

Казахский национальный университет имени Аль-Фараби

УДК 532.517.43

На правах рукописи

ИНКАРБЕКОВ МЕДЕТ КАРКЫНБЕКОВИЧ

Высокопроизводительный 3D симулятор для моделирования турбулентных реагирующих течений методом крупных вихрей с использованием фильтрованной функции плотности

6D060300 — Механика

Диссертация на соискание ученой степени
доктора философии (PhD)

Отечественный научный консультант:
доктор физико-математических наук,
профессор Калтаев Айдархан

Зарубежный научный консультант:
профессор Питтсбургского Университета,
доктор Пейман Гиви

Республика Казахстан
Алматы, 2020

СОДЕРЖАНИЕ

НОРМАТИВНЫЕ ССЫЛКИ	3
ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	4
ВВЕДЕНИЕ	6
1 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ	17
1.1 Основные уравнения	17
1.2 Фильтрованная функция плотности	18
2 ЧИСЛЕННАЯ МОДЕЛЬ	22
2.1 Разрывный метод Галеркина	24
2.2 Схема расщепления по времени.....	26
2.3 Пространственная дискретизация основных уравнений.....	27
2.4 Решение FDF методом Монте Карло.....	31
3 ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ	35
3.1 Разработка параллельного кода DG с помощью технологии CUDA.....	39
3.2 Разработка параллельного кода MC с помощью технологии CUDA.....	43
4 ПАРАМЕТРЫ ТЕЧЕНИЯ	47
5 ВЫСОКОПРОИЗВОДИТЕЛЬНЫЙ DG-MC-LES СИМУЛЯТОР	50
5.1 Примеры применения.....	50
5.1.1 Двумерный развивающийся по времени слой смешения	51
5.1.2 Трехмерный развивающийся по времени слой смешения.....	56
5.1 Вычислительная производительность.....	64
ЗАКЛЮЧЕНИЕ	73
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	75

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей диссертации использованы ссылки на следующие стандарты:
ГОСО РК 5.04.034-2011: Государственный общеобязательный стандарт образования Республики Казахстан. Послевузовское образование. Докторантура. Основные положения (изменения от 23 августа 2012 г. №1080).

ГОСТ 7.32-2001 (изменения от 2006 г.). Отчет о научно-исследовательской работе. Структура и правила оформления.

ГОСТ 7.1-2003. Библиографическая запись. Библиографическое описание. Общие требования и правила составления.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

Обозначения

x и y, z	- продольная и поперечные координаты, соответственно,
i, j, k	- номера узлов по координатам x, y, z соответственно,
t	- время,
u_i	- компоненты вектора скорости,
p	- давление,
φ_α	- массовая доля компоненты α ,
S_α	- массовая скорость образования α компоненты,
$\tau(a, b)$	- подсеточная дисперсия скалярных функции a и b ,
$R(a, b)$	- разрешенная дисперсия скалярных функции a и b ,
$r(a, b)$	- полная дисперсия скалярных функции a и b ,
T_{ij}	- тензор напряжения вязкости,
Σ_{ij}	- подсеточное напряжение,
G	- фильтр функция,
Δ_G	- размер фильтра,
ν	- кинематическая вязкость
ν_t	- подсеточная вязкость,
P_L	- фильтрованная функция плотности,
δ	- дельта-функция,
$\langle S_{ij} \rangle_L$	- фильтрованный тензор деформации,
Re	- число Рейнольдса,
Da	- число Дамкелера
Ω_m	- частота смешивания,
Sc	- число Шмидта,
Sc_t	- подсеточное число Шмидта,
ε	- скорость диссипации,
η_K	- длина колмогоровского масштаба
τ_K	- время колмогоровского масштаба
ε_α	- скорость диссипации энергии подсеточных вихрей,
X_i	- лагранжевые координаты стохастической частицы,
W_i	- процесс Винер-Леви,
Δt	- шаг интегрирования по времени,
$\Delta x, \Delta y, \Delta z$	- шаг интегрирования по пространственным координатам,
Nm	- количество мод
Nq	- количество квадратурных точек
Ne	- количество элементов
FN	- мод функции

DFDX	- мод производной
STIFF_X	- поверхностный интеграл
SMX	- матрица жесткости
FLUXR	- правый граничный интеграл
FLUXL	- левый граничный интеграл

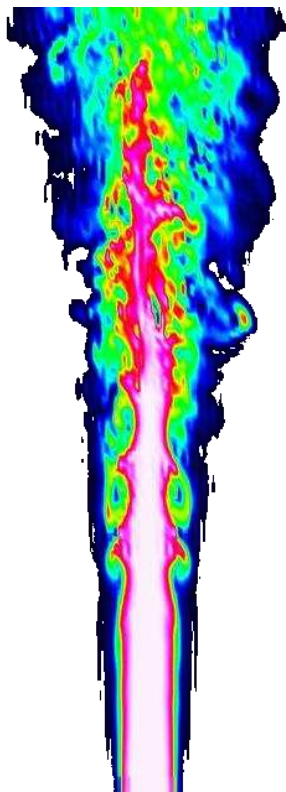
Сокращения

FDF	- фильтрованная функция плотности,
PDF	- функция плотности вероятности,
RANS	- осреднённые по Рейнольдсу уравнения Навье-Стокса,
LES	- моделирование крупных вихрей,
DNS	- прямое численное моделирование уравнения Навье-Стокса,
MC	- метод Монте-Карло,
DG	- разрывный метод Галеркин,
МКР	- метод конечных разностей,
МКО	- метод конечных объемов,
МКЭ	- метод конечных элементов,
ОДУ	- обыкновенное дифференциальное уравнение,
СДУ	- стохастическое дифференциальное уравнение,
ММН	- метод минимальных невязок.

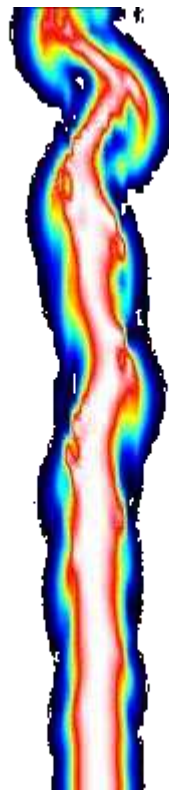
ВВЕДЕНИЕ

Актуальность работы. Один из наиболее сложных вопросов в области исследования энергетики и окружающей среды связан с точным предсказанием турбулентных реагирующих потоков. Даже во времена Осборна Рейнольдса (более века тому назад) существовала необходимость в разработке точных методов предсказания поведения химически инертных и реагирующих турбулентных потоков [1].

Один из возможных способов классификации методов расчета турбулентных течений базируется на соотношении “точно” разрешаемой и моделируемой частей энергетического спектра турбулентности. На базе этих соображений существует три основных подходов численного моделирования турбулентных течений: (I) Прямое численное моделирование (DNS), (II) осредненные по Рейнольдсу уравнения Навье-Стокса (RANS), (III) моделирование крупных вихрей (LES).



Прямое численное
моделирование
(DNS)



Моделирование крупных
вихрей
(LES)



Осреднённые по
Рейнольдсу уравнения
Навье-Стокса
(RANS)

Рисунок 1 – Основные методы численного моделирования турбулентных течений

Прямое численное моделирование (DNS) [2] является численным решением уравнений течения с физически согласованной точностью по пространству и времени. Аппроксимирующая схема должна разрешить все части энергетического

спектра турбулентности начиная с колмогоровского масштаба (т.е. описать вихрей всех масштабов). Если разрешение сетки достаточно густая и временной шаг достаточно короткий, и численный алгоритм разработан таким образом, чтобы минимизировать численную ошибку, то можно получить зависящее от времени точное трехмерное решение основных уравнений, в котором единственными погрешностями являются те, которые вводятся с помощью остаточных приближений, включенных в численную схему и в технологию представления чисел вычислительной машины.

Что касается пространственных размеров ячейки, они должны быть достаточными для разрешения Колмогоровской шкалы длин $\eta_K = (\nu^3/\epsilon)^{1/4}$, где ν и ϵ являются кинематической вязкостью и диссипацией кинетической энергии, соответственно. Шаг интегрирования по времени должен также соответствовать шкале времени Колмогорова $\tau_K = (\nu/\epsilon)^{1/2}$. Исходя из гипотезы Тейлора, можно показать, что соотношение между большими и малыми масштабами длины пропорционально связан с числом Рейнольдса $l_t/\eta_K = Re^{3/4}$, и, соответственно, объем вычислений для задач с высокими числами Рейнольдса существенно превосходят возможности современных компьютеров. Примерное требуемое время на вычислительных машинах с производительностями 200 Mflop/s и 1 Tflop/s для DNS расчетов течения в канале показано в Таблице 1 для разных чисел Рейнольдса.

Таблица 1 – Вычислительное время для канала с помощью DNS [3]

Вычислительная мощность	Число Рейнольдса Re			
	$5 \cdot 10^3$	$1 \cdot 10^4$	$1 \cdot 10^5$	$1 \cdot 10^6$
200 Mflop/s	68 дней	444 дней	610 лет	∞
1 Tflop/s	13 дней	88 дней	122 года	∞

Для примера, число Рейнольдса течения воздуха, обтекающего автомобиля со скоростью 100 км/час составляет около 10^6 , а число Рейнольдса атмосферных течений, определяющие погоду, варьируются в пределах $10^9 - 10^{11}$.

Таким образом, DNS позволяет моделировать потоки только при малом числе Рейнольдса. Кроме того, на практике очень трудно определить наименьшие масштабы до начала моделирования. Для примера, всегда можно определить кинематическую вязкость ν , но оценить скорость диссипации ϵ можно только приблизительно. Таким образом, не легко дается оценить сколько информации потеряно и, следовательно, насколько результаты отличаются от реальности.

Средние значения химических и термодинамических параметров в турбулентных потоках можно также рассчитать по усредненным уравнениям переноса, представляющим собой законы сохранения массы, импульса, энергии и вещества. Знания средних параметров потока достаточно во многих практических случаях, и нет необходимости рассчитывать мгновенные значения параметров,

которые невозможно получить при возрастании числа Рейнольдса [4]. Эти уравнения содержат члены, которые не связаны напрямую со средними параметрами и должны быть смоделированы, чтобы получить замкнутую систему уравнений. Это называется проблемой замыкания турбулентности. Оставшиеся второй и третий подходы связаны с этой проблемой.

Моделирование с помощью осредненных по Рейнольдсу уравнений Навье-Стокса (RANS) [5], включающие корреляции скорость-скорость, которые существуют даже в случае нереагирующих потоков, где основной задачей является моделирование связи между неизвестными корреляциями в этих уравнениях и осредненными переменными переноса. Такие замыкания моментов используют смесь математического анализа и физического соображения. Несмотря на это, разработка окончательной математической модели требует дополнительных предположений и эмпирических взаимосвязей.

RANS предназначен для статистического описания потока. Усреднение по времени используется чтобы уменьшить диапазон масштабов, присутствующих в турбулентных потоках. Время усреднения намного превышает самый большой временной масштаб турбулентных флуктуаций, и в результате получается уравнение сохранения, которое описывает эволюцию только средних величин потока. Величины потока, такие как скорость и давление, подразделяются на среднюю и флуктуационную составляющие на основе разложения Рейнольдса. Влияние снятых флуктуаций турбулентности на среднее течение включено в тензор напряжений Рейнольдса.

Чтобы замкнуть систему уравнений, требуется дополнительная модель для выражения напряжений Рейнольдса через средние величины потока. Широко принятый подход к моделированию напряжений Рейнольдса использует гипотезу Буссинеска [6]. Основываясь на аналогии с ламинарным потоком, он утверждает, что тензор напряжений Рейнольдса может быть связан с градиентами средней скорости через турбулентную «вихревую» вязкость. В этом подходе моделирования эффект турбулентности как добавленной вязкости широко используется для моделирования турбулентного потока. Он очень практичен в качестве инженерного метода, потому что время вычислений слабо зависит от числа Рейнольдса. Также существует ряд методов для расчета турбулентной вихревой вязкости, такие как алгебраические модели, модели с одним уравнением и модели с двумя уравнениями. Модель $k - \epsilon$ и модель $k - \omega$ являются примерами моделей с двумя уравнениями [7], и часто применяются в практических задачах.

