01 - 1.81i$	$\lambda_7(C^T) = -3.27 - 3.14i$	$\lambda_7(C^T) = -2.27 - 1.51i$

Проблема нахождения собственных чисел

20 знаков	21 знаков	22 знаков	23 знаков	24 знаков
$\lambda_1(C) = 3.86$	$\lambda_1(C) = 3.98$	$\lambda_1(C) = 4.01$	$\lambda_1(C) = 4.00$	$\lambda_1(C) = 4.00$
$\lambda_2(C) = 2.00 + 1.04i$	$\lambda_2(C) = -4.01$	$\lambda_2(C) = -3.98$	$\lambda_2(C) = -4.00$	$\lambda_2(C) = -4.00$
$\lambda_3(C) = 2.00 - 1.04i$	$\lambda_3(C) = 2.03$	$\lambda_3(C) = 1.97$	$\lambda_3(C) = 2.00$	$\lambda_3(C) = 2.00$
$\lambda_4(C) = -0.71 + 1.49i$	$\lambda_4(C) = -1.96$	$\lambda_4(C) = -2.01$	$\lambda_4(C) = -2.00$	$\lambda_4(C) = -2.00$
$\lambda_5(C) = -0.71 - 1.49i$	$\lambda_5(C) = 0.86$	$\lambda_5(C) = 1.02$	$\lambda_5(C) = 0.98$	$\lambda_5(C) = 1.00$
$\lambda_6(C) = 2.82$	$\lambda_6(C) = 0.16$	$\lambda_6(C) = -0.18$	$\lambda_6(C) = 0.01$	$\lambda_6(C) = 0.00$
$\lambda_7(C) = -4.09$	$\lambda_7(C) = -1.08$	$\lambda_7(C) = -0.98$	$\lambda_7(C) = -1.00$	$\lambda_7(C) = -1.00$
20 знаков	21 знаков	22 знаков	23 знаков	24 знаков
$\lambda_1(C^T) = 4.01$	$\lambda_1(C^T) = 3.99$	$\lambda_1(C^T) = 4.01$	$\lambda_1(C^T) = 4.00$	$\lambda_1(C^T) = 4.00$
$\lambda_2(C^T) = 1.65 + 0.26i$	$\lambda_2(C^T) = 2.01$	$\lambda_2(C^T) = 1.99$	$\lambda_2(C^T) = 1.99$	$\lambda_2(C^T) = 2.00$
$\lambda_3(C^T) = 1.65 - 0.26i$	$\lambda_3(C^T) = 0.96$	$\lambda_3(C^T) = 1.02$	$\lambda_3(C^T) = 1.00$	$\lambda_3(C^T) = 1.00$
$\lambda_4(C^T) = -0.57 + 0.59i$	$\lambda_4(C^T) = 0.04$	$\lambda_4(C^T) = -0.02$	$\lambda_4(C^T) = -0.01$	$\lambda_4(C^T) = 0.00$
$\lambda_5(C^T) = -0.57 - 0.59i$	$\lambda_5(C^T) = -1.03$	$\lambda_5(C^T) = -0.97$	$\lambda_5(C^T) = -0.99$	$\lambda_5(C^T) = -1.00$
$\lambda_6(C^T) = -2.17$	$\lambda_6(C^T) = -1.98$	$\lambda_6(C^T) = -2.01$	$\lambda_6(C^T) = -2.00$	$\lambda_6(C^T) = -2.00$
$\lambda_7(C^T) = -3.98$	$\lambda_7(C^T) = -4.02$	$\lambda_7(C^T) = -3.99$	$\lambda_7(C^T) = -4.00$	$\lambda_7(C^T) = -4.00$

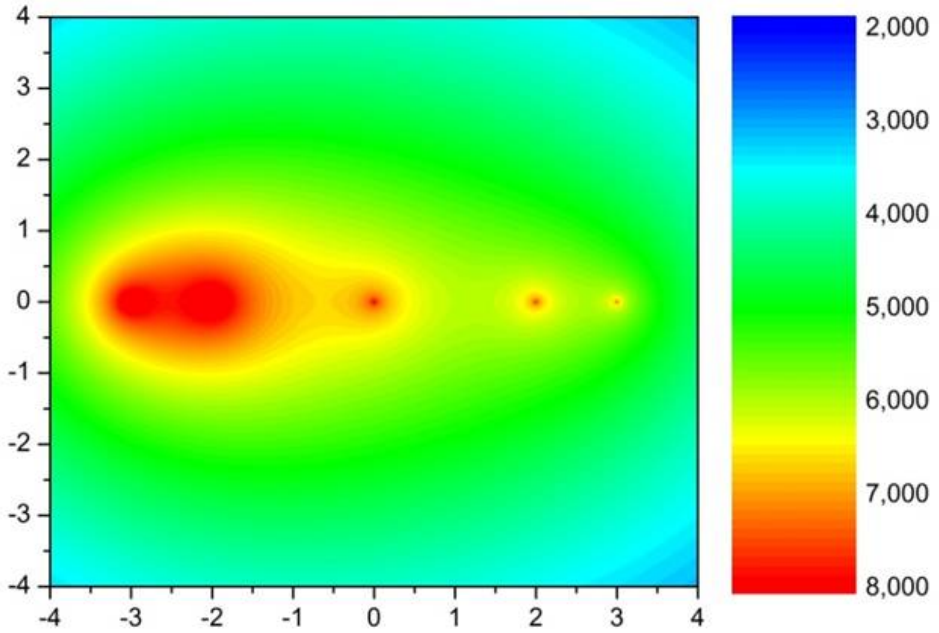
Для определения трёх значащих знаков потребовалось использовать двадцать четыре знака для представления самих матриц и для организации самих вычислений

Понятие ε-спектра

Число λ принадлежит ε-спектру, если

$$\left\|\left(\lambda I - A\right)^{-1}\right\| \geq \frac{1}{\varepsilon \left\|A\right\|}$$

$$\begin{pmatrix} -2 & 25 & 0 & 0 & 0 & 0 & 0 \\ 0 & -3 & 10 & 3 & 3 & 3 & 0 \\ 0 & 0 & 2 & 15 & 3 & 3 & 0 \\ 0 & 0 & 0 & 0 & 15 & 3 & 0 \\ 0 & 0 & 0 & 0 & 3 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 25 \\ 0 & 0 & 0 & 0 & 0 & 0 & -3 \end{pmatrix}$$



Проблема решения систем линейных алгебраических уравнений

Выберем для исследования следующую матрицу и вектор правой части:

$$A=\begin{bmatrix} 289 & 2044 & 336 & 128 & 80 & 32 & 16 \\ 1152 & 30 & 1312 & 512 & 288 & 128 & 32 \\ -29 & -1980 & 756 & 384 & 1008 & 224 & 48 \\ 512 & 128 & 640 & 1 & 640 & 512 & 128 \\ 1053 & 2136 & -604 & -384 & -856 & 800 & 108 \\ -287 & 4 & 1712 & -128 & 1968 & -30 & 2032 \\ -2176 & -185 & -1463 & -512 & -439 & -1152 & -187 \end{bmatrix} \quad x=\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad f=Ax=\begin{bmatrix} 2925 \\ 3454 \\ 411 \\ 2561 \\ 2253 \\ 5271 \\ -6114 \end{bmatrix}$$

$$\mu(A)=\|A\|\left\|A^{-1}\right\|=3.59\cdot 10^{16}$$

Проблема решения систем линейных алгебраических уравнений

MATLAB	MAPLE	SCILAB	IMSL
2.398 0.999 –0.398 1.000 2.398 –0.398 1.001	0.287 1.001 1.712 1.000 0.287 1.712 0.998	3.588 0.998 –1.588 1.000 3.588 –1.588 1.001	1.601 0.999 0.393 1.000 1.601 0.392 1.001

Ни один из пакетов и библиотек не выдал предупреждение о плохой обусловленности матрицы!

Проблема решения систем линейных алгебраических уравнений

$$\left((1-t)A+tI\right)x=f(t)$$

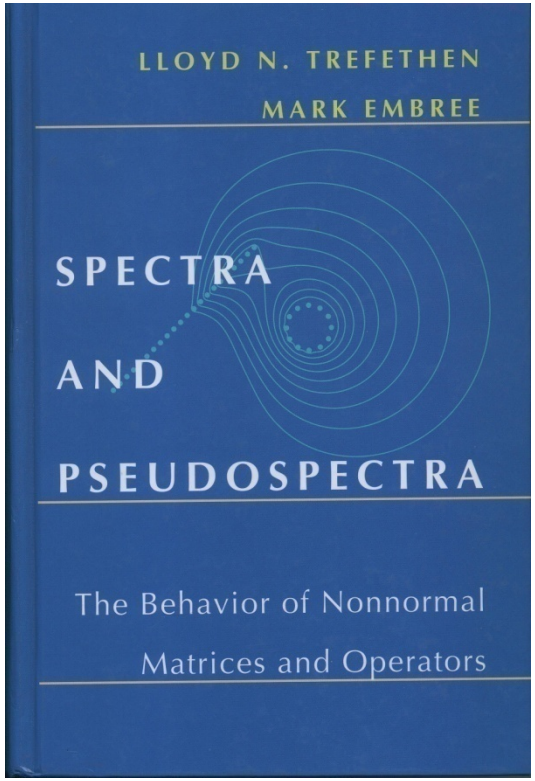
t	Число обусловленности	$\| x^{exact}-x^{t} \|$	$\| b^{exact}-Cx^{t} \|$
0.9999	2.388666e+00	5.000000e-16	0.000000e+00
0.999	1.248222e+02	2.837252e-15	2.449490e-15
0.99	1.322218e+08	9.767385e-09	2.500000e-14
0.98	2.822989e+10	9.073249e-07	3.572114e-14
0.97	5.723564e+11	6.171335e-06	1.000000e-14
0.96	4.807893e+12	2.919485e-05	1.421267e-13
0.95	2.525355e+13	4.500942e-03	1.414214e-13
0.94	9.927891e+13	3.620296e-03	1.445683e-13
0.93	3.180642e+14	1.915018e-02	2.547548e-13
0.92	9.203372e+14	1.212315e-02	3.112876e-13
0.91	2.220468e+15	8.000636e-02	1.743560e-13
0.90	6.661588e+15	1.350264e-01	4.242641e-13
0.89	1.482758e+16	2.841732e-01	3.618011e-13
0.88	3.775640e+16	2.551333e+00	2.794638e-13

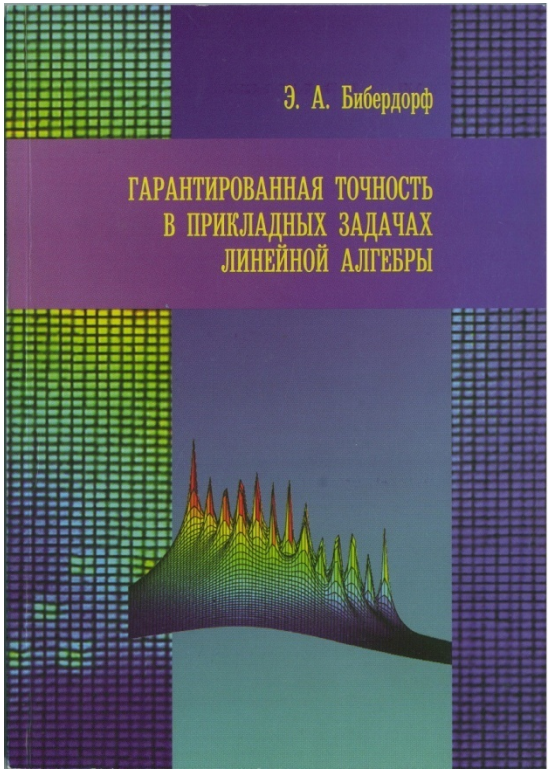
Проблема решения систем линейных алгебраических уравнений

16 знаков	17 знаков	18 знаков	19 знаков
0.28713 1.00035 1.71286 0.99999 0.28713 1.71286 0.99964	-0.26817 1.00062 2.26817 0.99999 -0.26817 2.26817 0.99937	-6.15397 1.00352 8.15398 0.99999 -6.15398 8.15397 0.99647	-0.19355 1.00058 2.19355 0.99999 -0.19355 2.19355 0.99941
20 знаков	**21 знаков**	**22 знаков**	**23 знаков**
0.91202 1.00004 1.08797 0.99999 0.91202 1.08797 0.99995	0.99835 1.00000 1.00164 0.99999 0.99835 1.00164 0.99999	1.00232 0.99999 0.99767 1.00000 1.00232 0.99767 1.00000	0.99975 1.00000 1.00024 0.99999 0.99975 1.00024 0.99999

Для определения четырех значащих знаков потребовалось использовать двадцать три знака для представления самих матриц и для организации самих вычислений

