Методы ускорения программной реализации переноса пассивной примеси на графических ускорителях

Гащук Е.М.[1,2,3], Дебольский А.В.[3,4], Мортиков Е.В.[3,2]

[1]Мехмат МГУ имени М.В. Ломоносова, Москва
[2]Институт Вычислительной математики им. Г.И. Марчука РАН, Москва
[3]НИВЦ МГУ имени М.В. Ломоносова, Москва
[4]Институт физики атмосферы им. А.М. Обухова РАН, Москва

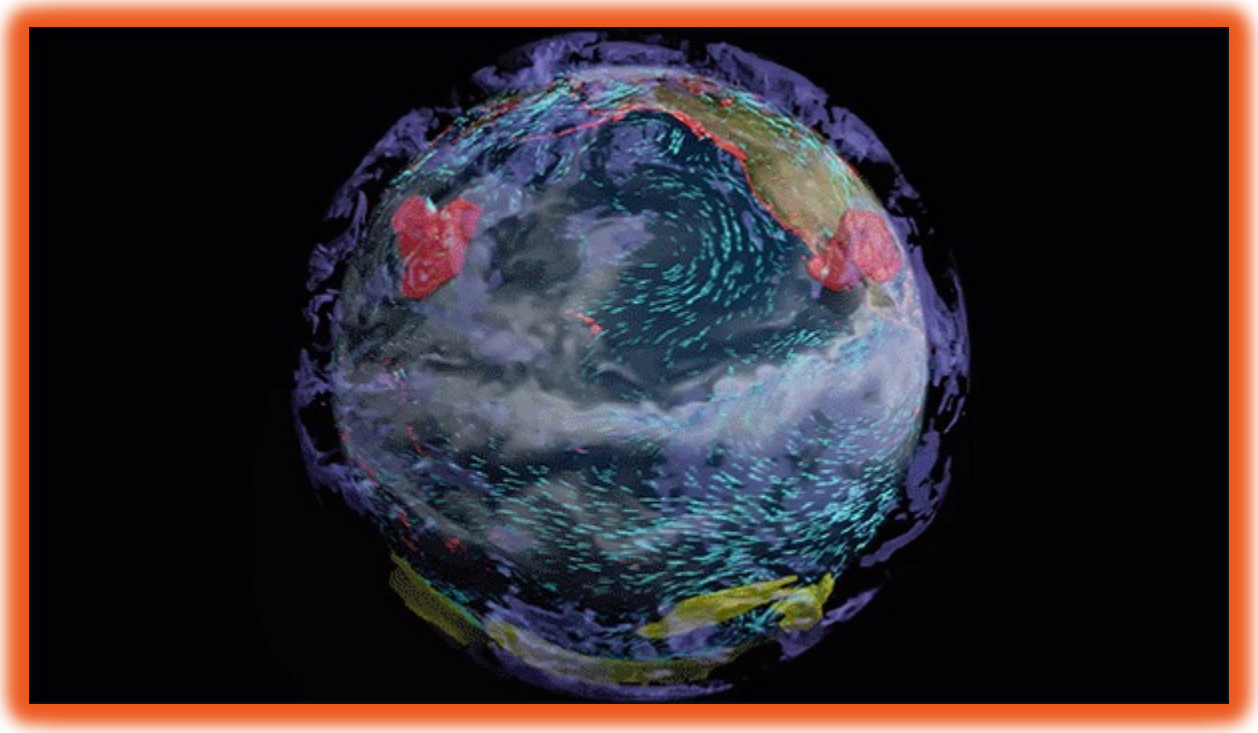
Зачем?

Расчеты для задач вычислительной гидродинамики, прогноза погоды и климата занимают большое количество времени.

Надо сокращать время вычислений по причине:

- сложность актуальных задач растет (ансамблевые прогнозы, моделирование большого числа примесей, например, реализация распространения аэрозолей и биохимических веществ)
- для актуальных задач необходимо воспроизводить численные эксперименты на сетках большого разрешения

Необходимо разрабатывать вычислительно-эффективные алгоритмы, учитывающие архитектуру суперкомпьютеров



Источник: Gadhia, Bhoomi, et al. Physics-Informed Machine Learning Platform NVIDIA Modulus Is Now Open Source, 23 Feb. 2023

Как?

Аппаратные вычисления в FP16 поддерживаются и для последних поколений CPU (например, Xeon Phi x200 и Skylake-X CPUs)

Перенос вычислений на графические ускорители и адаптация алгоритмов под современные HPC-системы

Понижение точности вплоть до половинной (поддерживается аппаратно на современных архитектурах GPU)

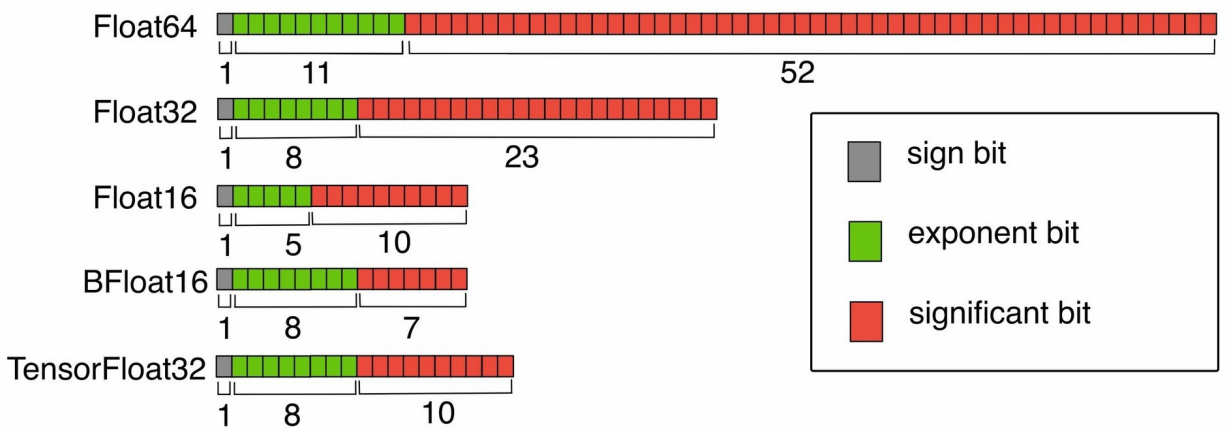


Рис. Битовое представление чисел с плавающей точкой, поддерживаемых современными архитектурами графическими ускорителей NVIDIA