Преимущество уравнений RANS состоит в том, что они формулируются непосредственно с учетом усредненных по времени характеристик потока, которые представляют первостепенный интерес в аэродинамических задачах. Это избавляет от необходимости рассчитывать локальные нестационарные (случайные) характеристики турбулентных течений, тем самым упрощает требование к вычислительным ресурсам.

Недостатком же является то, что все полученные RANS модели турбулентности не являются универсальными. Для каждого типа течения нужно подбирать соответствующую модель турбулентности. При этом, необходимо оценить степень достоверности результатов. Основные модели турбулентности основаны на закономерности простых "канонических" течений такие как: закон стенки, формула Колмогорова, и.т.д. Как только эти закономерности перестают выполняться, точность расчета падает. Константы в моделях турбулентности «настроены» на определенный набор потоков; каждая модель имеет свою «область применимости». Также полученные результаты недостаточно охватывают физику течения.

Третий подход, известный как моделирование крупных вихрей (LES) [8], в которой вместо того, чтобы моделировать все турбулентные величины потока, как это делается в RANS, крупные вихри решаются напрямую, а мельчайшие вихри подсеточного масштаба моделируются (Рисунок 2). Такое моделирование дает возможность решать прикладные задачи с высокой точностью, и в то же время требование к разрешению сетки гораздо меньше по сравнению с прямым численным моделированием (DNS). LES является более точным, чем подход RANS, так как большие вихри содержат большую часть турбулентной энергии и отвечают за большую часть передачи импульса и турбулентного перемешивания, и LES захватывает эти вихри во всех деталях непосредственно, в то время как они моделируются в подходе RANS. Кроме того, малые масштабы имеют тенденцию быть более изотропными и однородными, чем крупные, и, таким образом, моделирование движений SGS должно быть проще, чем моделирование всех масштабов в рамках одной модели, как в подходе RANS.

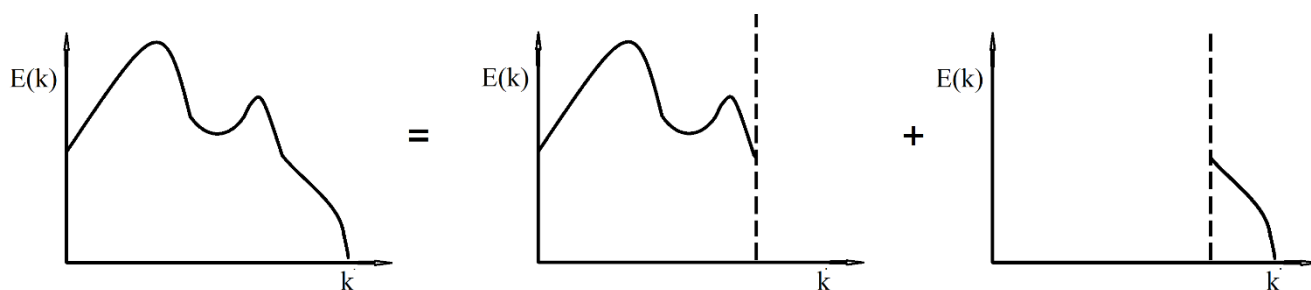


Рисунок 2 – Разделение на крупные и мелкие вихри в LES

Успешная реализация LES зависит от двух факторов: (1) насколько точно смоделированы величины подсеточного масштаба (SGS), (2) и насколько точно эти модели решены с помощью численных методов.

Методология фильтрованная функция плотности (FDF), оказалась наиболее эффективной для замыкания системы уравнений LES [9], [10]. Главное

преимущество FDF в том, что после его применения источник химических реакции в уравнении переноса скалярных переменных получается в замкнутом виде. Еще одно немаловажное преимущество в том, что замыкания величин подсеточного масштаба (например, подсеточный тензор напряжений, подсеточный поток массы, и т.д.) с помощью FDF эквивалентны SGS моделям второго порядка, в то время как большинство нынешних SGS моделей (например, замыкание Смагоринского) имеют нулевой порядок, включая динамических версий [11].

По существу, FDF является аналогом методов функции плотности вероятности (PDF), применяемых для моделирования инертных турбулентных течений. В работе [12] впервые разработано и продемонстрировано уравнение переноса FDF. Наиболее полная форма методологии, которая решает уравнение переноса для совместной статистики энергии, давления, скорости и скаляров, является EPVS-FMDF [13], которая применяется для течений с высоким числом Маха, и где химический источник зависит от давления. S-FDF и S-FMDF являются наиболее широко используемой формой методологии [14], [15], [16], [17], [18], [19], [20], [21], и рассматривают только скалярное поле несжимаемой и сжимаемой жидкости, соответственно. Гидродинамическое замыкание для несжимаемых, не реагирующих потоков было успешно достигнуто с помощью V-FDF, которая содержит статистику относительно компонент скорости [22]. Первая реализация LES горения углерода, известное как пилотируемое диффузионное пламя Sandia-D [23], была успешна проведена с помощью S-FMDF и VS-FMDF. Самой сложной формой модели на сегодняшний день для однофазного, сжимаемого, многокомпонентного течения является “Self-contained Filtered Density Function”, которая учитывает моделирование подсеточных функции всех соответствующих переменных переноса: скорость, давление, энергия, компоненты течения (PEVC-FMDF) [24]. В [25] представлен обзор последних достижений в методологии FDF, которая применяется для моделирования турбулентных потоков с помощью метода LES.

Эффективным путем решения уравнения переноса FDF является лагранжевый метод Монте-Карло (MC) [26], [27]. В этом методе физическая область дискретизируется стандартным форматом (с помощью методов конечных разностей, конечных объемов, и т.д.), и FDF представляется через ансамбль частиц. Каждая из этих частиц несет информацию, относящуюся к физическому полю (поле скалярных переменных, компонент вектора скорости и т.д.) и вектору местоположения. Также, процедура Монте-Карло должна быть комбинирована с Эйлеровым методом, который решает основные отфильтрованные уравнения течения. Комбинирование должно быть выполнено таким образом, чтобы общая точность решателя все время поддерживалась. В методе LES очень желательным фактором является то, что при увеличении разрешения сетки и/или повышении порядка точности применяемой схемы, влияние величин подсеточного масштаба уменьшалось (стремление к решению DNS). Также желательно, чтобы точность прогнозирования метода LES не зависела от размеров ячеек сетки. Разрывный

метод Галеркина (DG) в комбинации с методом Монте Карло способен удовлетворить всем этим критериям.

Первая численная схема разрывного Галеркина в комбинации с Монте Карло была разработана Sammak et al. [28] для SFDF. Такой численный код имеет ряд преимуществ. Существенным преимуществом является то, что метод сходится к решению DNS путем увеличения порядка аппроксимирующего полинома. Скорость сходимости имеет экспоненциальный характер, и делает метод намного эффективным по сравнению с традиционными методами, где применяется способ увеличения разрешения сетки. Так как переменные DG представляются с помощью простых полиномов внутри элемента, то их значения с легкостью интерполируются в местоположения частиц Монте Карло, что предотвращает потерю точности [29]. Более того, за счет высокого порядка полиномиальной аппроксимации размеры ячеек сетки являются намного крупнее чем ячейки в стандартных дискретизациях.

Такое сочетание математического и численного моделирования обеспечивает надежное прогнозирование турбулентных течений. Тем не менее, такой подход является требовательным к вычислительному ресурсу, особенно для прикладных задач, включающие химическую кинетику [30]. Причиной этому служит огромное количество частиц в метод Монте-Карло, которое имеет порядок от миллионов до миллиардов частиц. Для каждой частицы численно решается система стохастических дифференциальных уравнений. Даже при таком раскладе, требование метода LES к разрешению вычислительной сетки на несколько порядков меньше по сравнению с DNS.

Таблица 2 – Вычислительные требования для LES/FDF вычислений [31]

Количество узлов	Количество частиц	Количество химических компонент	GFLOP за один шаг по времени	Количество итерации	Время расчета (1.5 GFLOP/s)
10^6	6×10^6	9	29.5	60	13.7 дней
10^6	6×10^6	19	90.7	60	42 дней
5×10^6	50×10^6	19	544.7	220	2.5 года

Время симуляции прикладных задач с помощью предлагаемого LES/FDF модели с помощью последовательного кода, как показано в таблице 2, достигает порядка от месяцев до нескольких лет [32]. Поэтому требуется применение параллельных технологии. Параллельный код может быть адаптирован на вычислительных системах, основанных на CPU (центральное процессорное устройство) и/или на GPU (графическое устройство). Известно, что GPU имеет ряд преимуществ по сравнению с CPU [33]. Главное из них является стоимость. Если собрать вычислительную систему, основанную на GPU, то ее стоимость будет на

несколько порядков меньше стоимости эквивалентной по производительности вычислительной системы, основанной на CPU.

Степень пригодности методов DG для высокопроизводительных параллельных вычислений приводятся в работах [34], [35], [36]. Достаточное количество DG расчетов с успехом реализованы на GPU. В [37], [38] разработан численный GPU солвер для решения систем линейных трехмерных гиперболических уравнений. Результаты их разработок показали прекрасное использование памяти и впечатляющую арифметическую интенсивность. По сравнению с последовательной версией кода, построенный параллельный код достиг ускорение в 40-60 раз. Таким же успехом параллельные решатели систем нелинейных гиперболических уравнений построены в работах [39], [40]. В отличие от линейных решателей, нелинейные варианты пересчитывают интегралы, содержащие нелинейные потоки для каждого элемента на каждом временном шаге.

В нашей работе мы предлагаем разработать новый высокоэффективный симулятор для моделирования сложных турбулентных течений методом крупных вихрей. Для достижения высокой эффективности, вышеперечисленные математическая модель FDF, гибридная схема DG-МС и технология параллельных вычислений на основе графических процессоров будут включены в этот программный комплекс. Результаты работы расширить возможности FDF, что даст возможность решать более комплексные и прикладные задачи в области исследования реагирующих турбулентных течений.

Цель данной диссертационной работы заключается в создании численной методологии для моделирования крупных вихрей реагирующих турбулентных течений на основе модели фильтрованной функций плотности с использованием метода разрывного Галеркина и технологии параллельных вычислений на графических процессорах. Такой подход дает возможность надежно и эффективно решать широкий спектр турбулентных задач, включая химически реагирующие режимы.

Для достижения цели сформулированы следующие **задачи исследования**:

- Разработать основной решатель осредненных по методу моделирования крупных вихрей уравнений Навье-Стокса для несжимаемой жидкости на основе разрывного метода Галеркина.

- Разработать методику численного решения смоделированного уравнения переноса фильтрованной функции вероятности с помощью метода Монте Карло и интегрировать с основным решателем.

- Разработка и реализация параллельного алгоритма разрывного метода Галеркина с помощью технологии CUDA.

- Разработка и реализация параллельного алгоритма метода Монте Карло с помощью технологии CUDA.

- Анализ вычислительной и общей производительности разработанной численной методологии путем моделирования двумерного и трехмерного

развивающегося по времени слоя смешения как в условиях отсутствия химической реакции, так и в условиях химической реакции.

Объектом исследования является турбулентное реагирующее течение несжимаемой жидкости.

Предметом исследования является моделирование крупных вихрей развивающегося по времени слоя смешения на основе модели фильтрованной функции плотности и разрывного метода Галеркина с использованием графических процессоров.

Методы исследования: современные методы математического и численного моделирования динамики и химической кинетики медленных турбулентных течений.

Научная новизна заключается в том, что гибридная схема DG-MC для численного решения LES/FDF модели впервые разработана и реализована с помощью технологии CUDA для проведения расчетов на вычислительных системах, основанных на графических устройствах. Это существенно расширяет круг доступных для численного решения задач для исследователей.