Новая математика с гарантированной оценкой точности

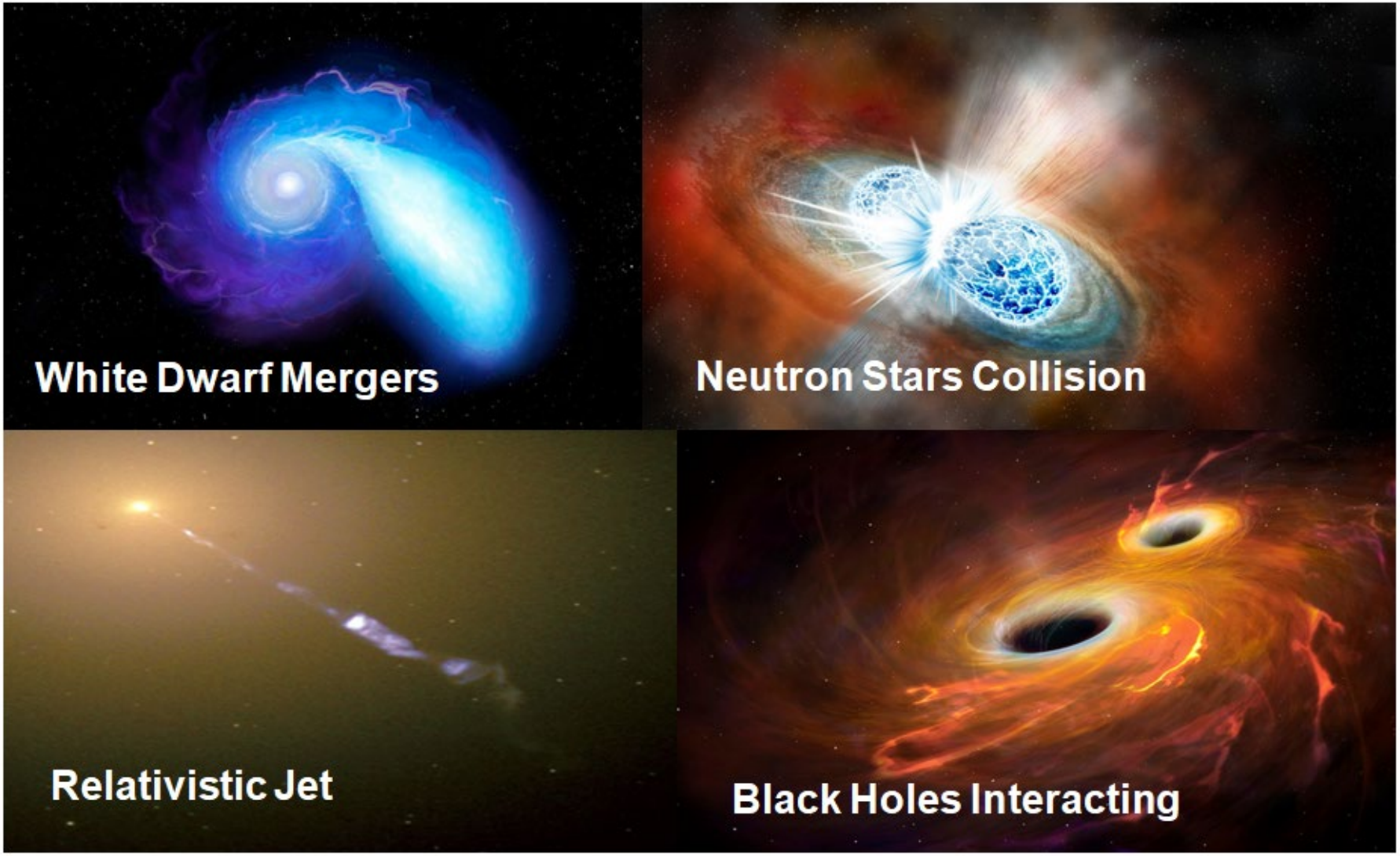




ACCUNA: ACCUracy linear algebra Numerical Algorithms

Куликов И.М. (2013)

Вселенная – динамическая система с множеством взаимодействий



Топ 10 (ноябрь, 2021)

1. Supercomputer Fugaku - A64FX 48C 2.2GHz, Fujitsu
2. Summit - IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, IBM
3. Sierra - IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, IBM/NVIDIA
4. Sunway TaihuLight - Sunway SW26010 260C 1.45GHz, Sunway
5. Perlmutter - AMD EPYC 7763 64C 2.45GHz, NVIDIA A100, HPE
6. Selene - AMD EPYC 7742 64C 2.25GHz, NVIDIA A100, NVIDIA
7. Tianhe-2A - Intel Xeon E5-2692v2 12C 2.2GHz, Matrix-2000, Inspur
8. JUWELS - AMD EPYC 7402 24C 2.8GHz, NVIDIA A100, Bull
9. HPC5 - Intel Xeon Gold 6252 24C 2.1GHz, NVIDIA Tesla V100, Dell
10. Voyager-EUS2 - AMD EPYC 7V12 48C 2.45GHz, NVIDIA A100, Microsoft

Топ 10 (ноябрь, 2021)

1. Supercomputer Fugaku - A64FX 48C 2.2GHz, Fujitsu
2. Summit - IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, IBM
3. Sierra - IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, IBM/NVIDIA
4. Sunway TaihuLight - Sunv
5. Perlmutter - AMD EPYC 7
6. Selene - AMD EPYC 7742
7. Tianhe-2A - Intel Xeon E5-
8. JUWELS - AMD EPYC 7402 24C 2.8GHz, NVIDIA A100, Bull
9. HPC5 - Intel Xeon Gold 6252 24C 2.1 NVIDIA Tesla Dell
10. Voyager-EUS2 - AMD EPYC 7V12 48C 2.45GHz, NVIDIA A100, Microsoft

SIMD-based суперЭВМ

26 апреля 2017 года
(Новосибирск, ССКЦ)

Параллельная реализация вычислительного ядра

Декомпозиция расчетной области

Распределение областей (MPI)

Распределение потоков (OpenMP)

Векторизация (AVX 512)

Вызовы векторных архитектур:

- организация вычислений 8-элементными векторами
- ассоциативность кэш памяти
- выравнивание данных
- распределение работ
- зависимость по данным

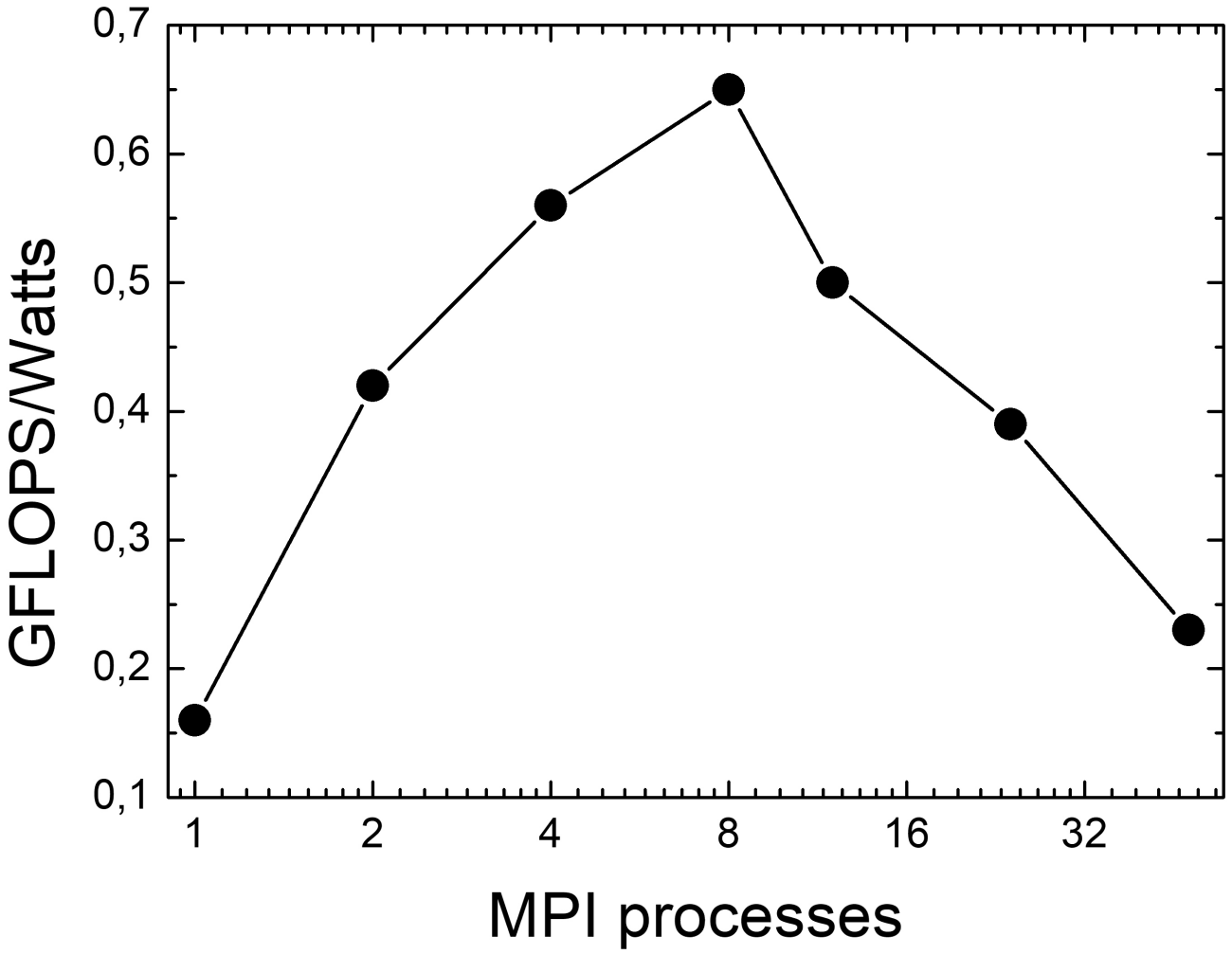
НКС-1П ССКЦ

48-кратное ускорение на узле

180 ГФЛОПС производительности

96% масштабируемость

CO_2 impact



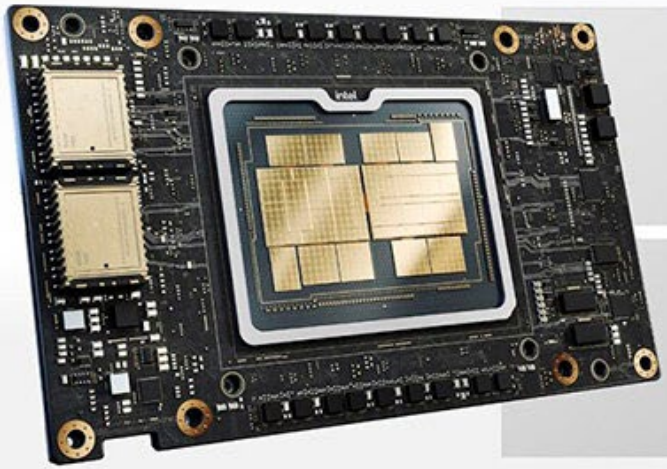
7 NVIDIA GPU суперкомпьютеров в Топ 10

1. Supercomputer Fugaku - A64FX 48C 2.2GHz, Fujitsu
2. Summit - IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, IBM
3. Sierra - IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, IBM/NVIDIA
4. Sunway TaihuLight - Sunway SW26010 260C 1.45GHz, Sunway
5. Perlmutter - AMD EPYC 7763 64C 2.45GHz, NVIDIA A100, HPE
6. Selene - AMD EPYC 7742 64C 2.25GHz, NVIDIA A100, NVIDIA
7. Tianhe-2A - Intel Xeon E5-2692v2 12C 2.2GHz, Matrix-2000, Inspur
8. JUWELS - AMD EPYC 7402 24C 2.8GHz, NVIDIA A100, Bull
9. HPC5 - Intel Xeon Gold 6252 24C 2.1GHz, NVIDIA Tesla V100, Dell
10. Voyager-EUS2 - AMD EPYC 7V12 48C 2.45GHz, NVIDIA A100, Microsoft

Новое поколение графических ускорителей Intel Ponte Vecchio с 45 TFLOPS

Ponte Vecchio

Execution Progress



A0 Silicon Current Status

> 45 TFLOPS	FP32 Throughput
> 5 TBps	Memory Fabric Bandwidth
> 2 TBps	Connectivity Bandwidth

For workloads and configurations visit www.intel.com/ArchDay21claims. Results may vary.

Architecture Day 2021

Under embargo until August 19th at 6:00 am Pacific time

intel. 179

Вместо CUDA: Intel oneAPI

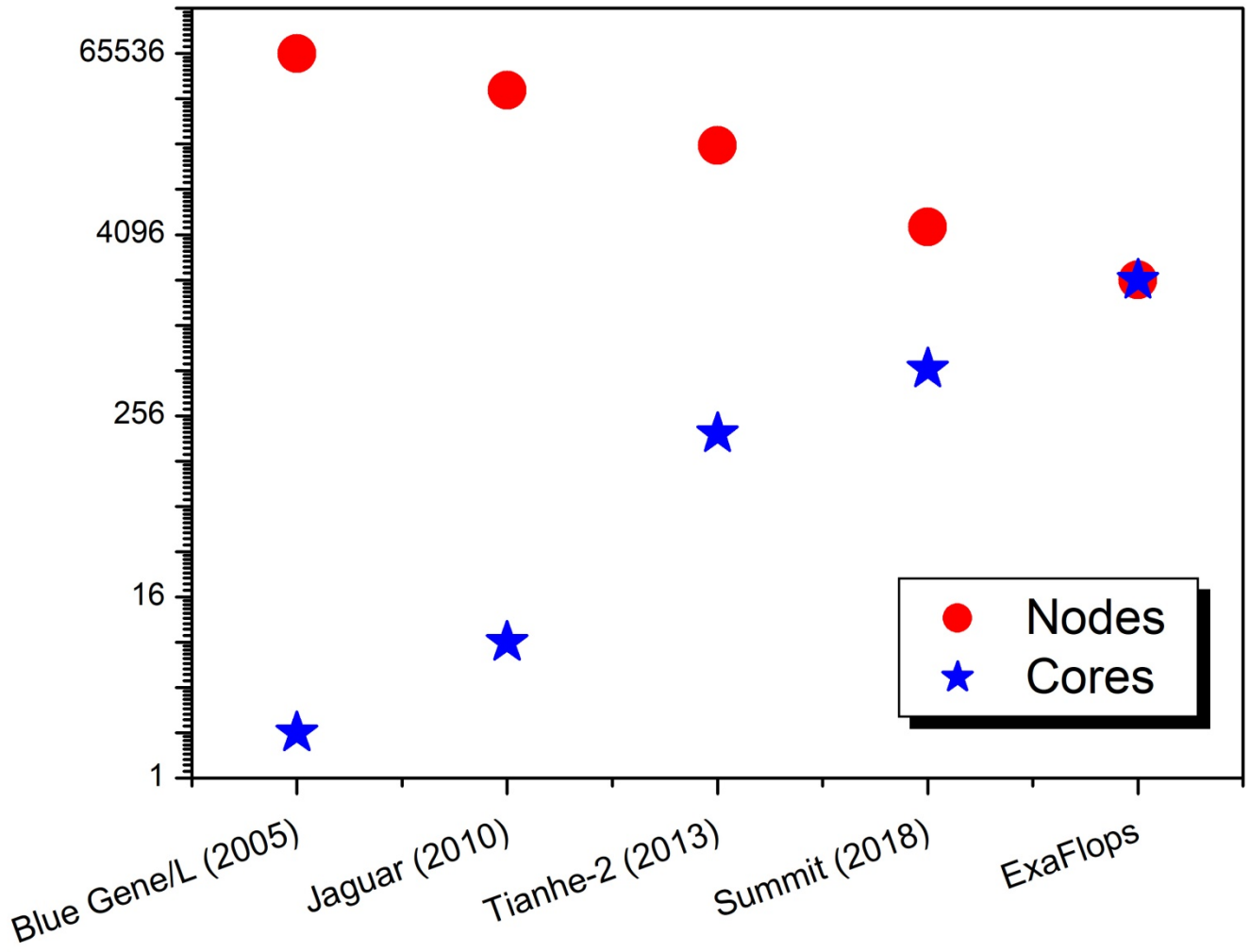
```
#pragma omp target parallel for reduction(+:dot) map(to: a,b)
for (int i = 0; i < N; i++)
    dot = dot + a[i] * b[i];
```