- M. Chantry , T. Thornes, T. Palmer and P. Düben, 2019: Scale-selective precision for weather and climate forecasting Mon. Wea. Rev.
- A. Dawson, P.D. Duben, D.A. MacLeod et al. 2018: Reliable low precision simulations in land surface models Clim Dyn

Моделирование океана

Для чего?

- Численные модели общей циркуляции океана являются важным инструментом в исследованиях Земной системы и необходимы для уточнения нашего понимания влияния океана на погоду и климат. Они являются неотъемлемой частью современных моделей прогноза погоды и климата
- Модели океана могут быть использованы для изучения переноса загрязнений, распространения льда, помощи в поисково-спасательных работах
- Сбор данных измерений в океане и проведение натурных экспериментов, особенно в глубинных слоях, связаны с большими трудностями и высокими затратами

Перенос концентрации примеси вдоль экватора

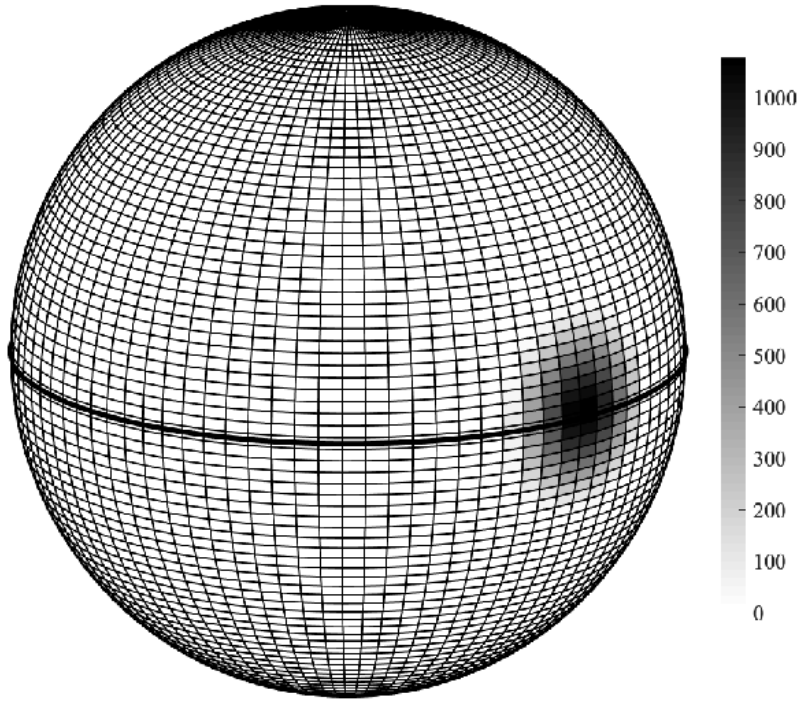


Рис. Распределение концентрации в начальном состоянии при моделировании переноса пассивной примеси на равномерной сетке в географической системе координат с горизонтальным разрешением $\Delta\lambda,\theta = 4^{\circ}$, жирная линия обозначает экватор

Проблема: для тестов на грубых сетках и ядро CPU, и узел оказались более эффективны, чем один GPU.