Научные положения, выносимые на защиту:

- Методика численного решения осредненных по методу моделирования крупных вихрей уравнений Навье-Стокса на основе разрывного метода Галеркина с использованием подсеточной модели фильтрованной функции плотности;

- Разработка параллельного алгоритма разрывного метода Галеркина для решения основных уравнений потока с использованием технологии CUDA;

- Разработка параллельного алгоритма метода Монте Карло для решения уравнения переноса фильтрованной функции плотности с использованием технологии CUDA;

- Анализ производительности разработанной численной методологии на основе моделирования двумерного и трехмерного развивающегося по времени слоя смешения

Достоверность и обоснованность научных положений, выводов и результатов диссертационной работы определяется использованием фундаментальных законов сохранения при построении математических моделей; удовлетворительным согласованием смоделированных численных результатов с данными прямого численного моделирования.

Теоретическая и практическая значимость исследования. Разработанная в диссертации новая численная методология фильтрованной функции плотности для моделирования крупных вихрей реагирующих турбулентных течений на основе метода разного Галеркина с использованием технологии CUDA для параллельных вычислений может быть применена для дальнейшего численного исследования двумерных и трехмерных турбулентных реагирующих течений теоретического и прикладного характера.

Связь диссертационной работы с другими научно-исследовательскими работами. Данная работа выполнена в рамках проекта программы грантового финансирования фундаментальных исследований в области естественных наук МОН РК «Высокопроизводительный 3D симулятор фильтрованной функции плотности для моделирования крупных вихрей турбулентных реагирующих течений, основанный на методах разрывного Галеркина и Монте Карло» (2018-2020 гг., № ГР 0118РК00564) [41], [42].

Апробация работы. Основные положения и результаты диссертационной работы докладывались и обсуждались на следующих научных мероприятиях:

- APS March Meeting 2017, Session V35: General Fluid Mechanics (March 13–17, 2017, New Orleans, Louisiana, USA);

- 70th Annual Meeting of the APS Division of Fluid Dynamics, Session M30: High Performance Computing (November 19–21, 2017; Denver, Colorado, USA);

- Актуальные проблемы информатики, механики и робототехники. Цифровые технологии в машиностроении (4-5 октября, 2018, Алматы);

- Seventeenth International Conference on Numerical Combustion (May 6-8, 2019 Aachen, Germany);

- International scientific conference «Inverse Problems In Finance, Economics and Life Sciences» (August 31– September 4, 2019, Almaty);

- научных семинарах лаборатории Laboratory for Computational Transport Phenomena Департамента инженерной механики Питтсбургского Университета (2017, 2018, г. Питтсбург, Пенсильвания, США);

- научных семинарах механико-математического факультета КазНУ им. аль-Фараби (2014-2019, Алматы);

- научных семинарах кафедры прикладной механики и инженерной графики КазНУ им. К.И. Сатпаева (2019, Алматы);

Публикации. По теме диссертации опубликовано 9 печатных работ, из них: 1 статья в зарубежном научном журнале, входящим в базу данных Scopus и Web of Science с импакт фактором 1.071 [43]; 3 статей в журналах, рекомендованных ККСОН МОН РК [44], [45], [46]; 3 тезисов, опубликованных в материалах зарубежных конференций [47], [48], [49]; 2 тезисов в материалах международных конференций [50], [51], проведенных в РК. Опубликованные по теме диссертации работы приводятся в списке литературы.

Структура и объем диссертации. Диссертационная работа состоит из введения, четырех разделов, заключения, списка использованных источников из 74 работ. Работа изложена на 80 страницах, содержит 29 иллюстрации, 6 таблицы.

Во введении отражены следующие моменты: актуальность, основные цели работы, новизна, научно-практическое значение диссертационной работы, степень ее разработанности.

Первый раздел посвящен математической модели течения и содержит в себе два подраздела. В первом подразделе выводятся основные фильтрованные

уравнения течения на основе метода моделирования крупных вихрей и описывается способ замыкания этих уравнений с помощью метода Смагоринского. Во втором подразделе описываются способ замыкания химических источников с использованием смоделированного уравнения переноса фильтрованной функции плотности.

Второй раздел посвящен численной модели течения и содержит в себе четыре подразделов. В первом подразделе дается общая формулировка разрывного метода Галеркина для уравнения сохранения в слабой форме с использованием базисных функции Лежандра. Во втором и третьем подразделах представлены временная дискретизация основных уравнения с помощью метода Хорина и пространственная дискретизация основных уравнений течения с помощью разрывного метода Галеркина, соответственно. В четвертом подразделе отражен численный алгоритм решения уравнения переноса фильтрованной функции плотности с помощью лагранжевого метода Монте Карло.

Третий раздел посвящен параллельной реализации численной методологии с помощью графических процессоров и содержит в себе два подразделов. В первом подразделе разработан параллельный алгоритм разрывного метода Галеркина с использованием технологии CUDA. Здесь детально описывается оптимальное использование каждого типа памяти и определяется наиболее соответствующий размер блока потоков. Во втором подразделе разработан параллельный алгоритм метода Монте Карло с использованием технологии CUDA. Здесь отражается детальная реализация каждой из пяти стадии, включающее выбора алгоритма генерации случайных чисел, распределение потоков в блоке и оптимальное использование памяти устройства.

В четвертом разделе представлены параметры течения. В качестве тестовой задачи выбран развивающегося по времени слой смешения как без учета химической реакции, так и с учетом химической реакции. Задача смоделирована в двумерном и трехмерном пространствах. Даны физические параметры, которые включают число Рейнольдса и число Дамкелера; и параметры численных расчетов, включающие разрешение сетки и порядок аппроксимирующего полинома.

Пятый раздел отражает результаты расчетов и содержит в себе два подразделов. В первом подразделе приводятся результаты относительно прогностических возможностей, согласованности и сходимости разработанной гибридной схемы LES-FDF-DG. Согласованность и сходимость схемы оценивается путем сравнений решений первых и вторых моментов скаляров полученные с помощью FDF и DG расчетов. Полученные результаты проанализированы статистически путем вычислений осредненных по Рейнольдсу вторых моментов, разрешенных дисперсий и полных дисперсий. Приводятся результаты с учетом химической реакции для разных чисел Дамкелера. Во втором подразделе приводятся результаты относительно вычислительной производительности разработанной численной методологии. Для двумерного случая

производительность оценивается путем сравнений результатов расчетов полученные с помощью параллельного алгоритма на графическом устройстве с результатами расчетов полученные путем последовательных вычислений на центральном процессоре. Производительность решателя при трехмерных вычислениях оценивается путем определения количества операций с плавающей запятой в секунду.

В заключении приводятся основные результаты и выводы, полученные в диссертационной работе.

1 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ

1.1 Основные уравнения

Рассматривается турбулентное, изотермически реагирующее течение несжимаемой жидкости, состоящее из N_S компонентов. В математической модели описания процесса основными переменными являются вектор скорости $u_i(\mathbf{x}, t)$ ($i = 1, 2, 3$), давление $p(\mathbf{x}, t)$, и массовые доли компонентов $\phi_\alpha(\mathbf{x}, t)$ ($\alpha = 1, 2, \dots, N_S$) – скалярных функции. Система уравнений, которая описывает данный процесс имеет вид [52]:

$$\frac{\partial u_i}{\partial x_i} = 0, \quad (1)$$

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial T_{ij}}{\partial x_j}, \quad (2)$$

$$\frac{\partial \phi_\alpha}{\partial t} + \frac{\partial \phi_\alpha u_j}{\partial x_j} = -\frac{\partial J_j^\alpha}{\partial x_j} + S_\alpha \quad (3)$$

где $S_\alpha(\mathbf{x}, t) \equiv \hat{S}_\alpha(\mathbf{\Phi}(\mathbf{x}, t))$ – скорость образования компонента α в химических реакциях, и $\mathbf{\Phi} \equiv [\phi_1, \phi_2, \dots, \phi_N]$ обозначает скалярный массив. Предполагая течение ньютоновским с законом Фика для диффузии, тензор напряжения вязкости T_{ij} и массовый поток α компоненты в j -ой направлений J_j^α представляются в следующем виде

$$T_{ij} = \nu \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right), J_j^\alpha = -\Gamma \frac{\partial \phi_\alpha}{\partial x_j} \quad (4)$$

где ν – кинематическая вязкость жидкости, $\Gamma = \frac{\nu}{Sc}$ – коэффициент диффузии, и Sc – молекулярное число Шмидта.

В рамках LES используется осреднение по пространству, описываемое процедурой фильтрации [53]

$$\langle f(\mathbf{x}, t) \rangle_L = \int_{-\infty}^{+\infty} f(\mathbf{x}', t) \mathcal{G}(\mathbf{x}', \mathbf{x}) d\mathbf{x}', \quad (5)$$

где \mathcal{G} – функция фильтра, $\langle f(\mathbf{x}, t) \rangle_L$ представляет фильтрованное значение переменной переноса $f(\mathbf{x}, t)$, и $f' = f - \langle f \rangle_L$ обозначает флуктуацию f от фильтрованного значения. После применение операции фильтрации к уравнениям переноса получается

$$\frac{\partial \langle u_i \rangle_L}{\partial x_i} = 0, \quad (6)$$

$$\frac{\partial \langle u_i \rangle_L}{\partial t} + \frac{\partial \langle u_i \rangle_L \langle u_j \rangle_L}{\partial x_j} = -\frac{\partial \langle p \rangle_L}{\partial x_i} + \frac{\partial \langle T_{ij} \rangle_L}{\partial x_j} - \frac{\partial \Sigma_{ij}}{\partial x_j}, \quad (7)$$

$$\frac{\partial \langle \phi_\alpha \rangle_L}{\partial t} + \frac{\partial \langle \phi_\alpha \rangle_L \langle u_j \rangle_L}{\partial x_j} = -\frac{\partial \langle J_j^\alpha \rangle_L}{\partial x_j} - \frac{\partial M_j^\alpha}{\partial x_j} + \langle S_\alpha \rangle_L \quad (8)$$

где $\Sigma_{ij} = \langle u_i u_j \rangle_L - \langle u_i \rangle_L \langle u_j \rangle_L$ и $M_j^\alpha = \langle \phi_\alpha u_j \rangle_L - \langle \phi_\alpha \rangle_L \langle u_j \rangle_L$ обозначают подсеточное напряжение и подсеточный массовый поток, соответственно.

Стратегия замыкания. В методе LES нереагирующих течений проблема замыкания связана с величинами $\Sigma_{ij} = \langle u_i u_j \rangle_L - \langle u_i \rangle_L \langle u_j \rangle_L$ и $M_j^\alpha = \langle \phi_\alpha u_j \rangle_L - \langle \phi_\alpha \rangle_L \langle u_j \rangle_L$. В реагирующих течениях требуется дополнительная модель для $\langle S_\alpha \rangle_L$. Для моделирования подсеточного напряжения и подсеточного массового потока, используются доступные на сегодняшний день замыкания, которые хорошо известны в нереагирующих течениях. Подсеточное напряжение моделируется следующим образом [54]

$$\Sigma_{ij} - (\delta_{ij}/3)\Sigma_{kk} = -2\nu_t \langle S_{ij} \rangle_L, \quad (9)$$

где $\langle S_{ij} \rangle_L$ – фильтрованный тензор деформации и ν_t – подсеточная вязкость. Для моделирования подсеточной вязкости, используется традиционное замыкание Смагоринского [55]

$$\nu_t = C_S \Delta_G^2 \sqrt{\langle S_{ij} \rangle_L \langle S_{ij} \rangle_L}, \quad (10)$$

где Δ_G – размерава фильтра, C_S – эмпирическая константа.

Типичная модель используется для замыкания подсеточного массового потока [56]

$$M_j^\alpha = -\Gamma_t \frac{\partial \langle \phi_\alpha \rangle_L}{\partial x_j}, \quad (11)$$

где $\Gamma_t = \nu_t / Sc_t$ является подсеточный коэффициент диффузии, Sc_t – подсеточное число Шмидта.