Задел: AstroPhi код

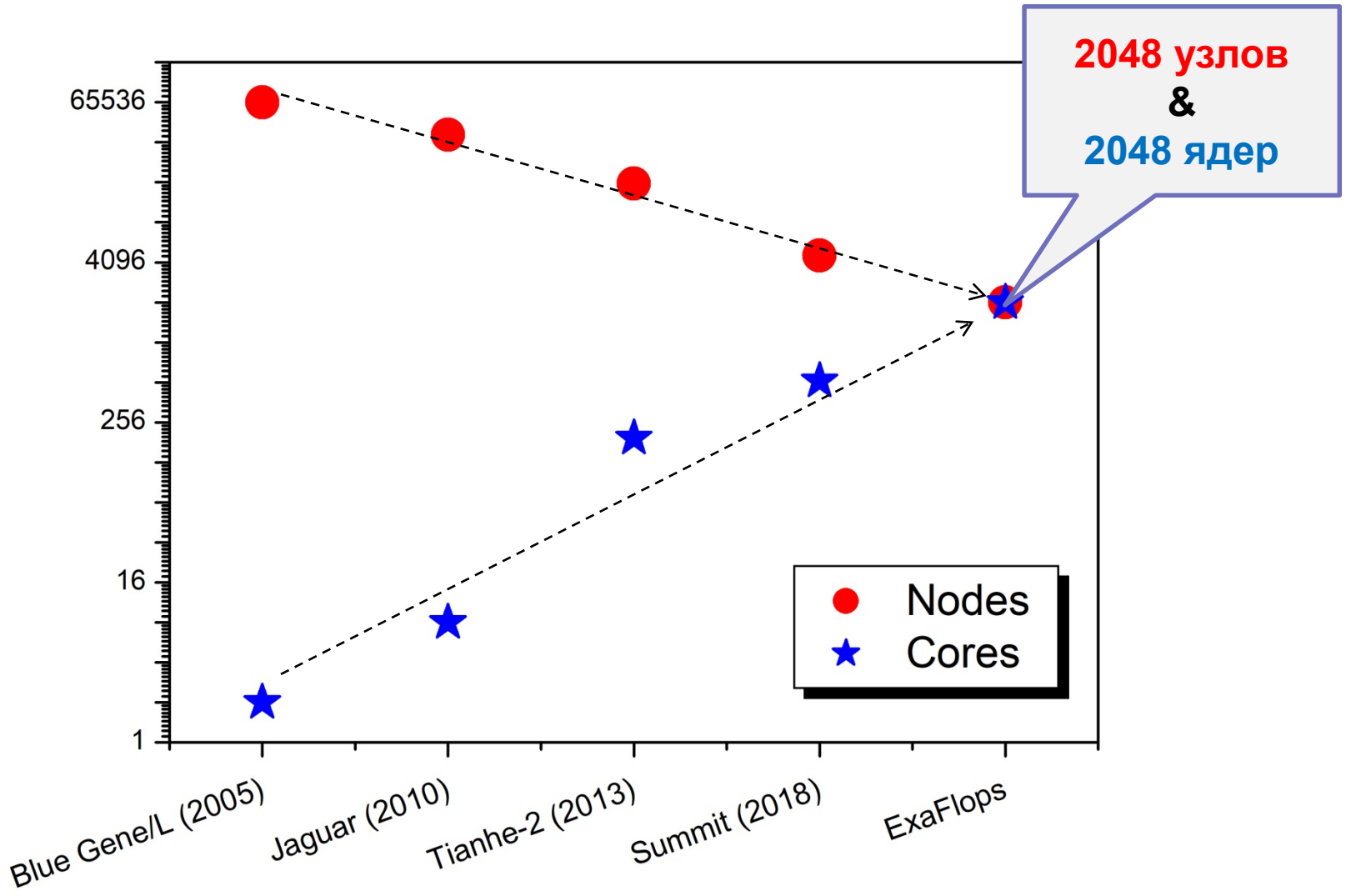
```
#pragma offload target (mic) in(a:length(N), b:length(N))
{
 #pragma omp parallel for reduction(+:dot)
 for (int i = 0; i < N; i++)
     dot = dot + a[i] * b[i];
}
```

Kulikov I.M., Chernykh I.G., Snytnikov A.V., Glinskiy B.M., Tutukov A.V. AstroPhi: A code for complex simulation of dynamics of astrophysical objects using hybrid supercomputers // **Computer Physics Communications. - 2015. - V. 186. - P. 71-80.**

Архитектура суперЭВМ



Архитектура суперЭВМ



Стратегия разработки кодов

- Использование ускорителей и сопроцессоров
- Использование intrinsics-функций (язык Си)
- Использование векторных операций (Фортран)
- Использование новейших инструкций OpenMP
- Возможная замена OpenMP на std::thread C++ 11
- Минимизация сетевых коммуникаций (MPI 3 & CAF)
- Достижения требований “C++ or Fortran Only”
- Архитектурно-ориентированные коды

Гравитация как ключевой фактор

Итерационный метод решения уравнения Пуассона на основе МСГ

$$\Delta\Phi = 4\pi\rho$$

$$\frac{\partial}{\partial t}\times \qquad \nabla\bullet F = 4\pi\rho$$

$$\frac{\partial}{\partial t}\nabla\bullet F = 4\pi\frac{\partial\rho}{\partial t} = -4\pi\nabla\bullet(\rho u)$$

$$\nabla\bullet\left(\frac{\partial F}{\partial t}\right) = -\nabla\bullet(4\pi\rho u)$$

$$\frac{\partial F}{\partial t} = \nabla\times \mathrm{A} - 4\pi\rho u$$

Чуев, ЖВМиМФ, 2020

$$\frac{\partial F}{\partial t} \approx \nabla\times(F\times u) - 4\pi\rho u$$

$$\Phi^{n}_{ikl}$$

Представление *Ax = f*
в операторном виде на сетке
(без построения СЛАУ)

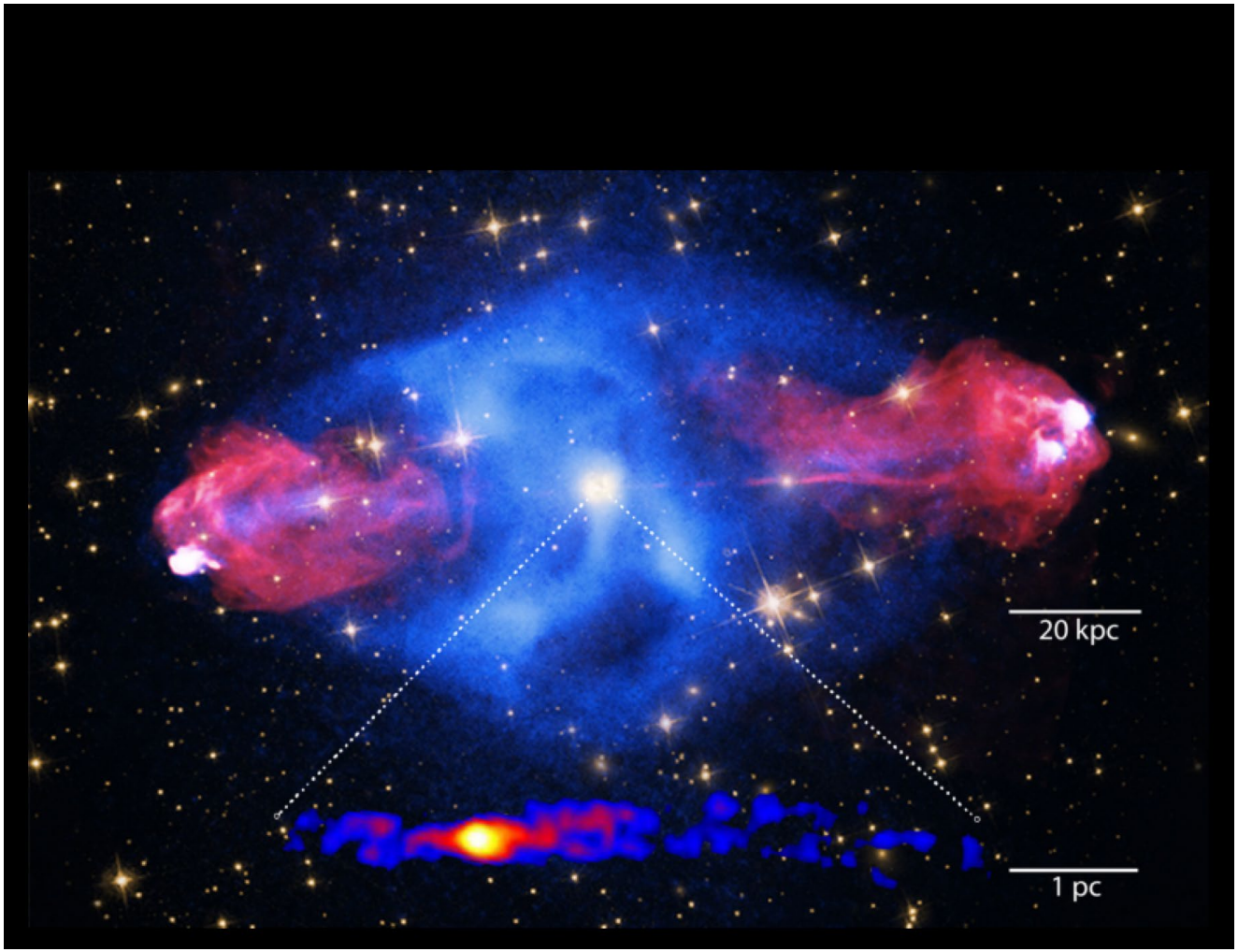
Запись всех конструкций через цикл forall и операции над массивами

$$\Phi^{n+1}_{ikl,0}$$

80 GFLOPS
vs.
3x Intel MKL

$$\mathrm{O}\left(n^{1+k}\right), k \ll 1$$

Эволюция релятивистского внегалактического джета



Развитие неустойчивости и обратное течение в релятивистском джете

nature.com > nature communications > articles > article

MENU

nature COMMUNICATIONS

Article | OPEN

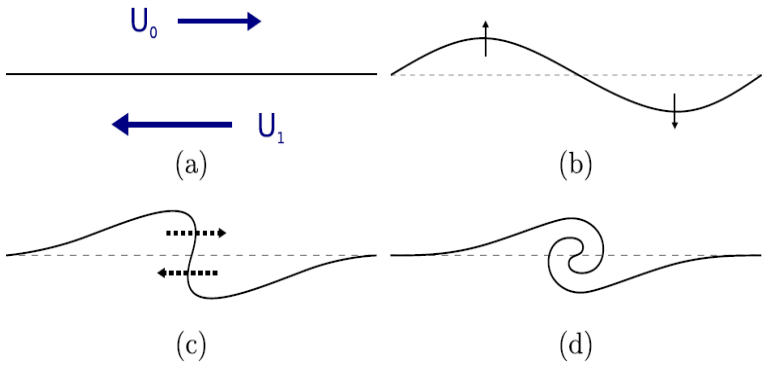
A galactic microquasar mimicking winged radio galaxies

Josep Martí ✉, Pedro L. Luque-Escamilla, Valentí Bosch-Ramon & Josep M. Paredes

Nature Communications **8**,
Article number: 1757 (2017)
doi:10.1038/s41467-017-01976-5
Download Citation

Galaxies and clusters
High-energy astrophysics

Received: 17 April 2017
Accepted: 27 October 2017
Published online: 24 November 2017



We have thus a clear case of a relativistic jet impinging on the ISM. This renders the **hydrodynamic backflow** as the most plausible explanation of the very, very long secondary lobes of GRS 1758−258.