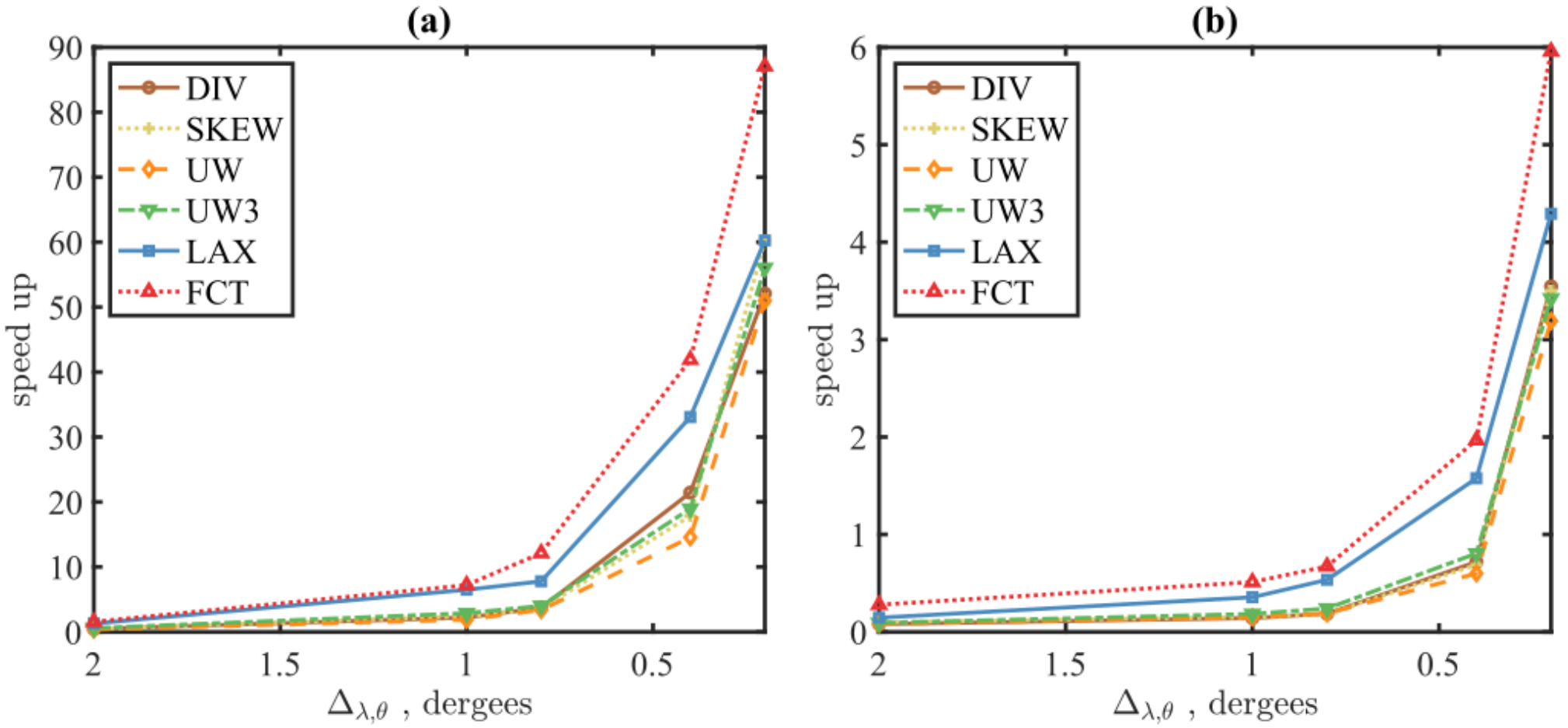


Рис. Ускорение выполнения расчетов переноса примеси на A100 GPU по отношению к CPU- ядру **(a)** и по отношению к CPU-узлу **(b)** в зависимости от схемы и горизонтального разрешения для двумерной модели.

Для выполнения расчетов были использованы вычислительные ресурсы СКЦ МГУ, ЦКП «Центр данных ДВО РАН»

Kernel fusion

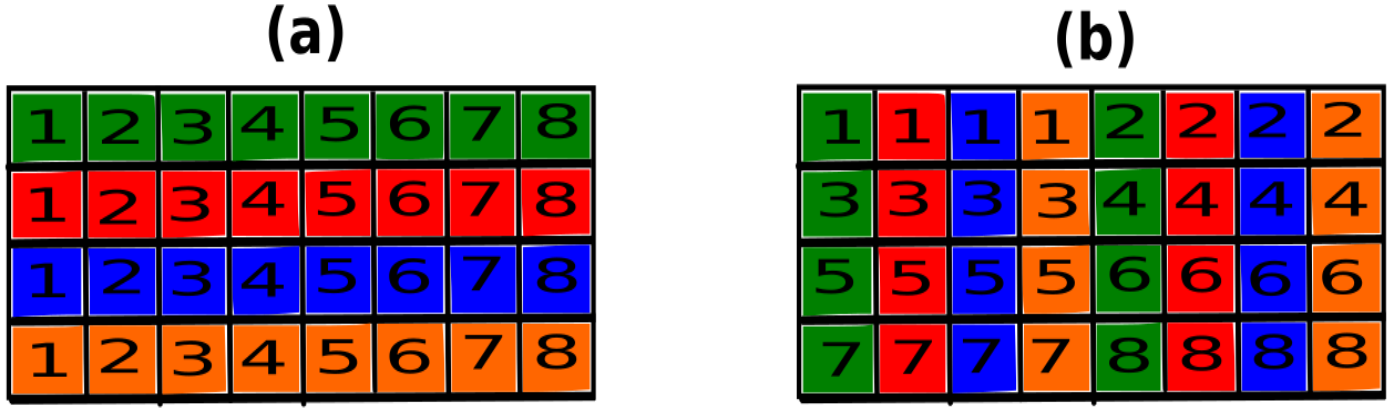
Kernel fusion

Если между вызовами последовательности ядер нет точек синхронизации, то можно объединить последовательность в одно ядро, избегая многократных вызовов

1. $$(\mathrm{ADV})_k^n = [\nabla \cdot (\mathbf{u} C_k)]_h^n$$
2. $$(\mathrm{RHS})_k^n = \frac{23}{12}(\mathrm{ADV})_k^n - \frac{4}{3}(\mathrm{ADV})_k^{n-1} + \frac{5}{12}(\mathrm{ADV})_k^{n-2}$$
3. $$C_k^{n+1} = C_k^n + (\mathrm{RHS})_k^n \cdot \Delta t$$

В таком виде алгоритм состоит из расчета адвекции **1**, интегрирования по времени **2** и обновления значений концентрации C_k для k-го вещества **3** на следующем (n + 1) шаге по времени. Мы объединили шаги **1** и **2** в одно ядро в случае схем **DIV**, **SKEW** и **UW3**.

Scalar fusion



Scalar fusion

При рассмотрении N_c примесей вычисления могут быть объединены для улучшения эффективности обращения к памяти. В результате каждый шаг интегрирования по времени выполняется для всего набора примесей вместо N_c отдельных вычислений для каждой концентрации C_k

Рис. Размещение памяти примесей в *Φ:* прямой порядок **(a)** и с переупорядочиванием **(b)**, число соответствует индексу ячейки вычислительной сетки, цвет - концентрации C_k

Для реализации такого подхода мы объединили массивы, используемые для хранения значений концентрации C_k в единый вектор состояния полной системы *Φ*. Расположение данных концентрации в *Φ* может быть различным. Мы реализовали этот подход двумя способами в зависимости от размещении памяти k-ой примеси в *Φ* :

1. прямой порядок: $C_k[m] = \Phi[k \cdot N_g + m]$,
2. с переупорядочиванием: $C_k[m] = \Phi[m \cdot N_c + k]$,

где порядок задан для k-го вещества, $k = 1, \ldots, N_c$, и m-го индекса ячейки вычислительной сетки (сетка состоит из $N_g = N_x \cdot N_y$ ячеек), $m = 1, \ldots, N_g$.