1.2 Фильтрованная функция плотности

Ключевым моментом в этой формулировке является рассмотрение скалярных флуктуации основного массива скаляров $\Phi(\mathbf{x}, t)$ в вероятностной форме. В этом случае «Фильтрованная функция плотности» (FDF) определяется как [53]

$$P_L(\Psi; \mathbf{x}, t) \equiv \int_{-\infty}^{+\infty} \zeta[\Psi, \Phi(\mathbf{x}', t)] G(\mathbf{x}' - \mathbf{x}) d\mathbf{x}', \quad (12)$$

$$\zeta[\Psi, \Phi(\mathbf{x}, t)] = \delta[\Psi - \Phi(\mathbf{x}, t)] \equiv \prod_{\alpha=1}^{N_S} \delta[\psi_\alpha - \phi_\alpha(\mathbf{x}, t)], \quad (13)$$

где P_L обозначает FDF, δ обозначает дельта-функцию и Ψ обозначает композиционную область скалярного массива. Значение $\zeta[\Psi, \Phi(\mathbf{x}', t)]$ называется «мелкозернистая» плотность (“fine-grained” density) [57], и уравнение (12) означает что FDF является фильтрованным по пространству значением мелкозернистой плотности. Таким образом, P_L есть плотность в композиционном пространстве жидкости вдоль \mathbf{x} взвешенную фильтром G . При условии положительности ядра фильтра G , P_L имеет все свойства функции плотности вероятности [58].

"Условно фильтрованное значение" переменной $Q(\mathbf{x}, t)$ определяется как

$$\langle Q(\mathbf{x}, t) | \Psi \rangle_L \equiv \frac{\int_{-\infty}^{+\infty} Q(\mathbf{x}', t) \zeta[\Psi, \Phi(\mathbf{x}', t)] G(\mathbf{x}' - \mathbf{x}) d\mathbf{x}'}{P_L(\Psi; \mathbf{x}, t)}, \quad (14)$$

где $\langle \alpha | \beta \rangle_L$ означает фильтрованное значение α в зависимость от β . Уравнение (14) подразумевает

$$(i) \quad \text{Для } Q(\mathbf{x}, t) = c, \quad \langle Q(\mathbf{x}, t) | \Psi \rangle_L = c, \quad (15)$$

$$(ii) \quad \text{Для } Q(\mathbf{x}, t) = \widehat{Q}(\Phi(\mathbf{x}, t)), \quad \langle Q(\mathbf{x}, t) | \Psi \rangle_L = \widehat{Q}(\Psi), \quad (16)$$

(iii) Интегральное свойство:

$$\langle Q(\mathbf{x}, t) \rangle_L = \int_{-\infty}^{+\infty} \langle Q(\mathbf{x}, t) | \Psi \rangle_L P_L(\Psi; \mathbf{x}, t) d\Psi \quad (17)$$

где c – постоянная, и $Q(\mathbf{x}, t) = \widehat{Q}(\Phi(\mathbf{x}, t))$, означает случай, когда переменная Q может быть полностью описана композиционным переменным $\Phi(\mathbf{x}, t)$. Из этих свойств следует, что фильтрованное значение любой скалярной функции (любые моменты, корреляции скаляров) получается интегрированием по композиционному пространству

$$\langle Q(\mathbf{x}, t) \rangle_L = \int_{-\infty}^{+\infty} \widehat{Q}(\Psi) P_L(\Psi; \mathbf{x}, t) d\Psi \quad (18)$$

FDF определяется точным уравнением переноса [9]

$$\frac{\partial P_L}{\partial t} + \frac{\partial P_L \langle u_j \rangle_L}{\partial x_j} = - \frac{\partial [\langle u_j | \Psi \rangle_L - \langle u_j \rangle_L] P_L}{\partial x_j} + \frac{\partial}{\partial \psi_\alpha} \left[\left\langle \frac{\partial J_j^\alpha}{\partial x_j} \middle| \Psi \right\rangle_L P_L \right] - \frac{\partial [\widehat{S}_\alpha(\Psi) P_L]}{\partial \psi_\alpha} \quad (19)$$

Последний член в правой части этого уравнения, источник химической реакции, находится в замкнутом виде. Второй член в левой части представляет фильтрованную конвекцию FDF в физическом пространстве и также является замкнутым (при условии что $\langle u_j \rangle_L$ известен). Незамкнутыми членами являются первый слагаемый в правой части, обозначающий последствие неразрешенной конвекции подсеточного масштаба, и второй слагаемый в правой части, представляющий влияние молекулярной диффузии. Для замыкания этих членов используются модель градиентной диффузии для конвекции, и модель оценки линейной среднеквадратичной для молекулярной диффузии [59], [60]. Они приведены в виде стохастических дифференциальных уравнений:

$$dX_i^+(t) = \left[\langle u_i \rangle_L + \frac{\partial(\Gamma + \Gamma_t)}{\partial x_i} \right] dt + \sqrt{2(\Gamma + \Gamma_t)} dW_i \quad (20)$$

$$d\phi_\alpha^+(t) = -\Omega_m [\phi_\alpha^+ - \langle \phi_\alpha \rangle_L] dt + S_\alpha(\Phi) dt \quad (21)$$

где $\Omega_m = C_\Omega (\Gamma + \Gamma_t) / \Delta_G^2$ является "частотой смешивания подсеточного масштаба", C_Ω является постоянной модели. В модели dW_i является процессом Винера-Леви [61]. Применяя технику перехода к уравнению Фоккера-Планка [62] получаем

$$\begin{aligned} \frac{\partial P_L}{\partial t} + \frac{\partial P_L \langle u_j \rangle_L}{\partial x_j} = & \frac{\partial}{\partial x_j} \left[(\Gamma + \Gamma_t) \frac{\partial P_L}{\partial x_j} \right] \\ & + \frac{\partial}{\partial \psi_\alpha} \left[\Omega_m (\psi_\alpha - \langle \phi_\alpha \rangle_L) P_L \right] - \frac{\partial [\widehat{S}_\alpha(\Psi) P_L]}{\partial \psi_\alpha}. \end{aligned} \quad (22)$$

Уравнение (22) является смоделированным уравнением переноса FDF. Путем интегрирования этого уравнения можно получить уравнения переносов моментов подсеточного масштаба относительно скалярных функций. Уравнениями для первого подсеточного момента $\langle \phi_\alpha \rangle_L$, и обобщенной подсеточной дисперсии $\tau(\phi_\alpha, \phi_\alpha) = \langle (\phi_\alpha)^2 \rangle_L - (\langle \phi_\alpha \rangle_L)^2$ являются

$$\frac{\partial \langle \phi_\alpha \rangle_L}{\partial t} + \frac{\partial \langle \phi_\alpha \rangle_L \langle u_j \rangle_L}{\partial x_j} = \frac{\partial}{\partial x_j} \left[(\Gamma + \Gamma_t) \frac{\partial \langle \phi_\alpha \rangle_L}{\partial x_j} \right] + \langle S_\alpha \rangle_L \quad (23)$$

$$\begin{aligned} \frac{\partial \tau(\varphi_\alpha, \varphi_\alpha)}{\partial t} + \frac{\partial \langle u_j \rangle_L \tau(\varphi_\alpha, \varphi_\alpha)}{\partial x_j} &= \frac{\partial}{\partial x_j} \left[(\Gamma + \Gamma_t) \frac{\partial \tau(\varphi_\alpha, \varphi_\alpha)}{\partial x_j} \right] - 2\Omega_m \tau(\varphi_\alpha, \varphi_\alpha) \\ &+ 2(\Gamma + \Gamma_t) \left[\frac{\partial \langle \phi_\alpha \rangle_L}{\partial x_j} \frac{\partial \langle \phi_\alpha \rangle_L}{\partial x_j} \right] + 2(\langle \phi_\alpha S_\alpha \rangle_L - \langle \phi_\alpha \rangle_L \langle S_\alpha \rangle_L) \end{aligned} \quad (24)$$

Эти уравнения идентичны тем уравнениям, которые получаются путем применения модели замыкания к фильтрованным уравнениям переноса компонентов течения. Тем не менее, эти уравнения требуют дополнительного моделирования для членов связанные с источником химических реакций.

2 ЧИСЛЕННАЯ МОДЕЛЬ

Для численного решения дифференциальных уравнений в частных производных (УЧП), как известно, есть ряд различных методов. Среди них широко используются методы конечных разностей, конечных элементов, и конечных объемов, которые являются методами используемые для получения дискретных представлений пространственных производных операторов. Если же нужно проинтегрировать заданное УЧП по времени, есть также различные методы численного интегрирования обыкновенных дифференциальных уравнений. С таким набором разнообразных, успешных, хорошо апробированных методов, возникает вопрос, почему существует необходимость рассмотреть еще один метод. Чтобы оценить это, необходимо рассмотреть сильные и слабые стороны стандартных методов.

Исторически самым первым методом является метод конечных разностей. Одним из наиболее привлекательных аспектов этого метода является его простота, то есть дискретизация общих задач и операторов часто интуитивна и для многих задач приводит к очень эффективным схемам. Кроме того, явная полудискретная форма дает гибкость в выборе методов временного шага, если это необходимо. Также, эти методы поддерживаются обширной теоретической базой [63], они достаточно надежны и эффективны, чтобы их можно было использовать для решения широкого круга задач.

Вместе с тем, зависимость от локальной одномерной полиномиальной аппроксимации является недостатком метода, так как это обеспечивает простую размерную структуру в более высоких измерениях. Дополнительные сложности, вызванные простой базовой структурой, возникают около границ и около разрывных внутренних слоев. Такие свойства делают метод конечных разностей плохо подходящим для рассмотрения задач со сложной геометрией, с точки зрения, как и общих вычислительных областей так и внутренних разрывов.

Приведенное выше обсуждение подчеркивает, что для обеспечения геометрической гибкости необходимо отказаться от простого одномерного приближения в пользу чего-то более общего. Наиболее естественным подходом является введение дискретизации на основе элементов. Следовательно, предполагается, что физическая область представляется коллекцией элементов, организованных неструктурированным образом для заполнения физической области.

Метод, тесно связанный с методом конечных разностей, но в добавок имеющий геометрическую гибкость, является метод конечных объемов [64], который в простейшем виде аппроксимирует решение на элементе константой в центре элемента. В методе аппроксимация и схема являются чисто локальными и, таким образом, не накладывают никаких условий на структуру сетки. В частности, все ячейки могут иметь разные размеры. Схема содержит потоковые источники, которые можно свести к чисто поверхностному источнику с помощью теоремы

Гаусса-Остроградского. Этот шаг вводит необходимость вычисления потоков на границах. Однако, поскольку наши неизвестные являются средними по ячейкам численного решения, вычисление этих потоков требует неоднозначных действий. Эта проблема реконструкции и последующее построение потоков на границах раздела могут быть решены многими различными способами, что приводят к различным методам конечных объемов.

Для линейных задач и равноотстоящих сеток все эти методы сводятся к методу конечных разностей. Однако легко понять, что реконструкция значений решения на интерфейсах является локальной процедурой и обобщается непосредственно на неструктурированные сетки в больших размерностях, обеспечивая тем самым желаемую геометрическую гибкость. Кроме того, построение потоков на границе раздела может быть сделано различными способами, которые тесно связаны с конкретными уравнениями. Это особенно важно, когда учитываются нелинейные законы сохранения.

Необходимость реконструкции схемы высокого порядка разрушает геометрическую гибкость метода конечных объемов в двумерных и трехмерных задачах. Это подрывает мотивацию для рассмотрения метода конечного объема. На неструктурированных сетках этот подход требует реконструкции, основанной на действительно многомерных многочленах с общими местоположениями центра ячейки, которая является одновременно сложной и подверженной проблемам стабильности. Основным ограничением методов конечного объема является его неспособность аппроксимировать уравнения с обеспечением высокого порядка точности на общих неструктурированных сетках.