Martí et al. 2017, Nature Communications

Computational Challenges

Масштабы

- Размер внегалактического джета 100 кпк
- Фрагментация на уровне 10 пк
  - Разрешение $10\ 000^3$
- Плотность межзвездной среды 1 $см^{-3}$
- Плотность фронта джета 10^5 $см^{-3}$
  - Контраст плотности 10^5
- Эволюция джета 10 Myr
- Прохождение звуковой волны через джет 1 Kyr
  - Необходимость 10^4 шагов по времени

Physics Challenges

Физические процессы

- Специальная релятивистская гидродинамика
- Сверхзвукововые течения с числом Маха ~ 3
- Лоренц-фактор ~ 100
- Нелинейная связь физических (примитивных) и консервативных переменных

Численный метод решения уравнений специальной релятивистской гидродинамики

$$\frac{\partial}{\partial t}\begin{pmatrix}\Gamma\rho\\ \Gamma^2\rho h\vec{u}\\ \Gamma^2\rho h-p\end{pmatrix}+\nabla\bullet\begin{pmatrix}\Gamma\rho\vec{u}\\ \Gamma^2\rho h\vec{u}\vec{u}-p\\ \Gamma^2\rho h\vec{u}\end{pmatrix}=0 \qquad h=1+\frac{\gamma}{\gamma-1}\frac{p}{\rho} \qquad \Gamma=\frac{1}{\sqrt{1-u^2}}$$

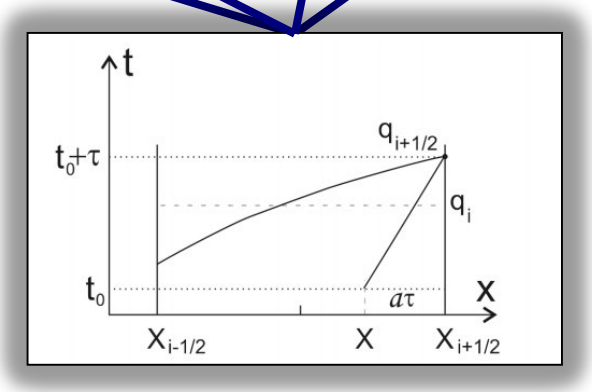
$$\frac{q_k^{n+1}-q_k^n}{\tau}+\frac{F_{k+1/2}-F_{k-1/2}}{h}=0$$

$$F_{k\pm1/2}=\frac{F(-\lambda\tau)+F(\lambda\tau)}{2}+\frac{\Im(q)}{2}\times\left(q(-\lambda\tau)-q(\lambda\tau)\right)$$

Численный метод решения уравнений специальной релятивистской гидродинамики

$$F_{k\pm 1/2} = \frac{F(-\lambda\tau) + F(\lambda\tau)}{2} + \frac{\Im(q)}{2} \times \left(q(-\lambda\tau) - q(\lambda\tau)\right)$$

Кусочно-параболическое представление решения



Ограничение скорости скоростью света. Схема типа Лакса-Фридрихса

Принцип максимума для параболических уравнений (!)

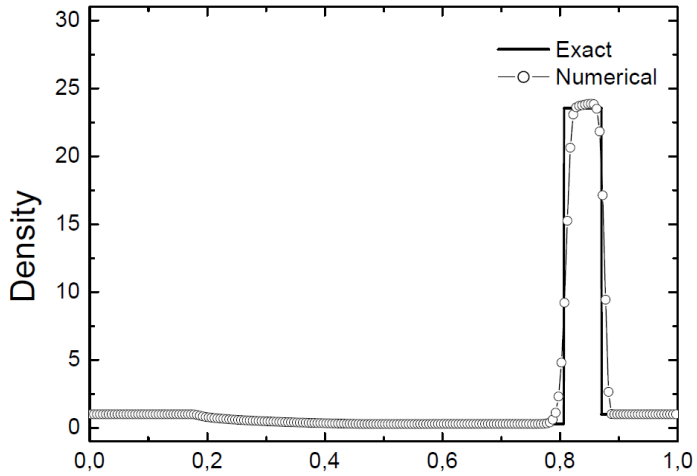
Достигается на границе области
или в начальных данных

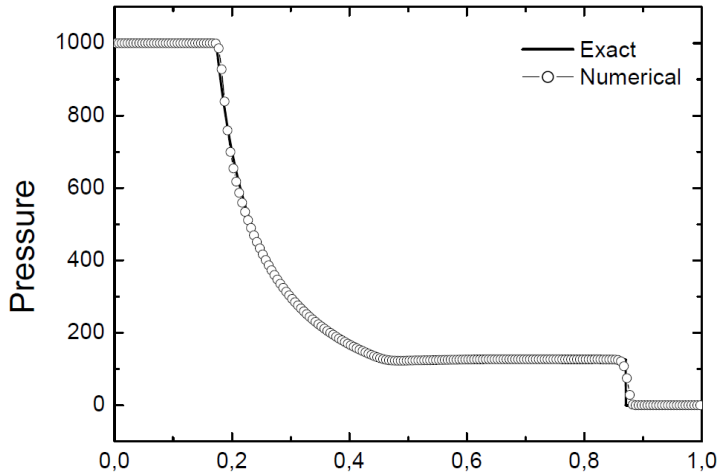
Физическое ограничения скорости
скоростью света |v| ≤ c = 1

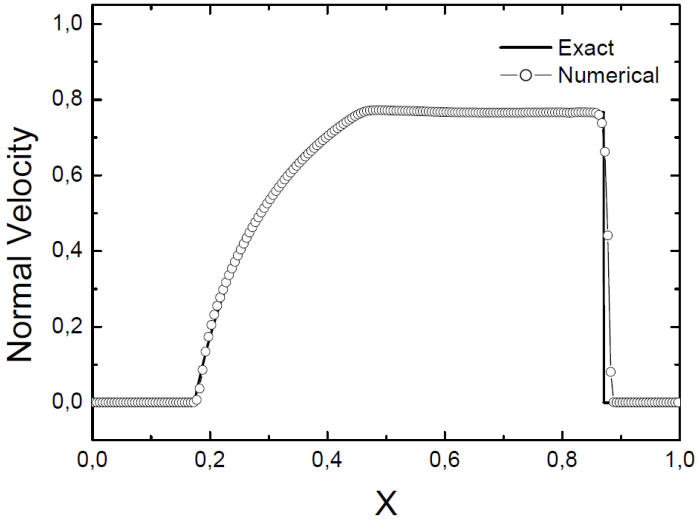
$$F_{k\pm 1/2} = \frac{F(-\lambda\tau)+F(\lambda\tau)}{2} + \frac{\Im(q)}{2}\times\left(q(-\lambda\tau)-q(\lambda\tau)\right)$$

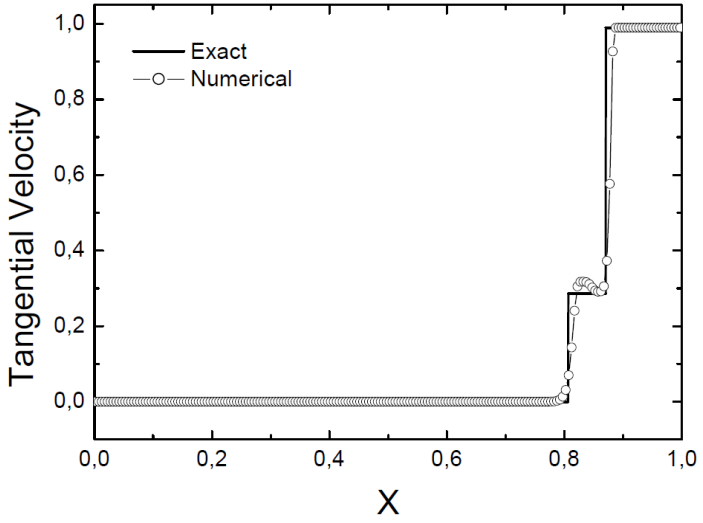
$$F_{k\pm 1/2} = \frac{F(-\lambda\tau)+F(\lambda\tau)}{2} + \frac{1}{2}\times\left(q(-\lambda\tau)-q(\lambda\tau)\right)$$

Верификация численного метода

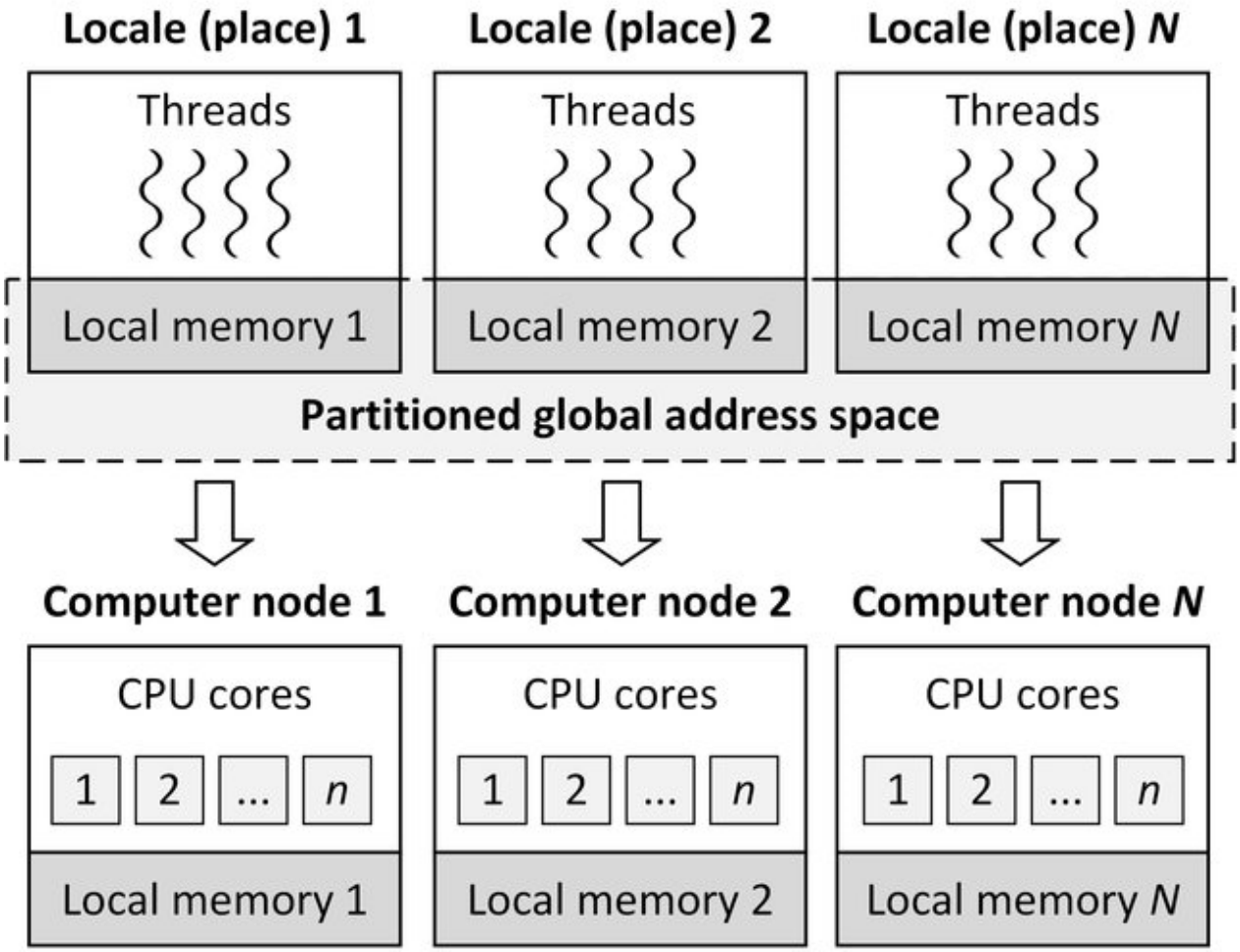








Технология PGAS
Partitioned Global Address Space

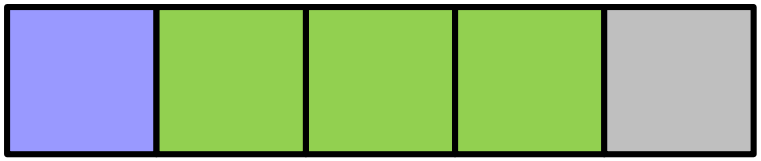


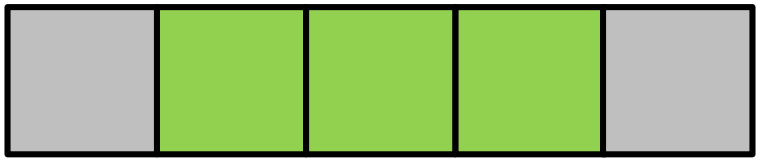
(*) Kulagin, Paznikov & Kurnosov, PaCT 2015, LNCS 9251

Реализация PGAS

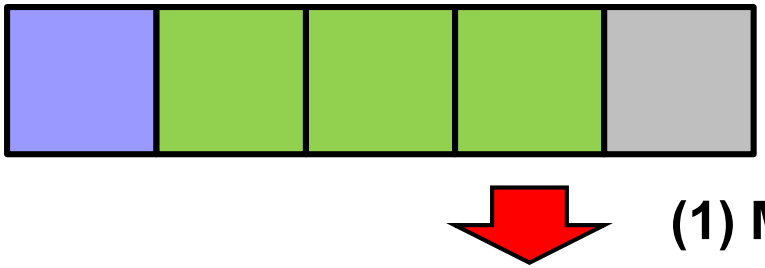
- Coarray Fortran (стандарт Фортран 2008)
- Unified Parallel C (диалект стандарта ISO C)
- Chapel (параллельный язык Cray)
- X10 (параллельный язык IBM)
- Fortress (параллельный язык Sun)
- UPC++ (библиотека шаблонов)
  - Remote Memory Access
  - Remote Procedure Call
- Global Arrays (библиотека)