Результаты оценки производительности

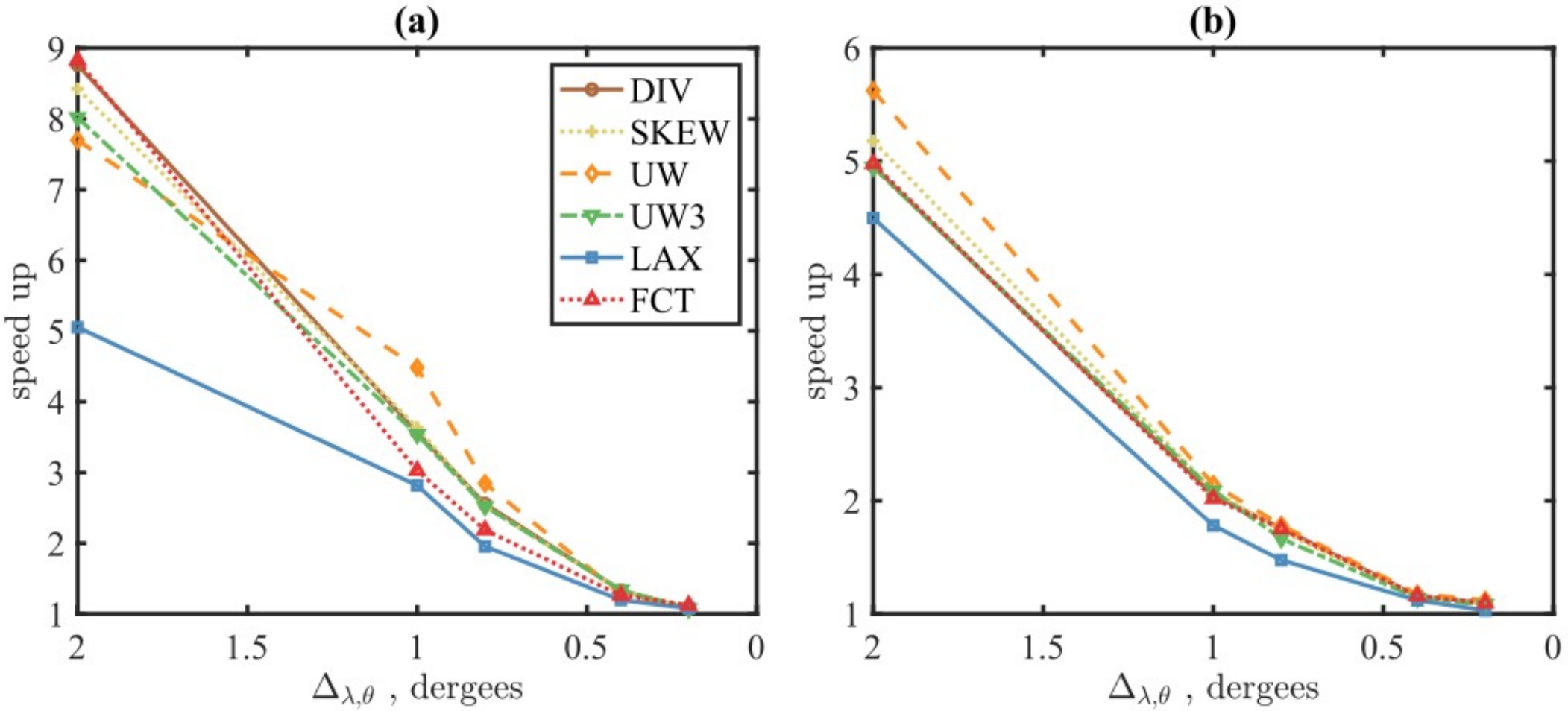


Рис. Ускорение метода ***scalar fusion*** *(*прямой порядок*)* по отношению к базовой реализации для двумерной модели на A100 **(a)** и V100 **(b)** GPUs

Для выполнения расчетов были использованы вычислительные ресурсы СКЦ МГУ, ЦКП «Центр данных ДВО РАН»