Ниже в таблице 3 приведены преимущества и недостатки классических численных методов. Где ✓ означает, что метод включает соответствующее преимущество или решает проблему; ✗ означает, что не имеет или не решает; и (✓) означает, что метод решает данную проблему, но с некоторыми модификациями.

Таблица 3 – Некоторые свойства рассмотренных методов

Методы	Сложная геометрия	Высокий порядок точности	Явная полу дискретная форма	Закон сохранения	Эллиптические проблемы
Метод конечных разностей	✗	✓	✓	✓	✓
Метод конечных объемов	✓	✗	✓	✓	(✓)
Метод конечных элементов	✓	✓	✗	(✓)	✓

Развитием классического метода конечных элементов, основанного на вариационной постановке Галеркина, является разрывный метод Галеркина. Метод был разработан в 1973 году [65]. Метод сочетает в себе преимущества метода конечных объемов и спектральных методов. Преимуществом метода конечных объемов является способность решать задачи в геометрически сложных областях, в то же время преимуществом спектральных методов является обеспечение высокой точности аппроксимации. Далее дается детальная постановка метода.

2.1 Разрывный метод Галеркина

В пространственной дискретизации для заданного закона сохранения

$$\frac{\partial q_h^k(\mathbf{x}, t)}{\partial t} + \nabla \cdot \mathbf{f}_h^k(q_h) = 0 \quad (25)$$

предполагается, что локальное решение q_h^k и локальный поток \mathbf{f}_h^k выражаются в модальной форме следующим образом

$$\mathbf{x} \in D^k: q_h^k(\mathbf{x}, t) = \sum_{i=1}^N \hat{q}_i^k(t) \varphi(x_i), \mathbf{f}_h^k(q_h(\mathbf{x}, t)) = \sum_{i=1}^N \hat{\mathbf{f}}_i^k(t) \varphi(x_i) \quad (26)$$

Здесь k - номер элемента, \hat{q}_i^k и $\hat{\mathbf{f}}_i^k$ являются модами (коэффициенты), а $\{\varphi(x_i)\}_{n=1}^N$ - полиномиальный базис порядка N .

Уравнение (25) умножается на гладкую тестовую функцию $v_h \in V_h$, где V_h - пространство всех кусочно-полиномиальных функций. Это приводит к следующей схеме DG в слабой форме

$$\int_{D_k} \left(\frac{\partial q_h^k(\mathbf{x}, t)}{\partial t} v_h + \mathbf{f}_h^k(q_h) \cdot \nabla v_h \right) d\mathbf{x} = \int_{\partial D_k} \hat{\mathbf{n}} \cdot \mathbf{f}^* v_h d\mathbf{x} \quad (27)$$

Здесь $\hat{\mathbf{n}}$ обозначает нормаль, а \mathbf{f}^* - численный поток, который вычисляется с помощью решателя Римана. В качестве набора базисных и тестовых функций выбираются полиномы Лежандра и используются в сочетании с квадратурными точками Гаусса.

Используя более компактную запись, можно восстановить полудискретную схему

$$M_{ij} \frac{d\hat{q}_i^k(x_i, t)}{dt} - S_{ij} \hat{\mathbf{f}}_i^k = \int_{\partial D_k} \hat{\mathbf{n}} \cdot \mathbf{f}^*(x_i) d\mathbf{x} \quad (28)$$

здесь

$$M_{ij} = \int_{D_k} (\varphi(x_i)\varphi(x_j)) dx, \quad S_{ij} = \int_{\partial D_k} \varphi(x_i)\nabla\varphi(x_j) dx \quad (29)$$

где M_{ij} – элементы массовой матрицы, а S_{ij} являются элементами матрицы жесткости.

В качестве базисных функции используется полиномы Лежандра, которые представляют собой систему полных и ортогональных многочленов. Первые несколько полиномов Лежандра:

$$P = \begin{pmatrix} 1 \\ x \\ \frac{3}{2}x^2 - \frac{1}{2} \\ \frac{5}{2}x^3 - \frac{3}{2}x \\ \frac{35}{8}x^4 - \frac{15}{4}x^2 + \frac{3}{8} \\ \frac{63}{8}x^5 - \frac{35}{4}x^3 + \frac{15}{8}x \end{pmatrix} \quad (30)$$

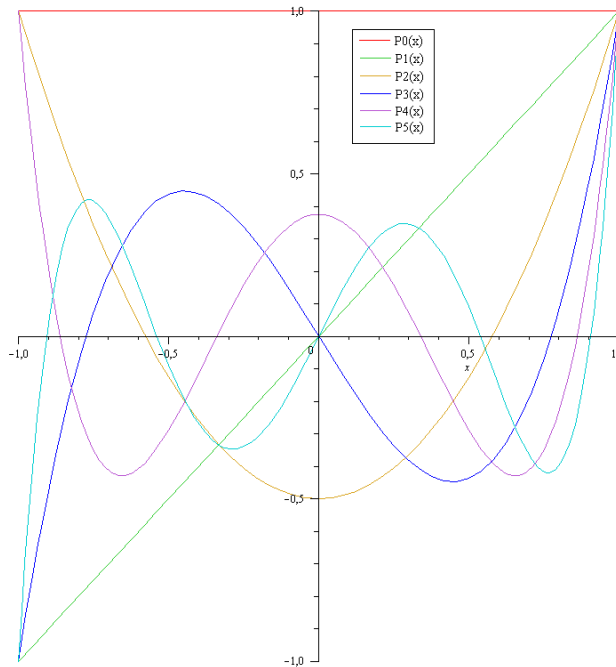


Рисунок 3 – Иллюстрация первых 6 многочленов Лежандра

2.2 Схема расщепления по времени

Эффективным средством численного решения зависящих от времени задач несжимаемой жидкости является проекционный метод проекции Чорина [66]. Для удобства запишем фильтрованное уравнение сохранения импульса (7) в векторной форме

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z} = 0 \quad (31)$$

где

$$U = \begin{pmatrix} u \\ v \\ w \end{pmatrix}, F = \begin{pmatrix} uu + \frac{p}{\rho} - v \frac{\partial u}{\partial x} + T_{xx} \\ uv - v \frac{\partial v}{\partial x} + T_{xy} \\ uw - v \frac{\partial w}{\partial x} + T_{xz} \end{pmatrix},$$

$$G = \begin{pmatrix} uv - v \frac{\partial u}{\partial y} + T_{xy} \\ vv + \frac{p}{\rho} - v \frac{\partial v}{\partial y} + T_{yy} \\ vw - v \frac{\partial w}{\partial y} + T_{yz} \end{pmatrix}, H = \begin{pmatrix} uw - v \frac{\partial u}{\partial z} + T_{xz} \\ vw - v \frac{\partial v}{\partial z} + T_{yz} \\ ww + \frac{p}{\rho} - v \frac{\partial w}{\partial z} + T_{zz} \end{pmatrix}.$$

тогда реализацию метода проекции можно осуществить в два этапа. На первом этапе вычисляются так называемые промежуточные значения вектора скорости U^* из основного уравнения сохранения импульса без учета градиента давления

$$\frac{U^* - U^n}{\Delta t} + \frac{\partial F^n}{\partial x} + \frac{\partial G^n}{\partial y} + \frac{\partial H^n}{\partial z} = 0 \quad (32)$$

где

$$U^* = \begin{pmatrix} u \\ v \\ w \end{pmatrix}^*, U^n = \begin{pmatrix} u \\ v \\ w \end{pmatrix}^n, F^n = \begin{pmatrix} uu - v \frac{\partial u}{\partial x} \\ uv - v \frac{\partial v}{\partial x} \\ uw - v \frac{\partial w}{\partial x} \end{pmatrix}^n,$$

$$G^n = \begin{pmatrix} uv - v \frac{\partial u}{\partial y} \\ vv - v \frac{\partial v}{\partial y} \\ vw - v \frac{\partial w}{\partial y} \end{pmatrix}^n, H^n = \begin{pmatrix} uw - v \frac{\partial u}{\partial z} \\ vw - v \frac{\partial v}{\partial z} \\ ww - v \frac{\partial w}{\partial z} \end{pmatrix}^n.$$

Здесь, U^n – вектор скорости на n -м шаге по времени.

На втором этапе алгоритма промежуточная скорость корректируется, чтобы получить окончательное решение U^{n+1} на $n+1$ временном шаге:

$$U^{n+1} = U^* - \nabla p^{n+1} \Delta t. \quad (33)$$

Для вычисления правой части на втором этапе требуется определить значения давления p на временном слое $n + 1$. Это осуществляется путем использования условия непрерывности $\nabla \cdot U^{n+1} = 0$ на временном шаге $n + 1$, что в результате приводит к уравнению Пуассона для давления

$$\nabla^2 p^{n+1} = \frac{\nabla \cdot U^*}{\Delta t}. \quad (34)$$

Подставляя решение этого уравнения в уравнение (33) находим скорость на временном шаге $n + 1$.

2.3 Пространственная дискретизация основных уравнений

После временной дискретизации существует ряд возможных стратегий для дискретизации пространственных производных с помощью метода DG [21]. Для нелинейных конвективных членов, уравнение (32), используется схема DG против потока, чтобы улучшить стабильность в потоках с преобладанием адвекции. Для вычисления давления, уравнение (34), используется центральная схема с достаточной стабилизацией, чтобы избежать ложных скачков решений между элементами. Для диффузионных членов, также применяется центральная схема. Далее мы распишем более детально аппроксимацию всех производных членов и общую реализацию алгоритма решения численной схемы.

Глобальное решение какой-либо функции можно представить, как прямую сумму локальных кусочно-полиномиальных решений в виде

$$f(\mathbf{x}, t) \cong f_h(\mathbf{x}, t) = \bigcup_{K=1}^{N_e} f_h^K(\mathbf{x}^K, t).$$

Предполагается, что локальные решения имеют вид

$$f_h^K(\mathbf{x}, t) = \sum_{i=1}^{N_p} \hat{f}_i^K(t) \psi_i(\mathbf{x}), \quad \mathbf{x} \in D_K = [\mathbf{x}_-^K, \mathbf{x}_+^K]$$

Здесь $\mathbf{x}_-^K \equiv (x_-^K, y_-^K, z_-^K)$ и $\mathbf{x}_+^K \equiv (x_+^K, y_+^K, z_+^K)$ являются граничными точками K-го элемента. Далее вводим аффинное отображение

$$\mathbf{x}(\xi) = \mathbf{x}_-^K + \frac{1+\xi}{2} \Delta \mathbf{x}^K, \quad \Delta \mathbf{x}^K = \mathbf{x}_+^K - \mathbf{x}_-^K,$$

где $\xi \equiv (\xi, \eta, \zeta) \in I = [-1, 1]$ является опорной переменной. В этом случае, локальные решения имеют следующий вид

$$f_h^K(\xi, t) = \sum_{i=1}^{N_p} \hat{f}_i^K(t) \psi_i(\xi).$$

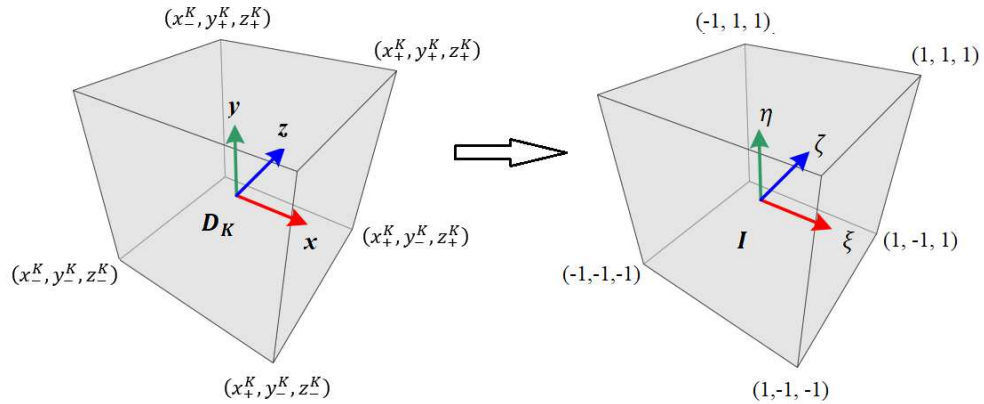


Рисунок 4 – Иллюстрация перехода в локальную систему координат

Если N является порядком полинома, то N_p представляет количество членов в локальном разложении. Эти два параметра связаны следующим выражением

$$N_p = \frac{(N+1)(N+2)(N+3)}{6}$$

Теперь необходимо искать ортонормированный базис как более подходящий и стабильный в вычислительном отношении подход. Для этого нормируем классические базисы Лежандра (30)

$$\psi_N(\xi) = \widehat{P}_{N-1}(\xi) = \frac{P_{N-1}(\xi)}{\sqrt{\gamma_{N-1}}}, \quad \gamma_N = \frac{2}{2N+1}.$$