Использование классического MPI

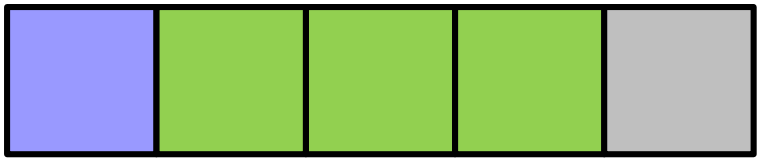




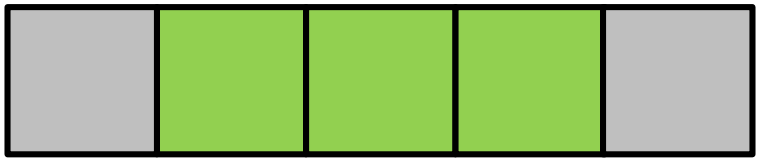
Использование классического MPI



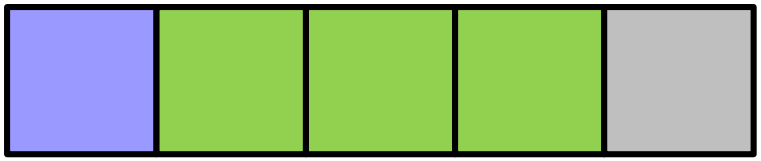
Использование классического MPI

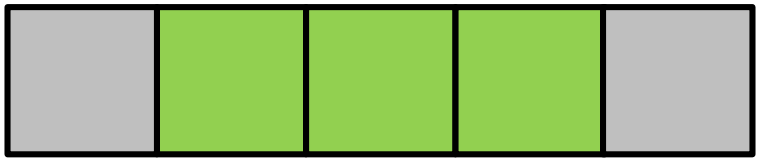


(2) MPI_Recv



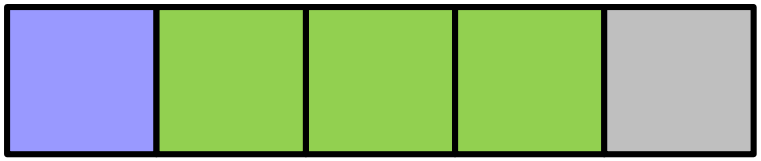
Использование классического MPI

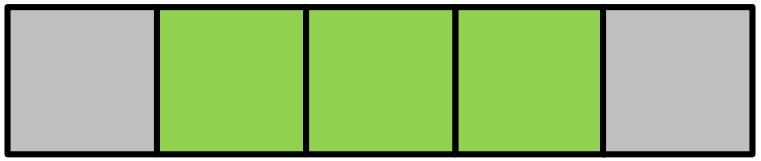


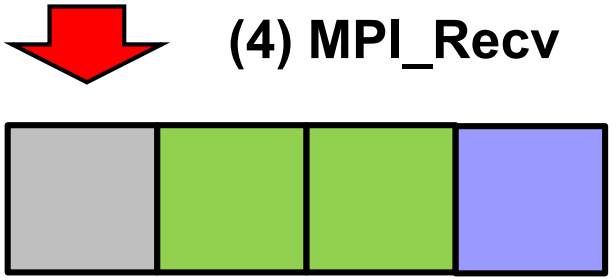


(3) MPI_Send

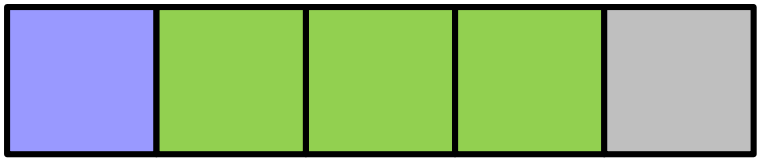
Использование классического MPI

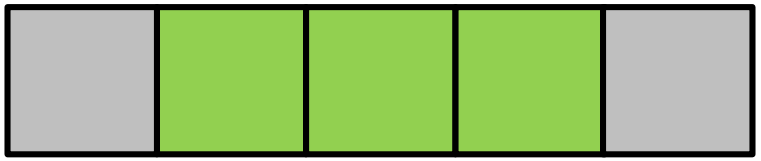






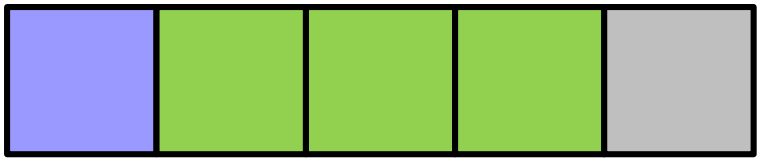
Использование классического MPI

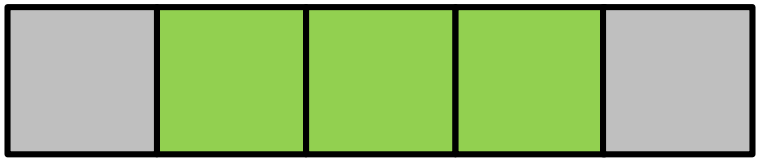




(5) MPI_Send

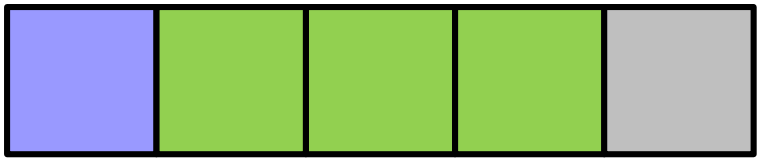
Использование классического MPI



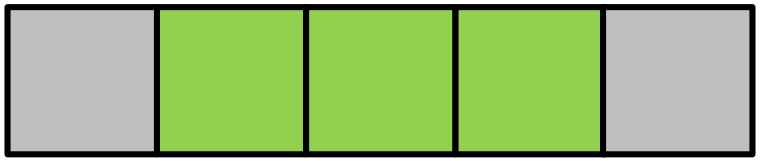


(6) MPI_Recv

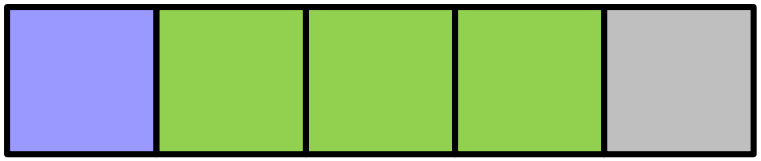
Использование классического MPI



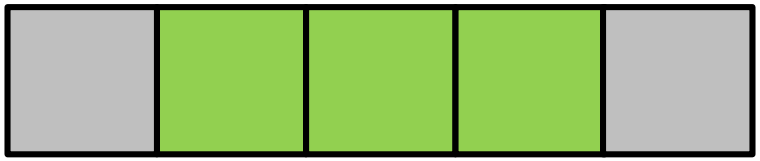
(7) MPI_Send



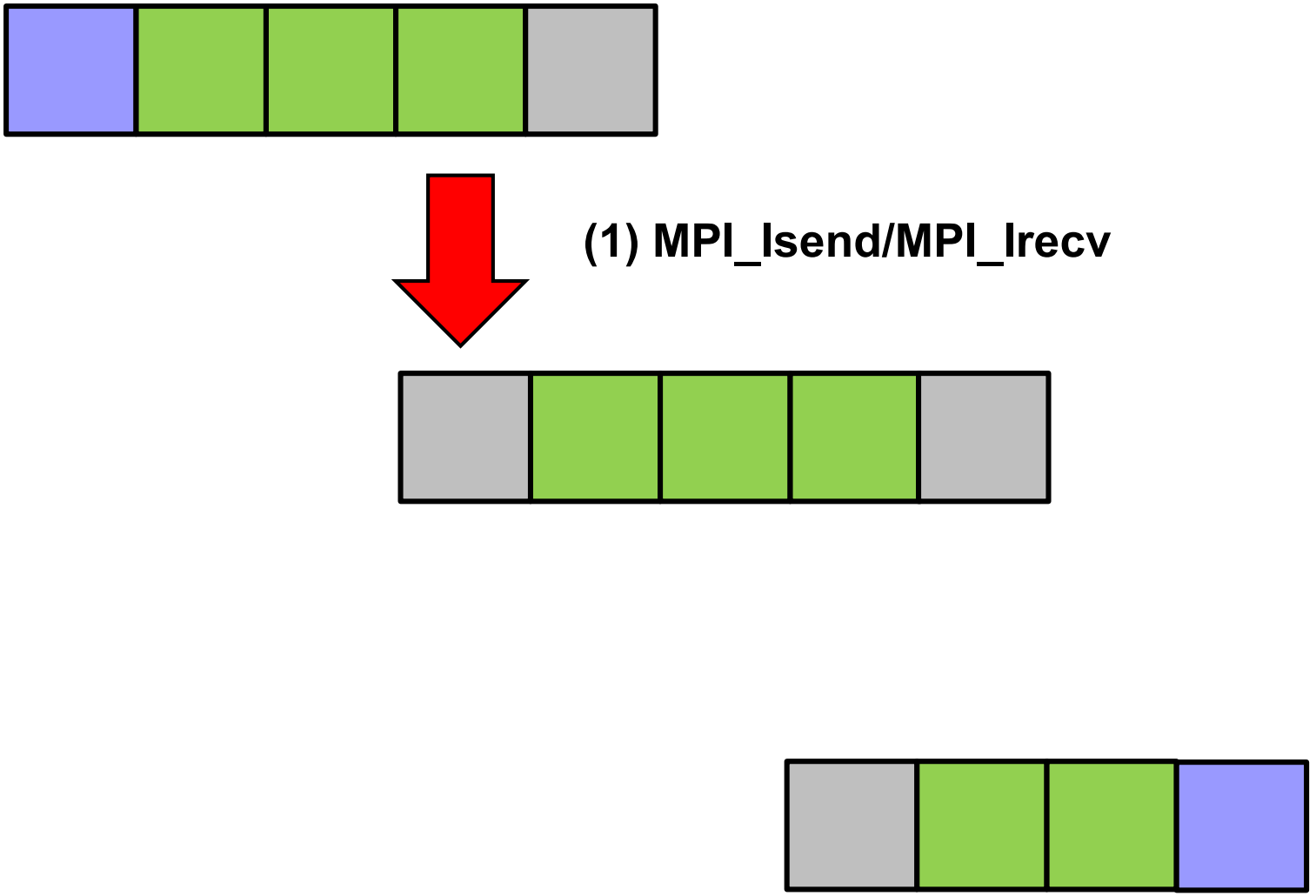
Использование классического MPI



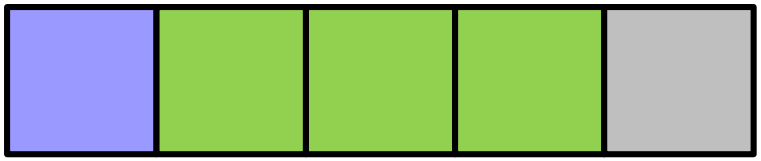
(8) MPI_Recv

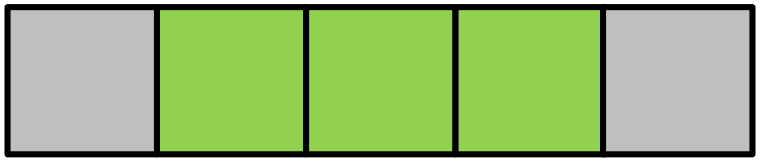


Оптимизация классического MPI

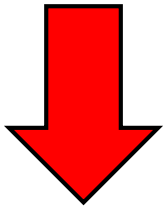


Оптимизация классического MPI