Результаты оценки производительности

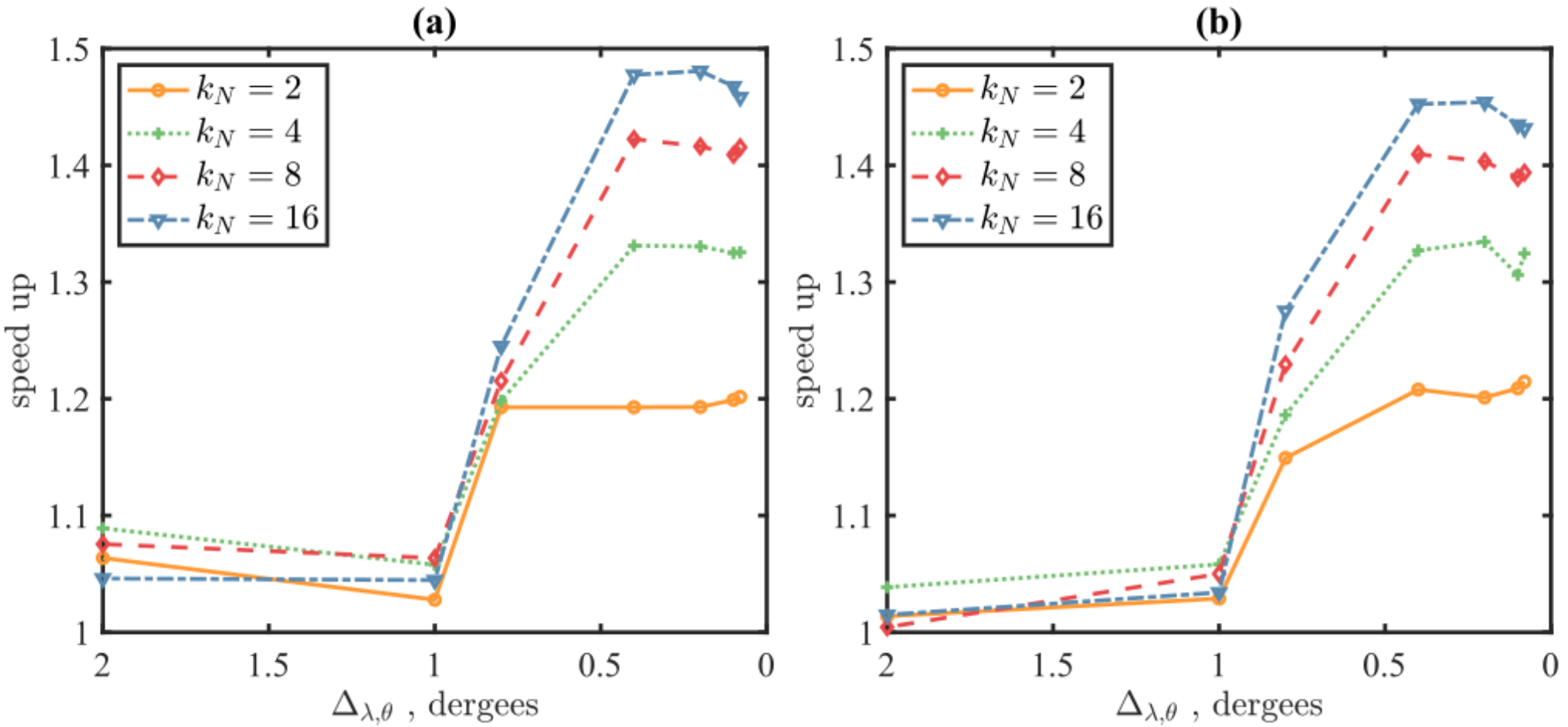


Рис. Ускорение метода ***scalar fusion*** с переупорядочиванием по отношению к ***scalar fusion*** с прямым порядком в случае двумерной модели для схем **DIV (a)** и **UW3 (b)** на V100 GPU

Для выполнения расчетов были использованы вычислительные ресурсы СКЦ МГУ, ЦКП «Центр данных ДВО РАН»

Моделирование турбулентных течений

Уравнение переноса:

$$\frac{\partial C}{\partial t} + \frac{\partial u_i C}{\partial x_i} = \frac{1}{Re \cdot Sc} \frac{\partial^2 C}{\partial x_i \partial x_i} - T^{-1} C$$

$\mathbf{x} = (x_1, x_2, x_3)^T \equiv (x, y, z)^T$ – пространственные координаты,

$\mathbf{u}(\mathbf{x}, t) = (u_1, u_2, u_3)^T \equiv (u, v, w)^T$ – вектор скоростей,

$Re = UH/\nu$ – число Рейнольдса, ν – кинематическая вязкость, C – концентрация скаляра, $Sc = \nu/\xi$ – число Шмидта, ξ – коэффициент диффузии, T – время реакции

Турбулентное течение Куэтта – течение между пластинами, движущимися в противоположных направлениях

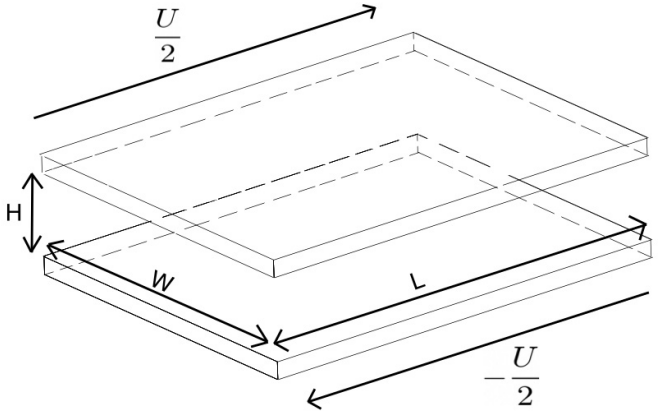


Рис. Течение Куэтта между движущимися пластинами: L, W – продольный и поперечный размеры вычислительной области, H – высота канала, U – относительная скорость движения стенок

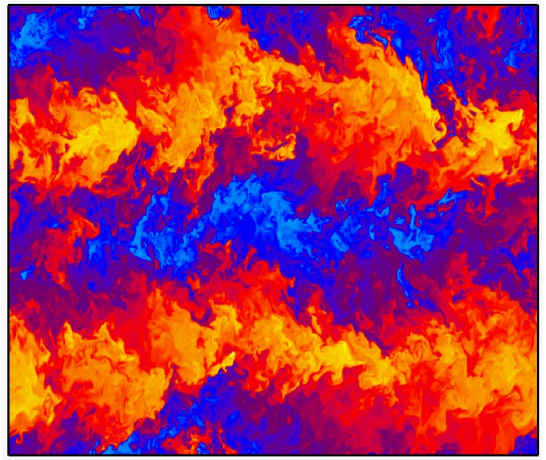


Рис. Продольная компонента скорости в течении Куэтта

Явная схема интегрирования уравнения переноса:

$$1.\ \mathrm{ADV}^n = -\Delta t \left[\frac{\partial u_i C}{\partial x_i}\right]_h^n,$$

$$2.\ \mathrm{DIFF}^n = \Delta t \left[\frac{1}{Re \cdot Sc}\frac{\partial^2 C}{\partial x_i \partial x_i},\right]_h^n,$$

$$3.\ \mathrm{RHS}^n = \tfrac{3}{2}(\mathrm{ADV}+\mathrm{DIFF})^n - \tfrac{1}{2}(\mathrm{ADV}+\mathrm{DIFF})^{n-1} - \left[\Delta t \cdot T^{-1} C\right]_h^n,$$

$$4.\ C^{n+1} = C^n + \mathrm{RHS}^n$$

Схемы 2-ого и 4-ого порядка точности

Компенсационное сумирование Кэхэна

Все данные и вся арифметика в половинной точности

Реализация в половинной точности

Для использования половинной точности (FP16) необходимо учитывать следующие вещи:

- для реализации арифметических операция надо использовать специальные функции, которые, например, представлены в CUDA

```
__device__  __half __hadd ( const __half a , const __half b  )
                  Performs half addition in round-to-nearest-even mode.

__device__  __half __hsub ( const __half a , const __half b  )
                  Performs half subtraction in round-to-nearest-even mode.
```

- максимальное по модулю представимое число в FP16 есть 65504, минимальное (субнормальное) положительное значение $\approx 5.96 \times 10^{-8}$ → возможное решение — нормировка используемых констант, смена порядка арифметических операций

$$ADV^n = -\Delta t \left[\frac{\partial u_i C}{\partial x_i} \right]_h^n$$ — расчет адвекции

$$4.\ C^{n+1} = C^n + \mathrm{RHS}^n$$

- при реализации в FP16 возможно возникновение стагнации → возможное решение — использование компенсационных алгоритмов

Численные результаты

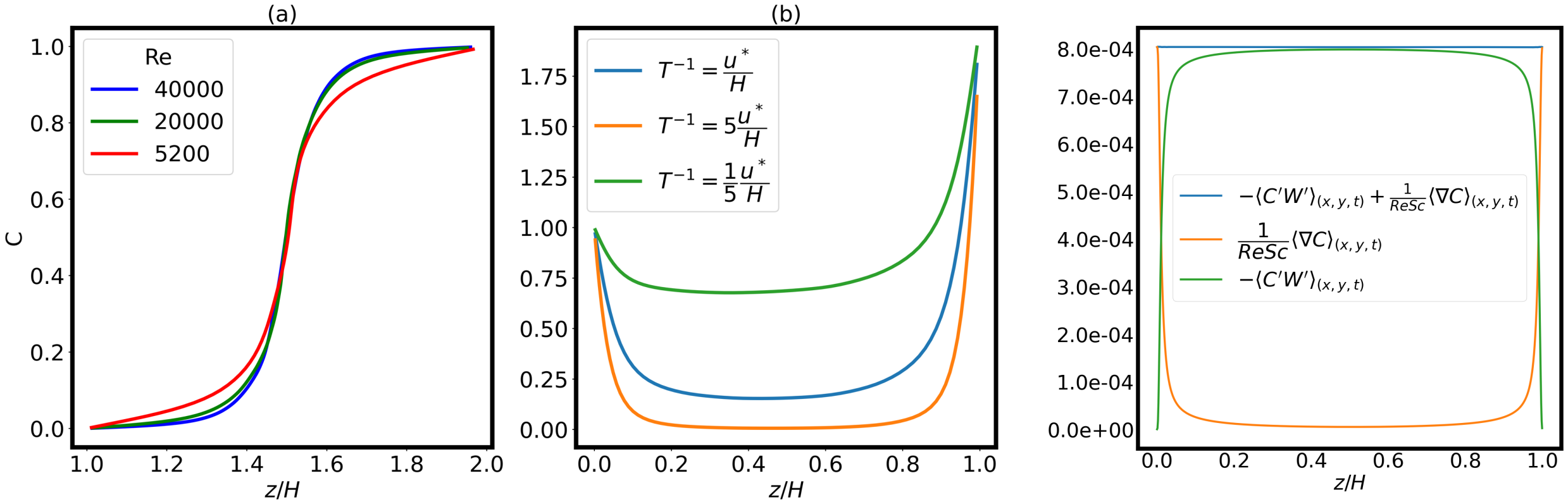


Рис. Средний профиль примеси **(a)** и средний профиль примеси при разном времени экспоненциального затухания концентрации **(b)** в FP16 и в FP32 (**порядок нормы ошибки не превышает 0.01%**)

Рис. Средний полный поток скаляра и его компоненты:
трубулентная и вязкая часть (Re = 40000)

$$\frac{\|res_{fp32} - res_{fp16}\|_{L_2}}{\|res_{fp32}\|_{L_2}} \cdot 100, \quad \|x = (x_1, \ldots, x_n)^T\|_{L_2} = \sqrt{x_1^2 + \ldots + x_n^2}$$