Здесь, \widehat{P}_N являются ортонормированными базисными функциями Лежандра, где γ_N есть нормирование. Построение новых ортонормированных базисных функции любого порядка можно осуществить с помощью простой рекуррентной формулы

$$\widehat{P}_N(\xi) = \frac{\xi \widehat{P}_{N-1}(\xi) - a_{N-1} \widehat{P}_{N-2}(\xi)}{a_N},$$

$$a_N = \sqrt{\frac{N^2}{(2N+1)(2N-1)}},$$

где первые два базисных функции определены как

$$\psi_1(\xi) = \widehat{P}_0(\xi) = \frac{\sqrt{2}}{2}, \quad \psi_2(\xi) = \widehat{P}_1(\xi) = \frac{\sqrt{6}}{2} \xi.$$

Далее переходим к применению метода DG для каждого шага временной аппроксимации по методу Чорина фильтрованных уравнений Навье-Стокса. Для этого поступаем аналогичным образом как расписано в главе 2.1. Уравнение сохранения (32) умножается на базисную функция и интегрируется по объему в каждом элементе. После всех математических манипуляций восстанавливаем окончательную схему общего вида

$$\begin{aligned} M_{i,j} \frac{(\widehat{U}_i^K)^* - (\widehat{U}_i^K)^n}{\Delta t} + \left[\frac{2}{\Delta x} \left(((F^K)^* \psi_i)|_{\xi=1} - ((F^K)^* \psi_i)|_{\xi=-1} - Sx_{ij} \widehat{F}_j^K \right) \right. \\ \left. + \frac{2}{\Delta y} \left(((G^K)^* \psi_i)|_{\eta=1} - ((G^K)^* \psi_i)|_{\eta=-1} - Sy_{ij} \widehat{G}_j^K \right) \right. \\ \left. + \frac{2}{\Delta z} \left(((H^K)^* \psi_i)|_{\zeta=1} - ((H^K)^* \psi_i)|_{\zeta=-1} - Sz_{ij} \widehat{H}_j^K \right) \right]^n = 0 \end{aligned} \quad (35)$$

где массовая матрица M_{ij} и матрицы жесткости Sx_{ij} , Sy_{ij} , Sz_{ij} имеют следующий вид

$$M_{ij} = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \psi_j \psi_i d\xi d\eta d\zeta = \delta_{ij},$$

$$Sx_{ij} = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \psi_j \frac{\partial \psi_i}{\partial \xi} d\xi d\eta d\zeta,$$

$$Sy_{ij} = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \psi_j \frac{\partial \psi_i}{\partial \eta} d\xi d\eta d\zeta,$$

$$Sz_{ij} = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 \psi_j \frac{\partial \psi_i}{\partial \zeta} d\xi d\eta d\zeta.$$

За счет ортонормированных базисных функции, массовая матрица M_{ij} является единичной матрицей. Это способствует тому, что конечная схема имеет явную форму.

Далее необходимо определить численные потоки, которые в схеме отмечены знаком “*”. Конвективные численные потоки f^* вычисляются с помощью метода Лакса-Фридриха, который в общем случае имеет следующий вид

$$f^*(u^{\text{int}}, u^{\text{ext}}) = \frac{f(u^{\text{int}}) + f(u^{\text{ext}})}{2} + \frac{C}{2}(u^{\text{int}} - u^{\text{ext}}),$$

где u^{int} и u^{ext} представляют собой внутреннее и внешнее значение, соответственно. Важным моментом при вычислении численных потоков является определение коэффициента C . Существует множество способов определения этого коэффициента, которые зависят от типа рассматриваемой задачи. Для моделирования несжимаемых течений можно применить монотонную схему Русанова где

$$C = \max(|f'(u^{\text{int}})|, |f'(u^{\text{ext}})|).$$

Теперь рассмотрим аппроксимацию диффузионных членов. В этом случае численные потоки вычисляются следующим образом

$$q^* = (\nabla u)^* = \frac{q^{\text{int}} + q^{\text{ext}}}{2} + \tau(u^{\text{int}} - u^{\text{ext}}),$$

$$u^* = \frac{u^{\text{int}} + u^{\text{ext}}}{2}.$$

Здесь численный поток u^* возникает при вычислении первых производных ∇u , и численный поток q^* возникает при вычислении вторых производных $\nabla \cdot q$. Штрафной параметр τ можно выбрать из условия

$$\tau \geq \Theta \frac{(N+1)^2}{h}, \Theta \geq 1.$$

где h представляет размер сеточного элемента.

Далее необходимо определить граничные условия для вычисления ∇u и $\nabla \cdot \mathbf{q}$. Однородные граничные условия Дирихле накладываются путем определения внешних значений u^{ext} и q^{ext}

$$u^{\text{ext}} = -u^{\text{int}}, q^{\text{ext}} = q^{\text{int}},$$

отсюда получается

$$q^* = q^{\text{int}} - 2\tau u^{\text{int}},$$

$$u^* = 0.$$

Аналогичным образом накладываются условия Неймана:

$$u^{\text{ext}} = u^{\text{int}}, q^{\text{ext}} = -q^{\text{int}},$$

в итоге численные потоки имеют следующий вид

$$q^* = 0,$$

$$u^* = u^{\text{int}}.$$

2.4 Решение FDF методом Монте Карло

Решение уравнения переноса для FDF предоставляет всю статистическую информацию относительно скалярной переменной $\Phi(x, t)$. Это уравнение может быть решено эффективно с помощью лагранжевого метода Монте-Карло [67]. Преимущество этого метода хорошо известен тем что в нем особо не проявляется численная диффузия. Основой Лагранжевого решения уравнения переноса для FDF является принцип эквивалентных систем. Две системы с разными мгновенными значениями могут иметь идентичную статистику, который полностью удовлетворяет уравнение переноса для FDF. В Лагранжевом методе Монте-Карло каждая частица перемещается согласно определенным уравнениям. Эти частицы подвергаются движению за счет конвекции из-за фильтрованной средней скорости потока, и диффундируют за счет молекулярной и подсеточной диффузии. Главный диффузионный процесс представляется в стохастической форме и описывается следующим стохастическим дифференциальным уравнением (СДУ) [67] [62]

$$dX_i(t) = D_i(X_i(t), t)dt + E(X_i(t), t)dW_i(t), \quad (36)$$

где X_i лагранжевое расположение стохастической частицы, D_i и E известные как коэффициенты “сдвига” и “диффузии” соответственно, W_i означает процесс Винер-Леви.

$$E \equiv \sqrt{2(\Gamma + \Gamma_t)}, \quad D_i \equiv \langle u_i \rangle_L + \frac{\partial(\Gamma + \Gamma_t)}{\partial x_i}, \quad (37)$$

Поэтому СДУ, которое определяет пространственное перемещение FDF, является

$$dX_i(t) = \left(\langle u_i \rangle_L + \frac{\partial(\Gamma + \Gamma_t)}{\partial x_i} \right) dt + \sqrt{2(\Gamma + \Gamma_t)} dW_i(t). \quad (38)$$

Композиционный состав частиц развивается одновременно благодаря действию подсеточного смешивания и реакции

$$\frac{d\varphi_\alpha^+}{dt} = -\Omega_m (\varphi_\alpha^+ - \langle \varphi_\alpha \rangle_L) + S_\alpha, \quad (39)$$

где $\varphi_\alpha^+ = \varphi_\alpha(X_i(t), t)$ является скалярной величиной частицы с лагранжевым вектором расположения X_i .

В численной реализации, фильтрованная функция плотности представляется через совокупность частиц Монте Карло, каждая с набором скаляров $\varphi_\alpha^1(X_i^1(t), t)$ и с лагранжевым вектором положения $X_i(t)$. Численно говоря, используется процедура расщепления в которой перенос в физической и композиционной областях рассматриваются отдельно. Для решения уравнения (38) используется простая явная схема Эйлера-Маруяма

$$X_i^1(t_{n+1}) = X_i^1(t_n) + D_i^1(t_n)\Delta t + E^1(t_n)(\Delta t)^{\frac{1}{2}}\xi_i^1(t_n) \quad (40)$$

где $D_i^n(t_k) = D_i^n(X_i^n(t), t)$, $E^n(t_k) = E^n(X_i^n(t), t)$ и $\xi_i^n(t_k)$ - случайная переменная с Гауссовой функцией распределения плотности. Для уравнения (38) имеются схемы более высокого порядка точности, но нужно быть очень аккуратным при применении их к LES, так как диффузионный член в уравнении (42) зависит от стохастического процесса $X_i^n(t)$. Коэффициенты D_i и E требуют ввода фильтрованных средних значений компонентов скорости и коэффициентов диффузии. Эти переменные получаются путем решения фильтрованных уравнений Навье-Стокса на фиксированной сетке, в нашем случае разрывным методом Галеркина, затем интерполируются к расположениям частиц.

Уравнение (39) может быть численно проинтегрирована для получения мгновенных значений φ_α^+ . Так как $\varphi_\alpha^+ = \varphi_\alpha(X_i(t), t)$, шаг по времени должен соответствовать (40). После численного решения системы стохастических

дифференциальных уравнений (38)-(39), можно построить фильтрованные значения моментов любого порядка относительно скаляров ϕ_α .

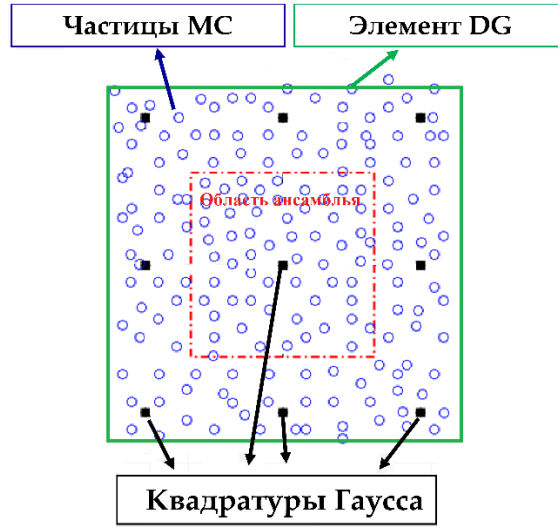


Рисунок 5 – Двумерная визуализация реализации метода MC на сетке DG с квадратурными точками

Статистика строится путем рассмотрения частиц в объеме с центром в интересующей точке. По сути, этот объем представляет собой «область ансамбля», характеризующуюся масштабом длины ΔE , в котором FDF представлен дискретно. Это показано на рисунке 5, где показана квадратная область ансамбля. Для надежной статистики с минимальной дисперсией требуется максимизировать количество частиц и минимизировать размер области ансамбля. Фильтрованные значения функции вычисляются по формуле

$$\langle g \rangle_E = \frac{1}{N_E} \sum_{i \in \Delta E} g^{(i)}, \quad (41)$$

где N_E обозначает количество частиц MC, находящихся в области ансамбля с характерной длиной ΔE с центром вокруг каждой квадратурной точки DG. Переменная частицы $g^{(i)}$ обозначает информацию, переносимую i -й частицей MC, относящуюся к транспортной переменной g

На рисунке 6 представлена общая блок-схема разработанного LES симулятора. Симулятор состоит из двух основных модулей: DG и MC. Сначала мы задаем начальные условия: в модуле DG задаются значения всех переменных и параметров течения во всех узлах сетки; в модуле MC задаются начальное распределение частиц и значения скалярных функции. Затем запускается глобальный цикл – итерация по времени, который состоит из:

- I. Вычисление коэффициентов “сдвига” и “диффузии” (уравнение (37)).