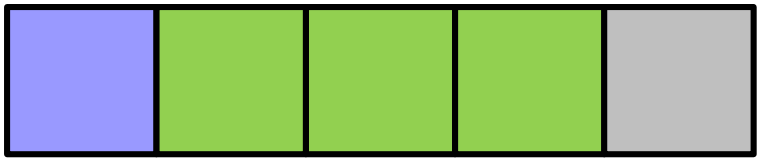


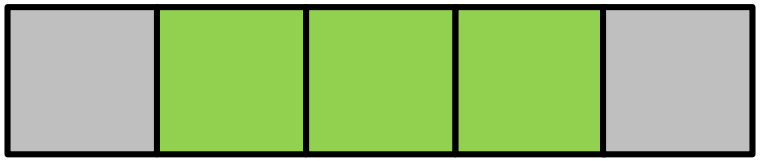


(2) MPI_Isend/MPI_Irecv

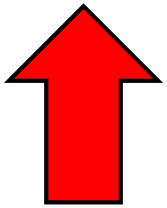


Оптимизация классического MPI

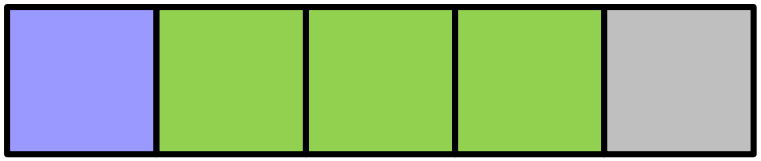




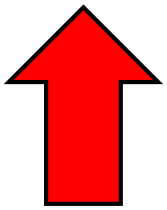
(3) MPI_Isend/MPI_Irecv

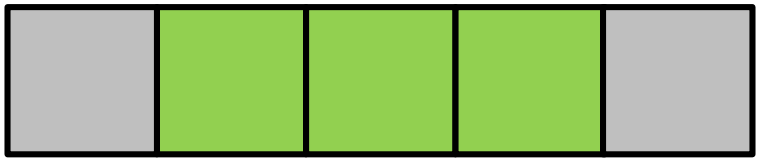


Оптимизация классического MPI

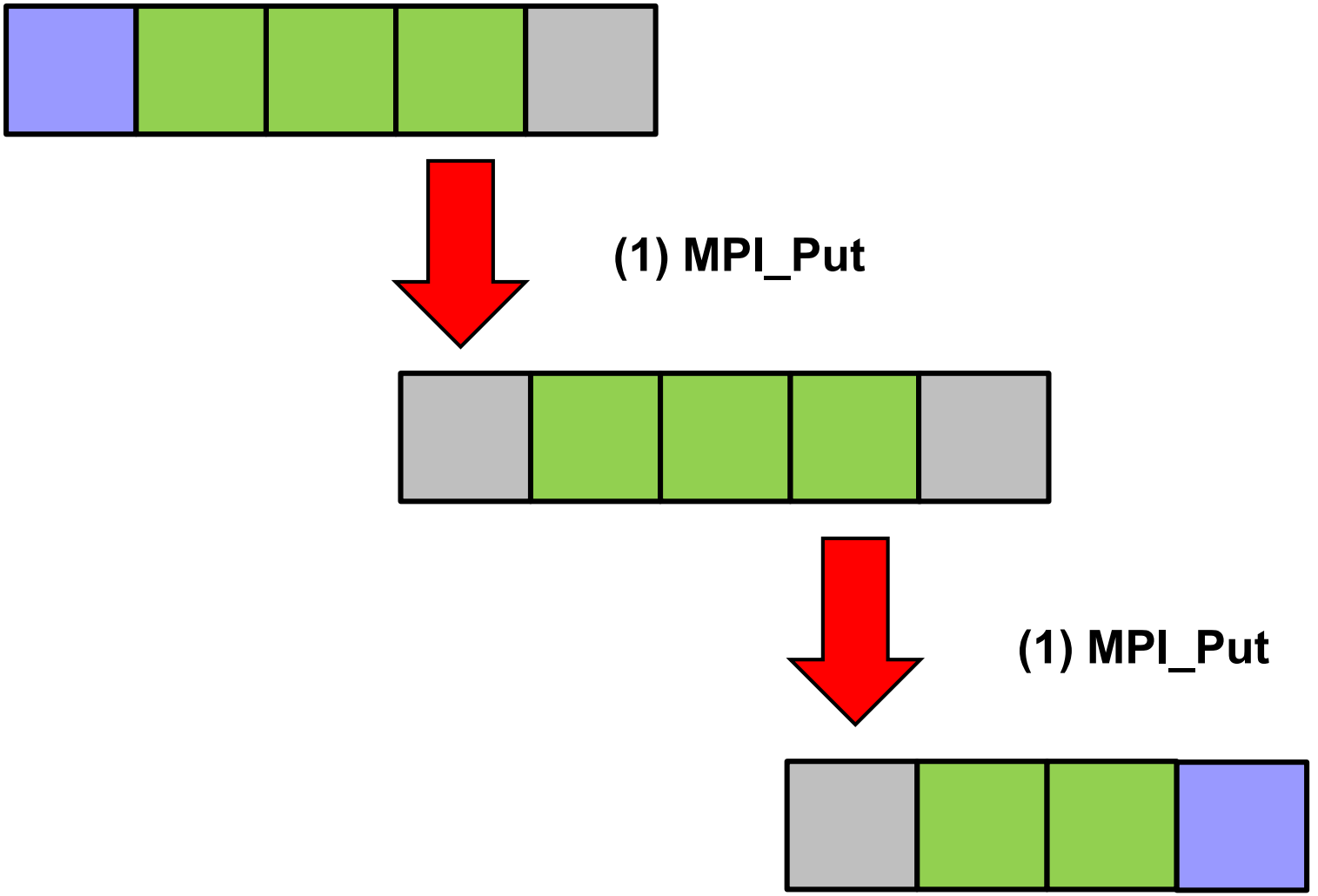


(4) MPI_Isend/MPI_Irecv

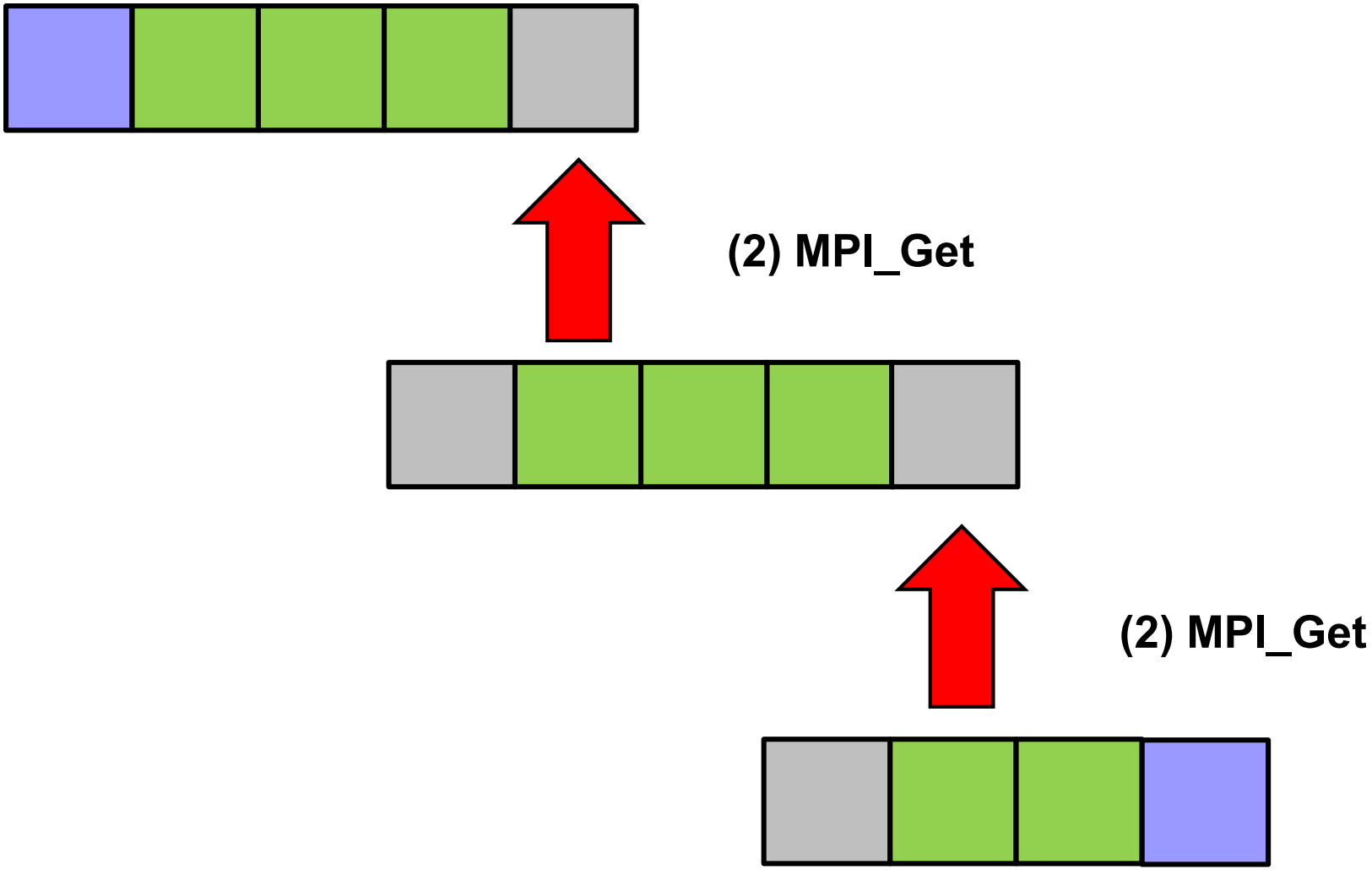




Использование MPI 3.0



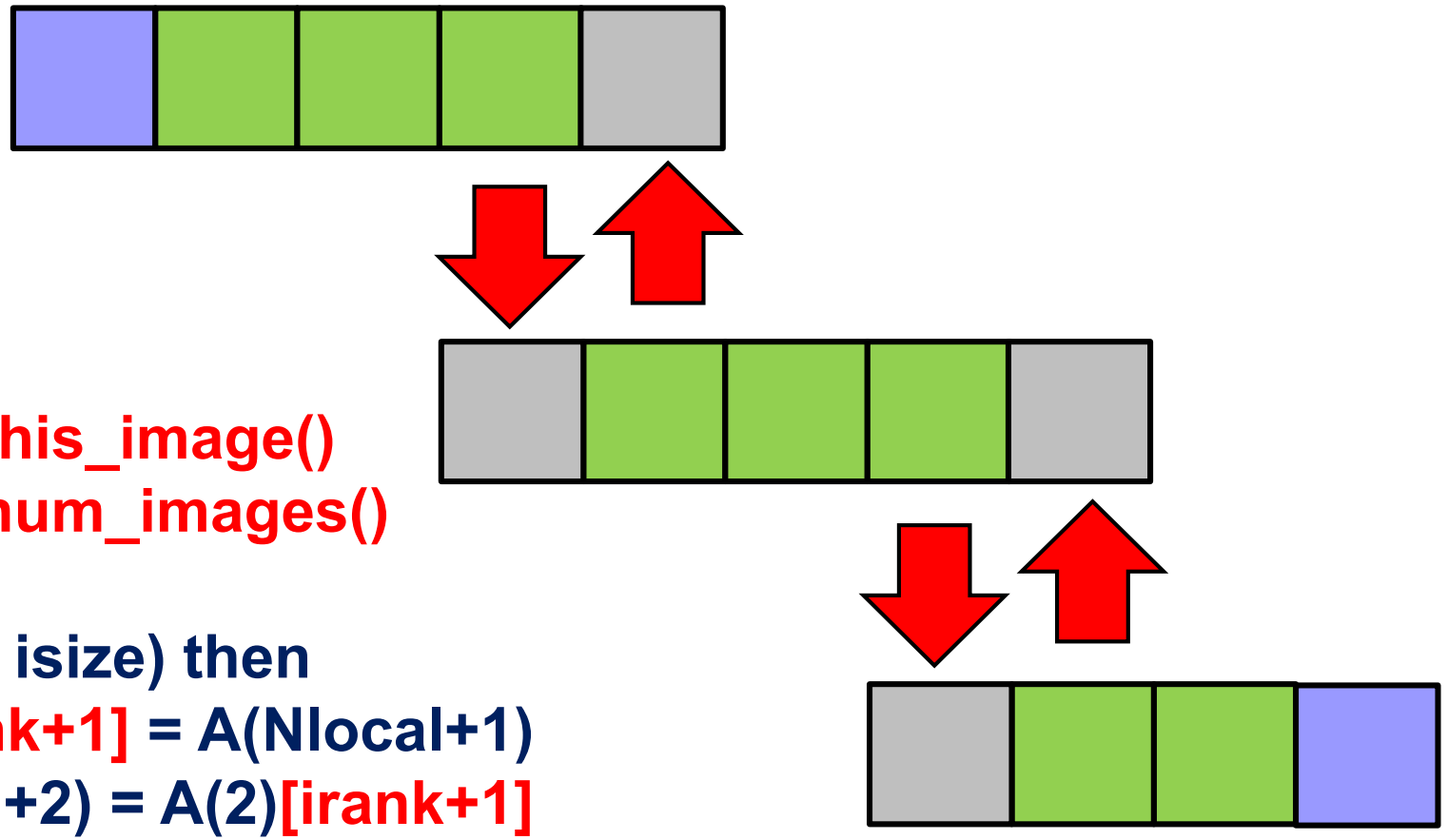
Использование MPI 3.0



Обзор использования Coarray Fortran

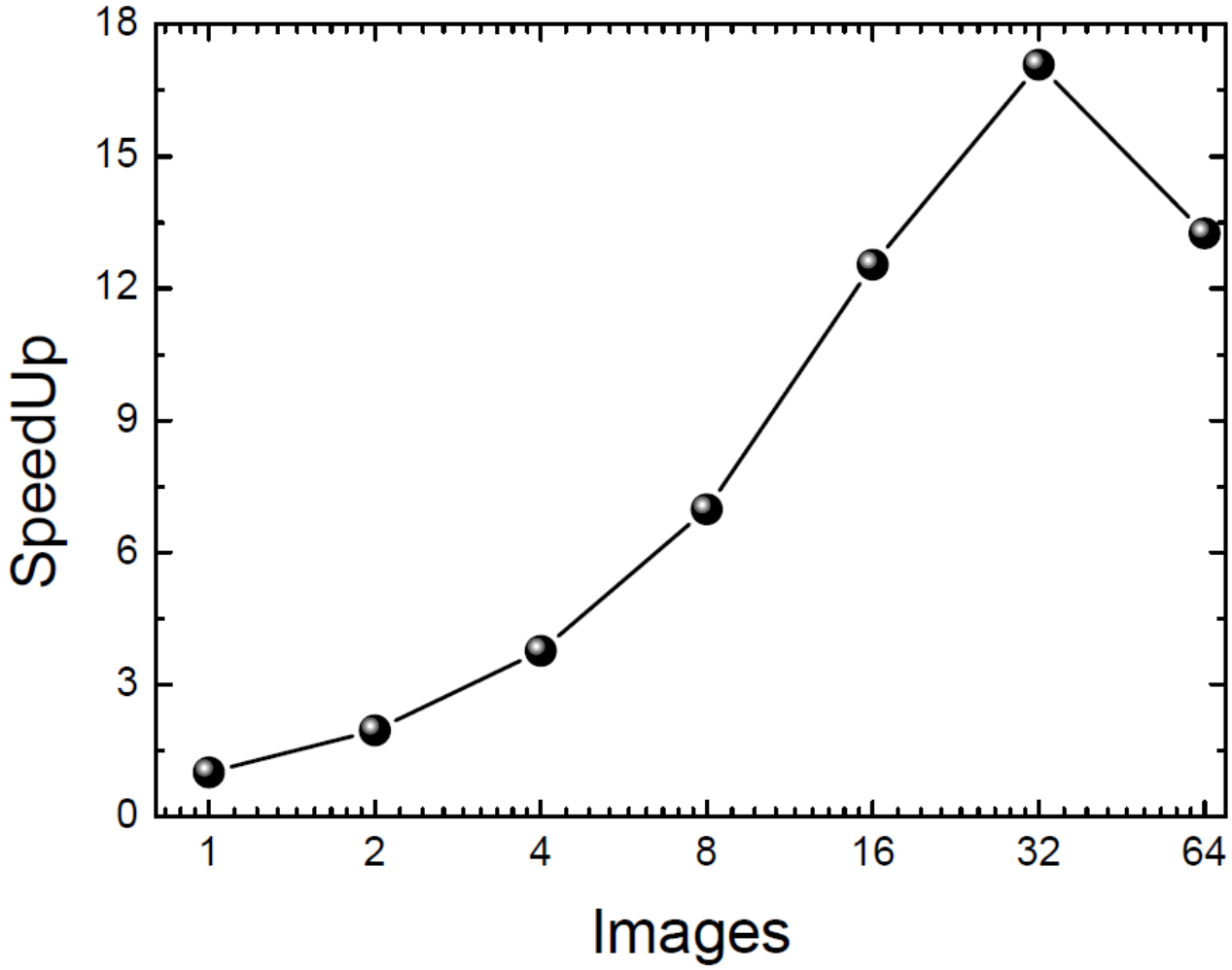
- **Reshetova, Cheverda & Khachkova, LNCS 2019**
  - **CAF превосходит производительность MPI и MPI+OpenMP при большом числе процессов**
- **Reshetova, Cheverda & Khachkova, CCIS 2019**
  - **CAF значительно проще в использовании, чем технология MPI в различных вариантах коммуникаций**
- **Reshetova, Cheverda & Koinov, CCIS 2022**
  - **CAF превосходит по производительности функции MPI_Isend/MPI_Irecv**
- **Garain, Balsara & Reid, JCompPhys 2015**
  - **CAF эквивалентна по производительности MPI 3.0**
- **Shterenlikht & Cebamanos, Parallel Computing 2019**
  - **Масштабируемость до 100 000 процессов**