— норма ошибки

Выполняется также для первых, вторых и третьих статистических моментов

Результаты оценки производительности

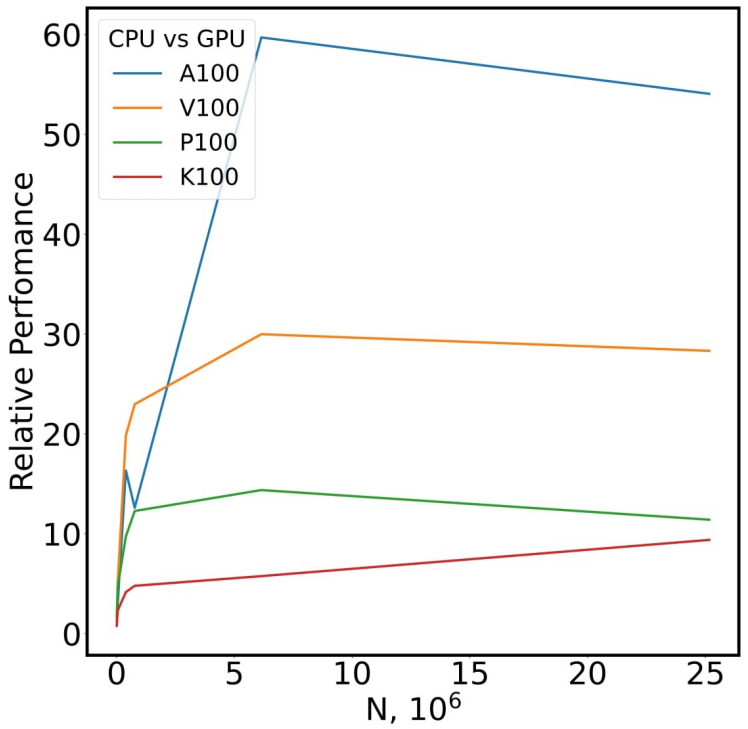


Рис. Ускорение моделирования переноса пассивной примеси на GPU к CPU в FP32 в зависимости от размера сетки (N).

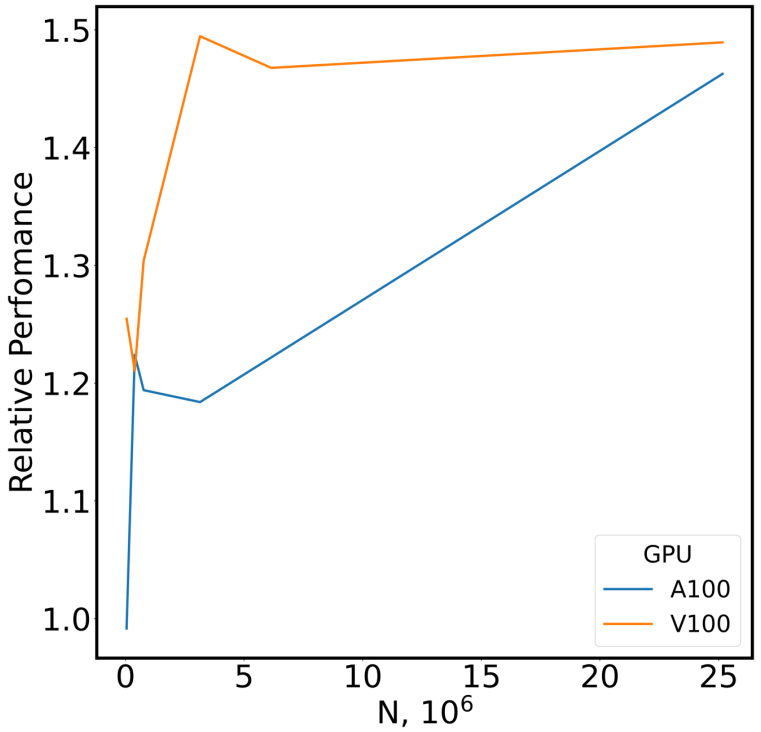


Рис. Ускорение моделирования переноса пассивной примеси на GPU в FP16 к FP32 в зависимости от размера сетки (N).

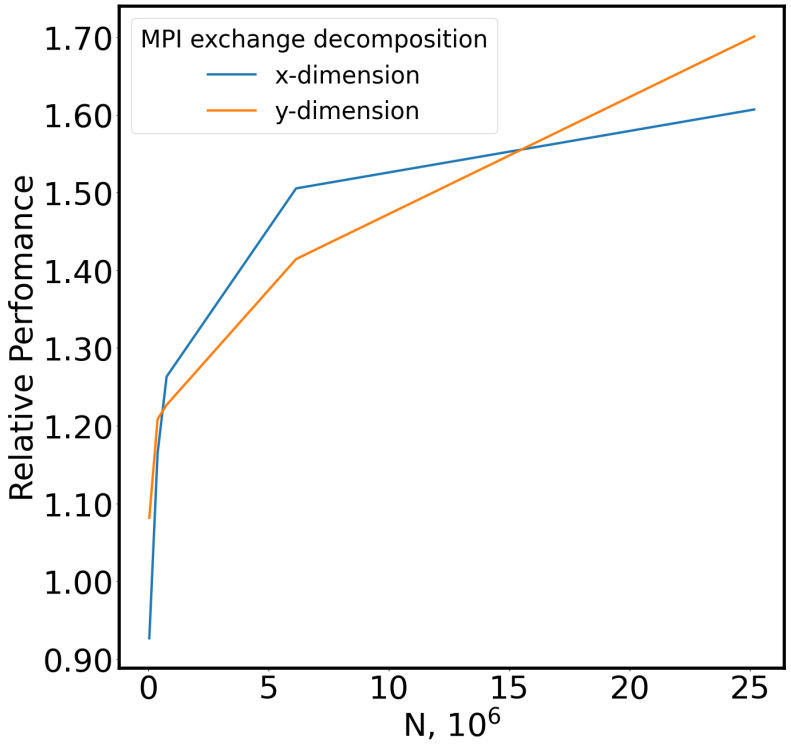


Рис. Ускорение MPI - обменов массива скаляра в FP16 к FP32 в зависимости от размера сетки (N).