II. Интерполяция значений коэффициентов “сдвига” и “диффузии” на местоположение частиц.

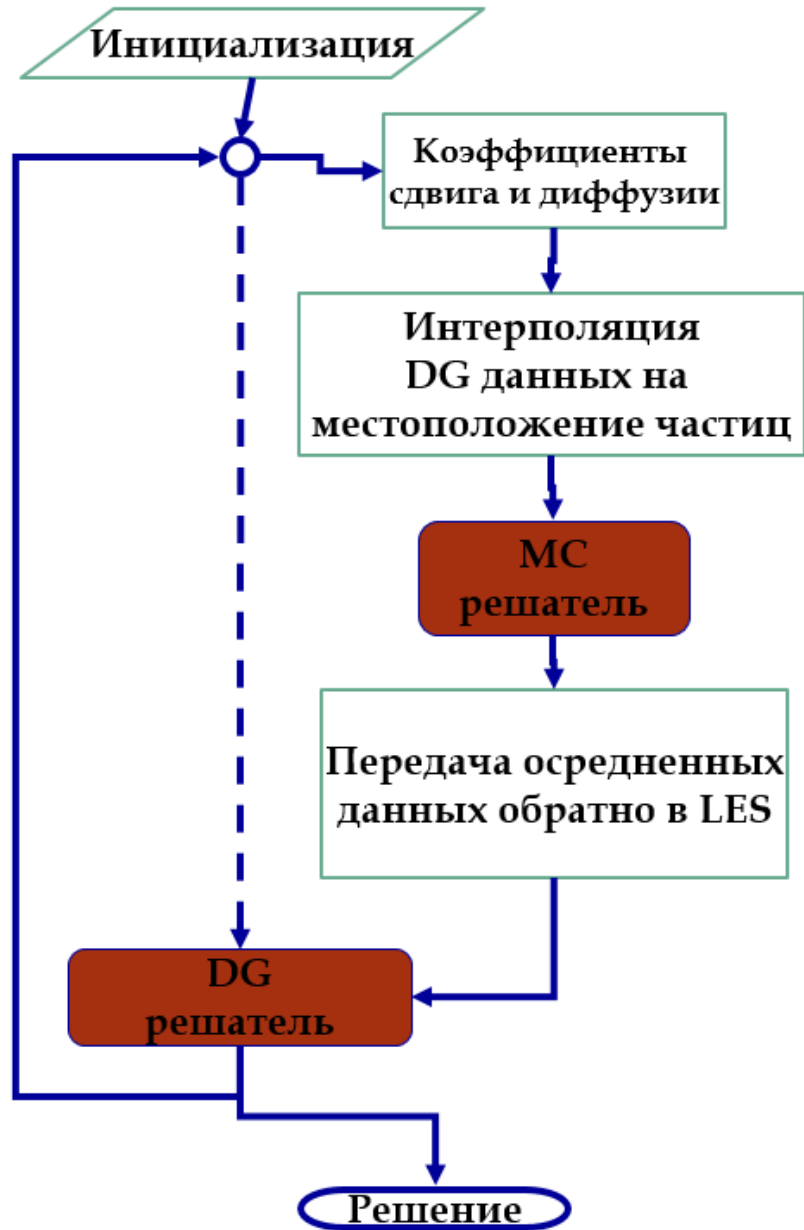


Рисунок 6 – Общий алгоритм гибридной схемы DG-МС

III. Запуск модуля МС. Здесь вычисляются значения φ_{α}^{+} с помощью уравнений (38) и (39). Затем вычисляются осредненные значения относительно скалярных функции с помощью уравнения (41). Приведем примеры расчетов осредненного значения первого момента $\langle \varphi_1 \rangle_L$ и источника химической реакции $\langle S_1 \rangle_L = \langle -K\varphi_1 \varphi_2 \rangle_L$:

$$\langle \varphi_1 \rangle_L = \frac{1}{N_E} \sum_{l=1}^{N_E} (\varphi_1^+)^l$$

$$\langle S_1 \rangle_L = \frac{1}{N_E} \sum_{l=1}^{N_E} (S_1^+)^l = \frac{1}{N_E} \sum_{l=1}^{N_E} K(\varphi_1^+)^l (\varphi_2^+)^l$$

IV. После модуля MC запускается модуль DG, где вычисляются осредненные значения компонентов скорости, давления и скалярных функции. Затем опять переходим на начало цикла по времени с обновленными значениями переменных течения. Цикл повторяется пока не достигнем заданного времени.

В модуле DG, при решении уравнения переноса скалярных функций (8), осредненные значения источника химической реакции $\langle S_\alpha \rangle_L$ получаются из модуля MC. Нужно уточнить, что нет необходимости вычислять значения скалярных функции с помощью модуля DG, так как модуль MC предоставляет все необходимые значения относительно скалярных функций. В нашем случае значения скалярных функций вычисляются, как и с помощью DG, так и с помощью MC с целью проверки консистенции результатов.

3 ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ

Параллельные вычисления существуют в той или иной форме на протяжении многих десятилетий [68]. На ранних стадиях это обычно ограничивалось практикующими, которые имели доступ к большим и дорогим машинам. Сегодня все по-другому. Почти все потребительские настольные и портативные компьютеры имеют центральные процессоры или процессоры с несколькими ядрами. Основной причиной почти повсеместного присутствия многоядерных процессоров в процессорах является неспособность производителей процессоров повысить производительность в одноядерных разработках за счет повышения тактовой частоты .

В то время как центральные процессоры доступны с несколькими или десятками ядер, это количество параллелизмов бледнеет по сравнению с количеством ядер в графическом процессоре. Например, как показано на рисунке 1, если центральный процессор Core I7-6700 от Intel содержит 4 ядра, в то время как графический процессор GeForce GTX 1080Ti от NVIDIA содержит 3584 ядер. Графические процессоры были высокопараллельными архитектурами с самого начала, в середине 1990-х годов, поскольку обработка графики является изначально параллельной задачей.

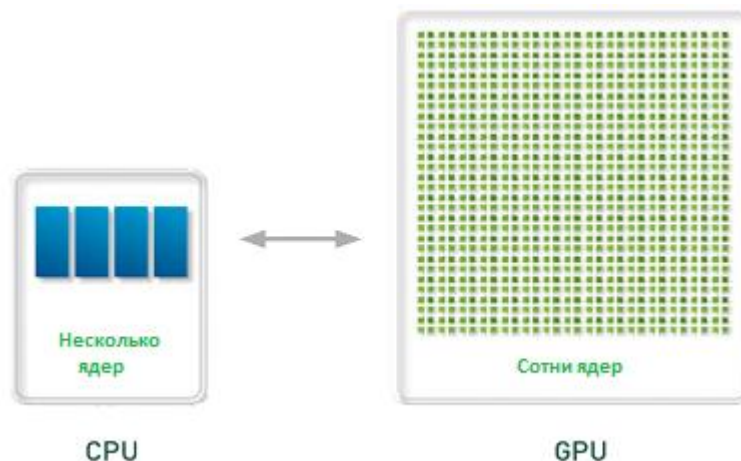


Рисунок 7 – Схематическая визуализация центральных и графических процессоров

Видеоадаптер состоит из основных элементов: видеопамять и графический процессор. Графический процессор состоит из множества мультипроцессоров. Если обычный центральный процессор имеет одно устройство контроля исполнения, кэш и несколько арифметико-логических устройств, в то время как графический процессор имеет набор мультипроцессора, имеющие свои устройства контроля исполнения, область разделяемой памяти, области регистров и значительно расширенный набор арифметико-логических устройств. Благодаря такой структуре можно добиться параллельного выполнения алгоритма на разных мультипроцессорах, и в то же время параллельной обработки данных.

Параллельный алгоритм решателя DG-МС реализована с помощью PGI CUDA Fortran Compiler, а вычисления проводились с использованием видеокарты NVIDIA GTX 1080Ti, которая стоит 700 долларов США на официальном сайте NVIDIA [69]. Данный графический процессор (GPU) оснащен 3584 ядрами CUDA с тактовой частотой 1582 МГц, каждое. А также устройство имеет 11 Гб памяти с пропускной способностью 484 Гбит/с и полосой пропускания шины памяти 352 бит. Производительность построенного алгоритма сравнивается с оптимизированным последовательным алгоритмом. Последовательные вычисления проводились на центральном процессоре (CPU) Core i7-6700 с тактовой частотой 3400 МГц, который стоит около 300 долларов США. Данный процессор оснащен оперативной памятью с пропускной способностью 19,2 Гб/с.

Архитектура CUDA разработана компанией NVIDIA позволяющая проводить вычисления с использованием GPU. Тем самым CUDA является драйвером и языком программирования GPU путем интеграции с существующими языками программирования (C, C++, Python, итд), что позволяет программисту проводить вычисления на графических устройствах, не имея знаний в области компьютерной графики. Также стоит отметить что CUDA является гибридной моделью вычислений, которая может одновременно использовать CPU и GPU. Это дает

возможность переносить для CPU код на GPU. Далее приведем основные общепринятые определения в CUDA Fortran.

В контексте программирования хостом называется CPU и его память, а устройством называется GPU и его память. Термином CPU-код называется часть программы, исполняемый только на хосте, и подпрограмма, выполняемая на устройстве, называется ядром. Код, выполняемый на хосте, управляет памятью как на хосте, так и на устройстве, а также запускает ядра выполняемыми на устройстве. Эти ядра выполняются многими потоками GPU параллельно.

Учитывая неоднородный характер модели программирования CUDA, типичная последовательность операций для кода CUDA Fortran выглядит следующим образом:

- 1) Объявление и выделение памяти на хосте и устройстве.
- 2) Инициализация данных хоста.
- 3) Перенос данных с хоста на устройство.
- 4) Запуск одного или нескольких ядер.
- 5) Перенос результатов с устройства на хост.

Когда ядро запускается, CUDA генерирует сетку потоков, которые организованы в трехмерную иерархии [70]:

- Каждая сетка организована в массив потоковых блоков или просто блоков;
- Каждый блок может содержать до 1024 потоков;
- Количество потоков в блоке указывается в переменной `blockDim`;
- Размер потоковых блоков должен быть кратным 32.