Использование Coarray Fortran

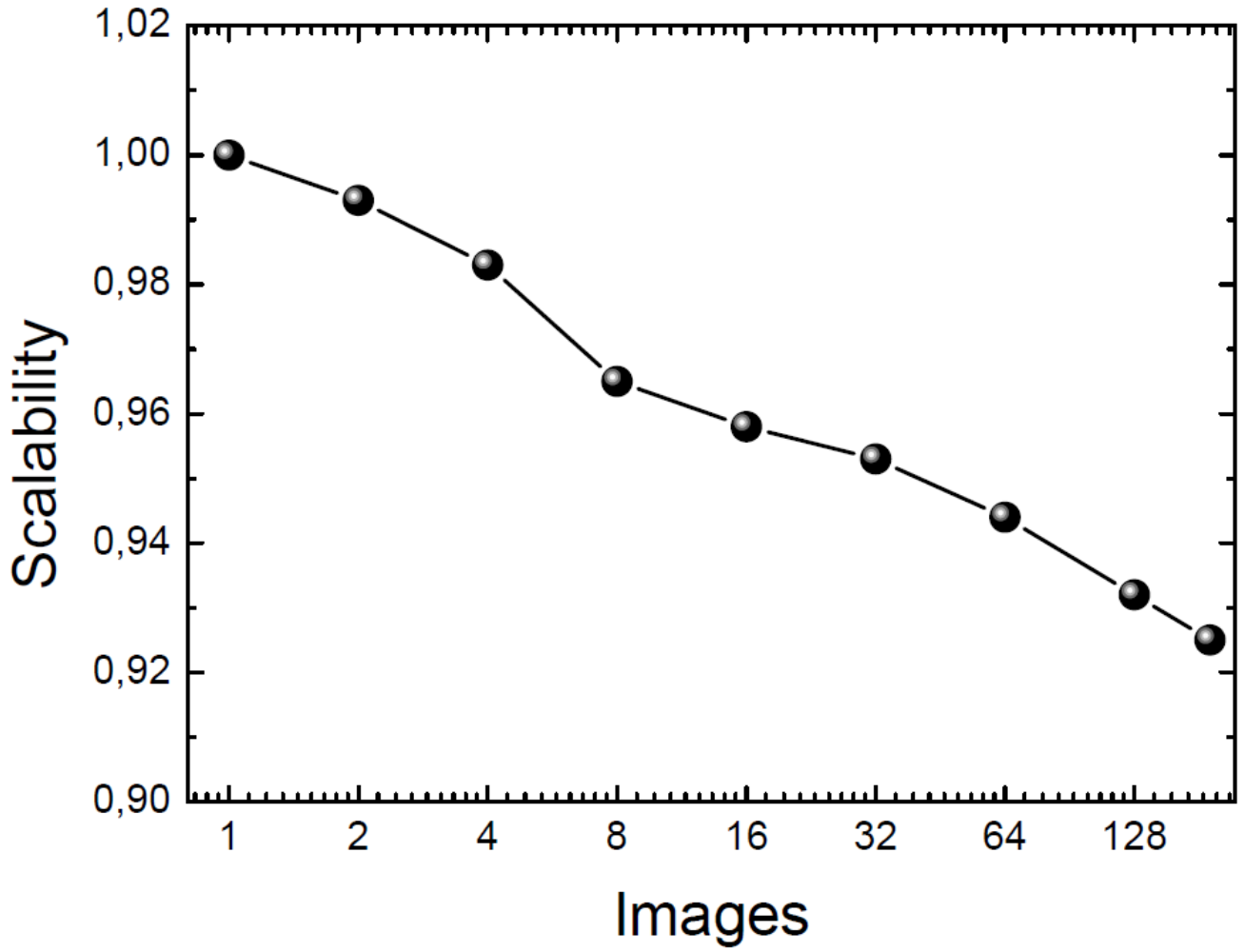


```
irank  = this_image()
isize  = num_images()

if(irank < isize) then
 A(1)[irank+1] = A(Nlocal+1)
 A(Nlocal+2) = A(2)[irank+1]
endif

sync all
```

Ускорение Coarray Fortran кода



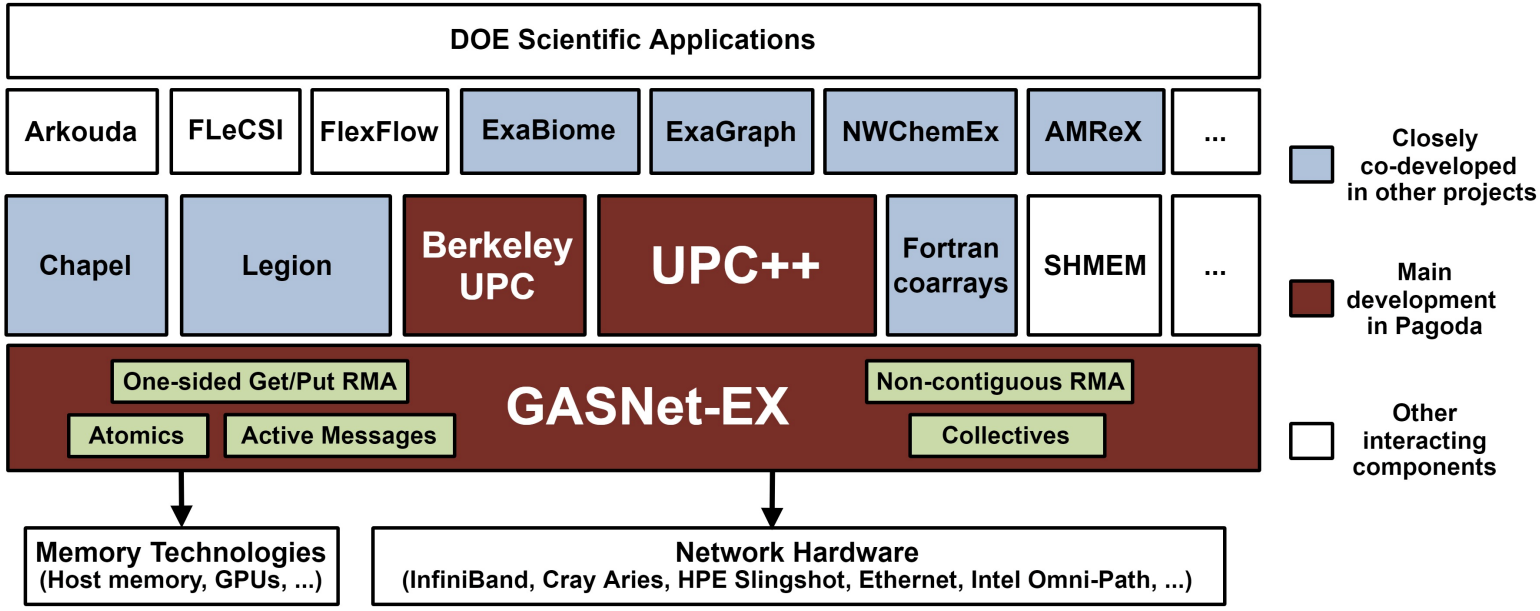
Масштабируемость Coarray Fortran кода



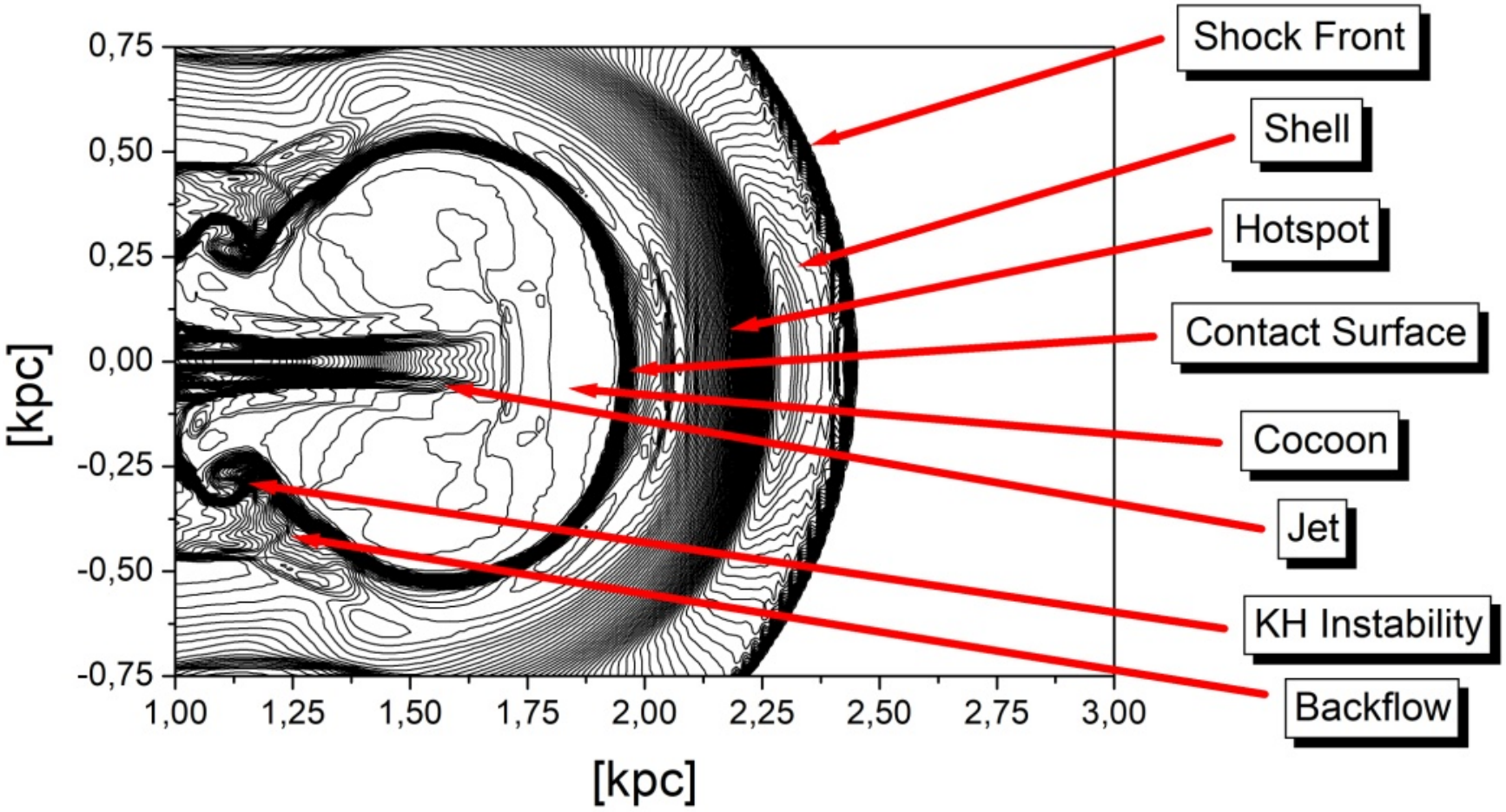
А ушли ли мы от MPI?