Для выполнения расчетов были использованы вычислительные ресурсы СКЦ МГУ, ЦКП «Центр данных ДВО РАН»

IPC(Inter-Process Communications)
Обмен между графическими процессорами, находящимися на одном узле

Обмен данными напрямую между GPU. Результаты оценки производительности

NCCL
Обмен между графическими процессорами, находящимися на разных узлах

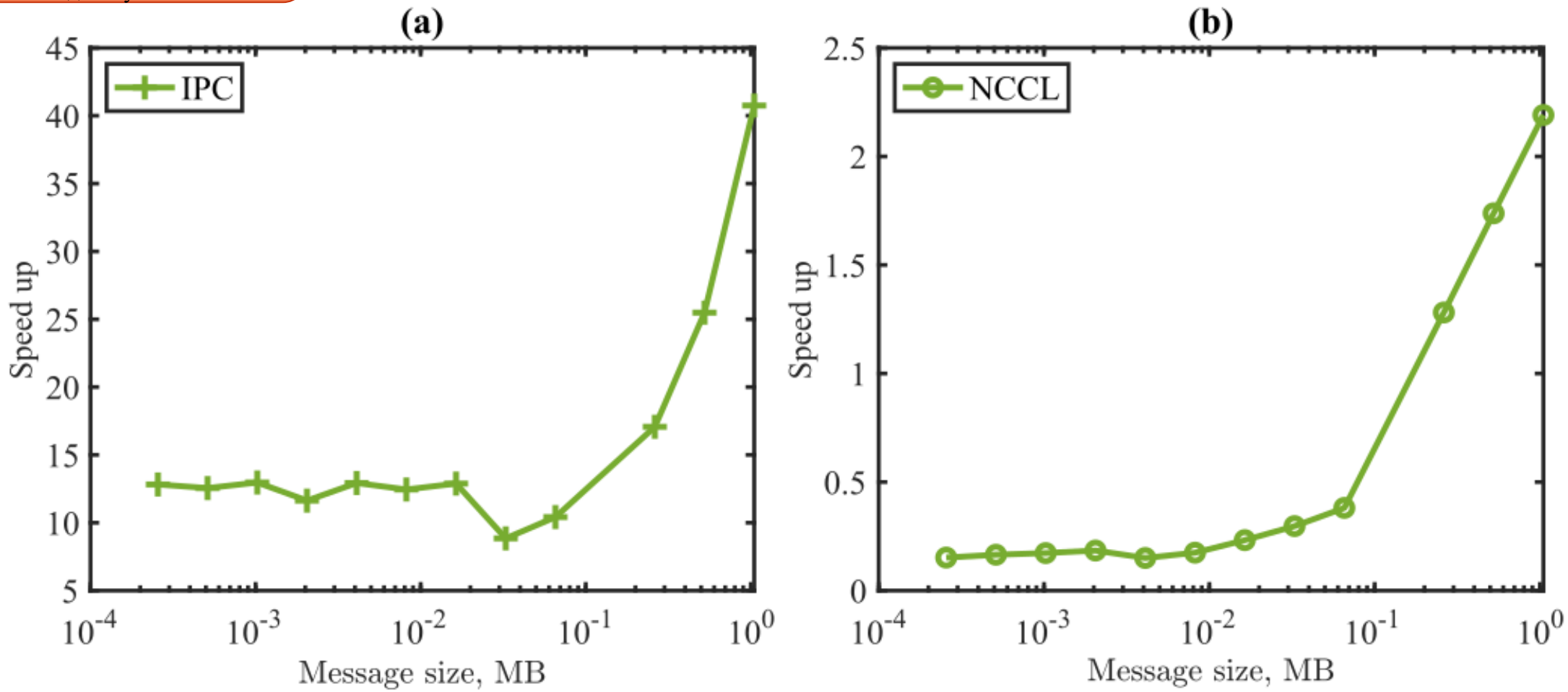


Рис. Ускорение реализации обмена с применением **IPC (a)** и **NCCL (b)** на V100 GPU в зависимости от размера пересылаемого сообщения.

Для выполнения расчетов были использованы вычислительные ресурсы СКЦ МГУ

Выводы

Ускорение переноса примеси в модели океана:

- Использование **shared memory** для реализации схем адвекции не дает значительного ускорения, а результаты применения **kernel fusion** показывают, что данный метод не всегда позволяет достичь ускорения расчетов
- **Прямой подход** техники **scalar fusion повысил эффективность** реализации **во всех** проведенных **тестах**, для двумерной модели метод обеспечил **ускорение** примерно в **5** раз на V100 и примерно в **9** раз на A100 **на самой грубой сетке**. **Scalar fusion** с переупорядочиванием обеспечил **увеличение скорости вычислений** примерно в **1.5** раза на V100 по сравнению с прямым методом для двумерного случая при проведении экспериментов на **точных сетках** для схем **DIV** и **UW3**.

Ускорение за счет FP16:

- Полная реализация блока переноса примеси в **FP16** с использованием алгоритма Кэхэна дает **достаточно точные** численные результаты
- **Ускорение** вычислений GPU - реализации в **FP16** до **1.5** раз в сравнении с исполнением в FP32
- **Уменьшение используемой памяти** в **1.5** раза в сравнении с FP32 с учетом дополнительной памяти для хранения ошибки округления в алгоритме Кэхэна
- **Ускорение** вплоть до **1.6** раз исполнения **MPI-обменов** за счет уменьшения объема передаваемых данных

Ускорение обмена между GPU: использование **IPC ускоряет обмен во всех экспериментах** (**максимальное ускорение** ≈ **40** раз для сообщений размером 1МБ). Прменение **NCCL** позволяет получить **выигрыш в скорости** проведения обмена только в случае **сообщений размером более 0.26МБ** (**максимальное ускорение** ≈ **2.5** раза для сообщений размером 1МБ)

Спасибо за внимание!