В компиляторе Intel используются технологии MPI 3.0 для реализации Coarray Fortran операций, но в части компиляторов GNU Fortran и OpenUH используется технология GASNet

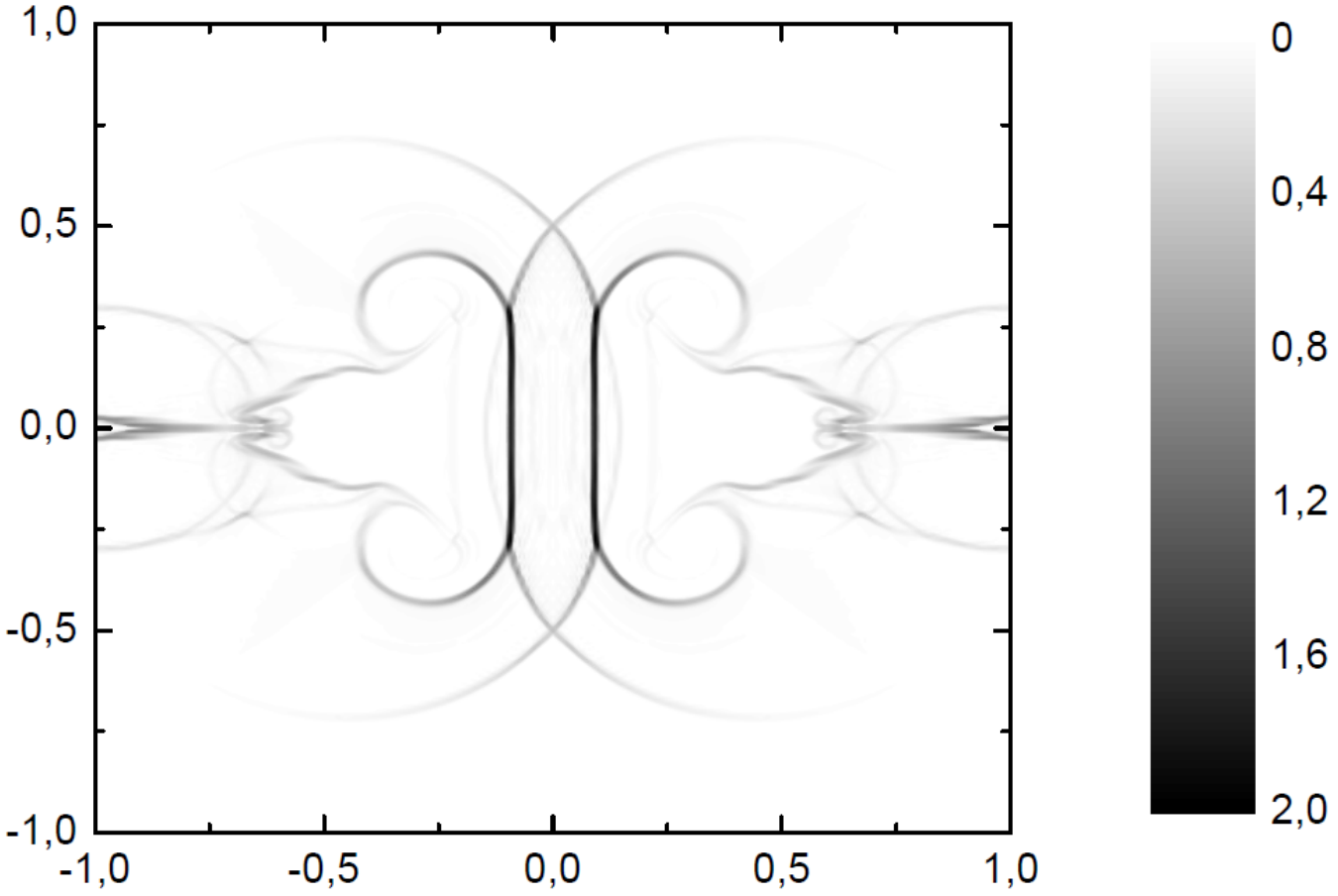
Технология GASNet как альтернатива MPI



Эволюция релятивистского джета

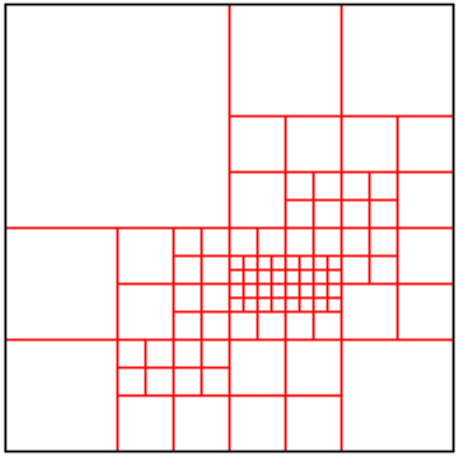


Взаимодействие релятивистских джетов

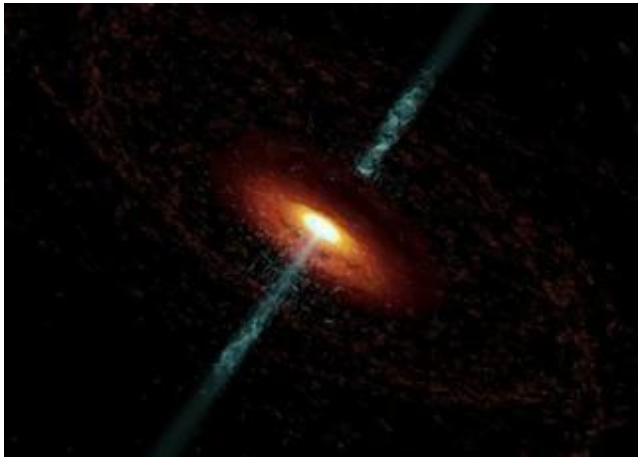


Перспектива 2025 года

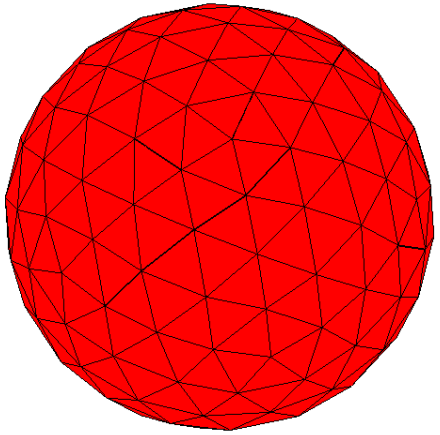
Адаптивные сетки



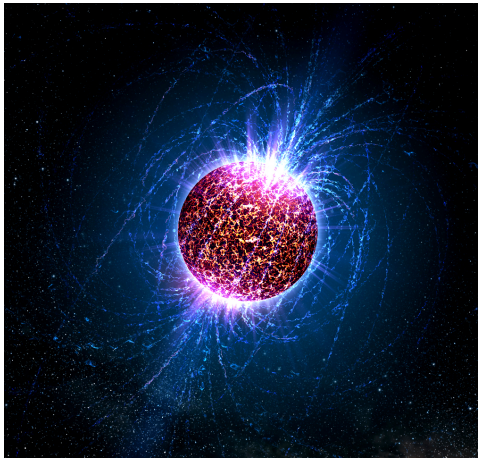
Релятивистские джеты



Геодезические сетки



Нейтронные звезды



Публикации

1. **Kulikov I., Chernykh I., Tutukov A.** A new hydrodynamic code with explicit vectorization instructions optimizations, dedicated to the numerical simulation of astrophysical gas flow. I. Numerical method, tests and model problems // **The Astrophysical Journal Supplement Series. – 2019. – V. 243. – Article Number 4.**

2. **Kulikov I.** A new code for the numerical simulation of relativistic flows on supercomputers by means of a low-dissipation scheme // **Computer Physics Communications. – 2020. – V. 257. – Article Number 107532.**

3. **Chernykh I., Kulikov I., Tutukov A.** Hydrogen–helium chemical and nuclear galaxy collision: Hydrodynamic simulations on AVX-512 supercomputers // **Journal of Computational and Applied Mathematics. – 2021. – V. 391. – Article Number 113395.**

Публикация по Coarray Fortran

A new special relativistic hydrodynamics parallel code based on simple piecewise-parabolic on local stencil method for numerical modeling of colliding jets

Igor Kulikov, Igor Chernykh, Dmitry Karavaev, Vladimir Prigarin, Anna Sapetina, Ivan Ulyanichev, Oleg Zavyalov

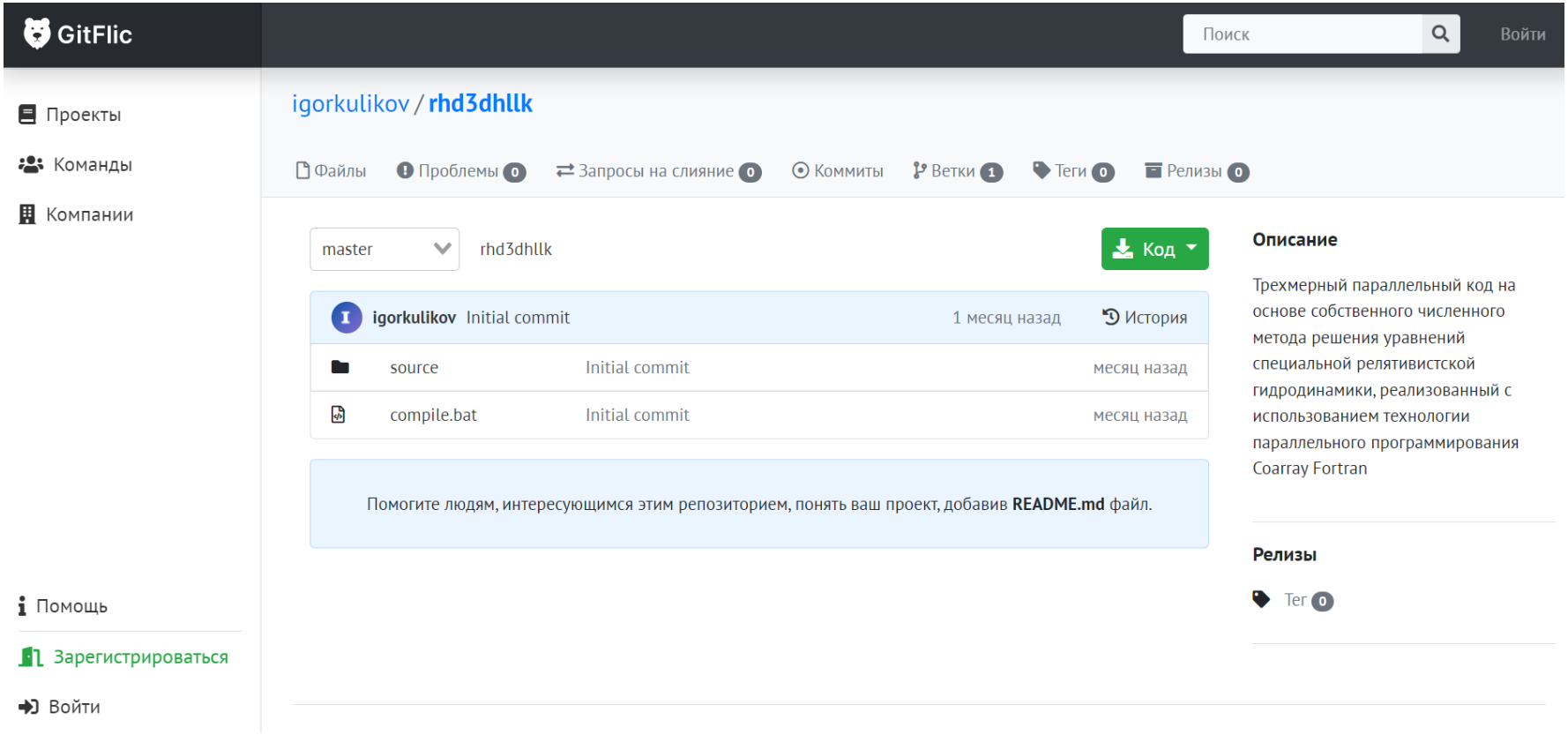
April 20, 2022

Abstract

The new parallel code for describing the interacting jets based on models of special relativistic hydrodynamics is presented in the paper. We present the new numerical method of high accuracy and results of its verification. The parallel implementation of the method by means Coarray Fortran technology and the study of the method effectiveness are described in details. The code scalability is 92 % on a cluster with Intel Xeon 6248R NKS-1P, with 192 coarray fortran images. As an astrophysical application we have considered the problem of two relativistic jets interaction.

Исходный код

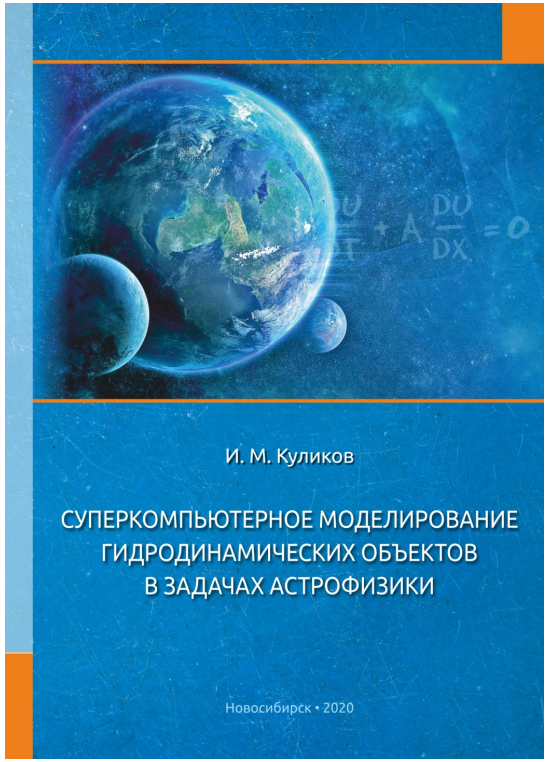
https://gitflic.ru/project/igorkulikov/rhd3dhllk



Дискуссия и заключение

- Нужно ли отказываться от библиотек? Конечно нет! Их нужно использовать там, где это возможно
- Библиотека как «объект-инструмент-объект» исследования
- Требование к новым кодам «language only»
- Уход от решения СЛАУ к операторным представлениям и тензорным разложениям с гарантированной точностью
- Переход к технологиям PGAS с поддержкой операций Remote Memory Access (RMA) и Remote Procedure Call (RPC)

Спецкурс «Высокопроизводительные вычисления на языке Фортран». Весна 2023 года



- Базовые конструкции и работа с файлами
- Функции, подпрограммы, модули
- Операции с массивами
- Использование OpenMP
- Использование CUDA
- Использование MPI
- Использование Intel oneAPI
- Средства Coarray Fortran

Спасибо за внимание!