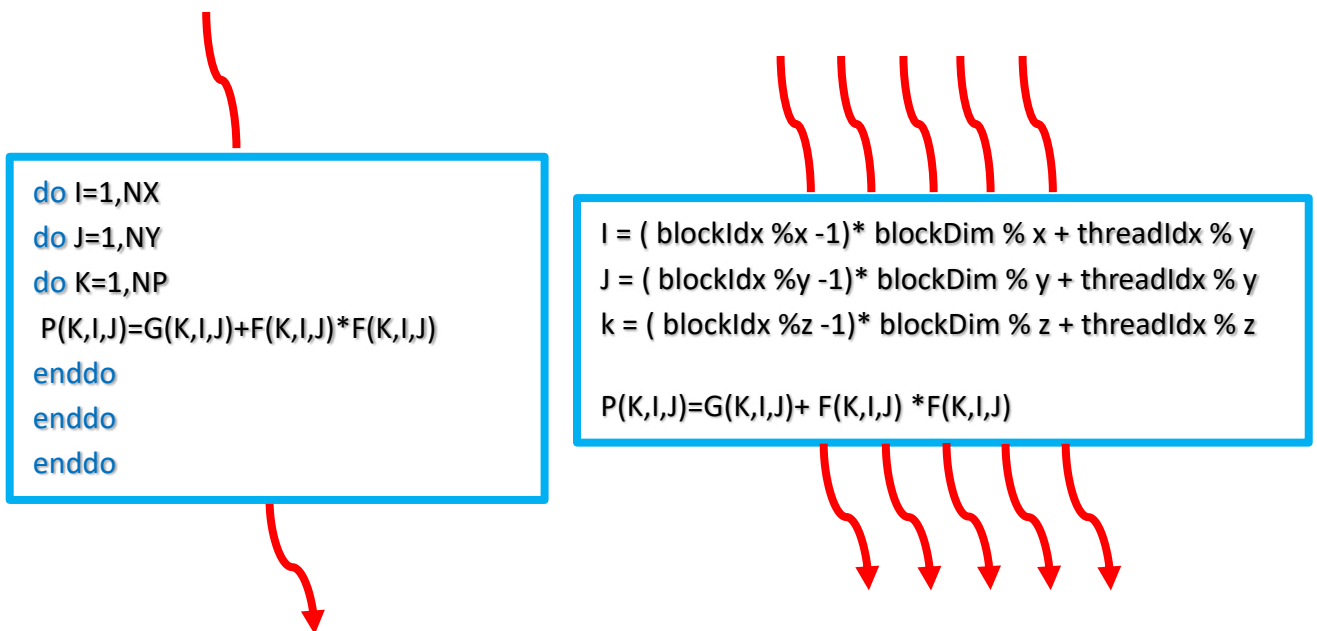


Рисунок 8 – Фрагменты последовательного и параллельного кода

Все потоки в сетке выполняют одну и ту же инструкцию, заданную функцией ядра. Распределение потоков может быть в 1D, 2D или 3D. В качестве примера показать трехмерное распределение потоков, которое генерирует глобальные индексы, которые в свою очередь используются для доступа к элементам массивов:

$$\begin{aligned} I &= (\text{blockIdx} \% x - 1) * \text{blockDim} \% x + \text{threadIdx} \% x \\ J &= (\text{blockIdx} \% y - 1) * \text{blockDim} \% y + \text{threadIdx} \% y \\ K &= (\text{blockIdx} \% z - 1) * \text{blockDim} \% z + \text{threadIdx} \% z \end{aligned}$$

где blockDim - количество потоков в блоке для определенного направления, threadIdx используется для доступа к индексу потока внутри блока потока, а blockIdx используется для доступа к индексу блока потока внутри сетки потоков.

Устройство с графическим процессором оснащено пятью типами памяти. Первый тип, глобальная память, как указывает ее имя, доступна для всех потоков, работающих на устройстве. Это самый объемный тип памяти и данные, хранящиеся в нем, существуют на протяжении всего приложения. Второй - это разделяемая память, которая доступна всем потокам, принадлежащим одному и тому же блоку потоков. Устройство удаляет все данные, хранящиеся в разделяемой памяти, после завершения работы ядра. Разделяемая память намного быстрее, чем глобальная память, благодаря расположению на кристалле, поэтому его латентность примерно в 10-15 раз ниже, чем у кешированной глобальной памяти. Объем разделяемой памяти может быть настроена до 48 КБ на один потоковый мультипроцессор. Разделяемую память можно использовать, чтобы избежать неэффективного доступа к глобальной памяти. Третий тип памяти - это постоянная память, которая предлагает еще одну альтернативу глобальной памяти, когда необходим глобальный доступ с высокой пропускной способностью. Постоянная память очень ограничена в объеме, 64 КБ, и, как видно из названия, используется для хранения данных, которые остаются неизменными в течение всего периода выполнения программы. Два других типа памяти, а именно регистровая и текстурная память, в нашей работе не использовались. При работе с глобальной памятью, группирование операций доступа может значительно повысить вычислительную эффективность. Существует термин, называемый варп, который представляет собой набор потоков, выполняемых в соответствии с принципом SIMT (одна инструкция, несколько потоков). Каждая команда на устройстве входит в основу потоков, и все потоки, принадлежащие этому варпу, выполняют эту команду синхронно. Группировка в варпы важна не только с точки зрения вычислений, но и с точки зрения доступа к глобальной памяти. Максимальная производительность достигается, когда потоки, принадлежащие к одному варпу, получают доступ к данным с использованием минимального количества транзакций.

Чтобы оптимизировать распределение памяти, чтение и запись данных из/в глобальную память должны быть реализованы объединенным способом. Язык

CUDA был разработан для работы со структурами данных predetermined размера. Поэтому в расчетах используются массивы данных постоянных размеров. Поскольку для идеального объединения требуется доступ к данным в 32 смежных элементах в глобальной памяти на каждую нагрузку, данные хранятся в массивах, размеры которых равны множителю 32. Когда всем потокам требуется доступ к одной и той же постоянной переменной, используется постоянная память. Если переменные используются повторно, они кэшируются в разделяемой памяти.

3.1 Разработка параллельного кода DG с помощью технологии CUDA

Существенным преимуществом используемого метода DG является то, что метод реализован элементарно-локальным способом, со слабой формулировкой. Кроме того, увеличение порядка степени полинома приводит к высокой арифметической интенсивности. Оба эти свойства привлекательны для аппаратной архитектуры GPU [37]. В DG используются массивный тип данных. Большая часть времени уходит на работу, связанную с трехмерными (в двумерной задаче) и четырехмерными (в трехмерной задаче) массивами, хранящимися в глобальной памяти. К этим данным относятся все переменные, размеры которых увеличиваются с разрешением сетки и степенью полинома. Те массивы, размеры которых изменяются только с полиномиальной степенью, сравнительно невелики и могут быть помещены в постоянную память для более быстрого доступа. К этим массивам относятся матрицы жесткости и локальные граничные интегральные матрицы. Графические процессоры NVIDIA обеспечивают 64 КБ постоянной памяти и до 48 КБ кэш-памяти первого уровня на каждый мультипроцессор. Когда степень аппроксимирующего полинома p равен нулю, остается только одна постоянная базисная функция (26), и схема эквивалентна дискретизации конечного объема второго порядка. Когда $p = 4$ в каждом элементе DG имеется 15 и 35 базисных функций для двумерного и трехмерного случаев, соответственно. В этом случае общий объем всех постоянных массивов составляет около 18 КБ и 42 КБ, соответственно.

В DG наиболее часто используемые ядрами являются ядра аппроксимации пространственных производных. Реализация ядра x -производной для двумерного случая описана далее. Как показано в уравнении (28), поверхностные и граничные интегралы необходимо вычислить для аппроксимации производных. Фрагмент ядра без оптимизации представлен ниже:

```

1  K = threadIdx % x
2  I = ( blockIdx % y - 1) * blockDim % y + threadIdx % y
3  J = ( blockIdx % z - 1) * blockDim % z + threadIdx % z
4  IF (K<=Nm) THEN
5  DO L=1, Nm
6  STIFF_X= STIFF_X +FH(L, I, J) *SMX(L, K)
7  ENDDO
8  DO L=1, Nm

```

```

9  FLUXR=FLUXR+0.5_DP*( IMRI (L, K) *FH (L, I, J) +IMRE (L, K) *FH (L, I+1, J) )
10 FLUXL=FLUXL+0.5_DP*( IMLI (L, K) *FH (L, I, J) +IMLE (L, K) *FH (L, I-1, J) )
11 ENDDO
12 DFDX(K, I, J)=2.0_DP*( FLUXR- FLUXL - STIFF_X) / DX(I)
13 ENDIF

```

Здесь IMRI, IMRE, IMLI и IMLE - матрицы граничных интегралов. Для высоких чисел полинома большинство элементов этих матриц равны нулю. Чтобы избежать нулевых операции, и чтобы достичь более высокой эффективности может быть использован алгоритм умножения разреженной матрицы на вектор. Алгоритм умножения разреженной матрицы на вектор разработан автором диссертации. Подробности алгоритма приведены в [приложении Ё](#). Здесь в фрагменте кода можно заметить, что каждое значение массива FH используется 4 раза. Так как переменная FH хранится в глобальной памяти, такое обращение является не эффективным из соображения вычислительной производительности. Если посчитать общее число тактов при обращении к FH которое необходимо для определения каждого значения DFDX, то получится число равному 1400. Это число можно уменьшит путем использования разделяемой памяти. Как мы уже говорили, что 1 доступ к разделяемой памяти в нашем устройстве стоит 28 тактов, тогда как к глобальной памяти стоит 350 тактов. Для сокращения количество тактов, определенный сегмент значений переменной FH загружаем в разделяемую память. В этом случае для этой операции число тактов будет равен $350+28=378$. После этого каждый из них считывается из разделяемой памяти 4 раза. Это дополнительные $4*28 = 112$ тактов. Суммируя все, мы получаем в общей сложности 490 тактов. Таким образом, с помощью разделяемой памяти число тактов в ядре аппроксимации производной сократилось 3,35 раза. Код оптимизированного ядра представлен ниже:

```

1  REAL (DP) , DIMENSION (NM*SHY, 0:SHX+1) , SHARED :: S
2
3  I = ( BLOCKIDX %X -1) * BLOCKDIM % X + THREADIDX % X
4  J = ( BLOCKIDX %Y -1) * BLOCKDIM % Y + THREADIDX % Y
5  K = THREADIDX % Z
6  I1= THREADIDX % X
7  J1= THREADIDX % Y
8  IF (K<=NM) THEN
9  S ((J1-1) *NP+K, I1) =FH (K, I, J)
10 S ((J1-1) *NP+K, 0) =FH (K, ( BLOCKIDX %X -1) * BLOCKDIM % X, J)
11 S ((J1-1) *NP+K, SHX+1) =FH (K, BLOCKIDX%X*BLOCKDIM%X+1, J)
12 CALL SYNCTHREADS ()
13 DO L=1, Nm
14 STIFF_X= STIFF_X + S ((J1-1) *NP+L, I1) *SMX (L, K)
15 ENDDO
16 DO L=1, Nm
17 FLUXR=FLUXR+0.5_DP*( IMRI (L, K) * S ((J1-1) *NP+K, I1)

```



```

18          +IMRE(L,K) * S((J1-1)*NP+K,SHX+1))
19 FLUXL=FLUXL+0.5_DP*(IMLI(L,K) * S((J1-1)*NP+K,I1)
20          +IMLE(L,K) * S((J1-1)*NP+K,0))
21 ENDDO
22 DFDX(K,I,J)=2.0_DP*(FLUXR-FLUXL-STIFF_X)/DX(I)
23 ENDIF

```

Здесь S двумерный массив для временного хранения данных расположенный в разделяемой памяти. Индексы I, J соответствуют координатам на сетке 128^2 . Индекс K используется в качестве координаты p фрагмента в разделяемой памяти, тогда как для представления локальных координат x и y фрагментов используются переменные $I1$ и $I2$, соответственно. Это ядро запускается с блоком, содержащим $NM*SHY \times SHX+2$, который вычисляет значения производной $DFDX$ в узлах фрагмента размером $NM*SHY \times SHX+2$. Выбор размера фрагментов обсуждается ниже. Стоит отметить, сто вызов `synctreads()` в строке 12 необходим, потому что потоки, читающие данные в строках 14, 17 и 19, - это не те же потоки, что записывают данные в разделяемую память в строках 9, 10 и 11.

Чтобы обеспечить оптимизацию распределения памяти, каждый блок потоков загружает фрагмент данных в разделяемую память, а затем каждый поток в блоке может обращаться ко всем элементам фрагмента разделяемой памяти по мере необходимости. Здесь ключевой задачей является выбор формы и размера фрагмента. Это может быть сделано с учетом характеристик схемы DG . В качестве тестового примера мы рассмотрим DG с полиномиальным порядком $p = 4$ на сетке размером 128^2 . В этом случае мы имеем размер расчетной сетки $15 \times 128 \times 128$. Поскольку один варп состоит из 32 потоков, идеальное объединение достигается, когда эти потоки получают доступ к данным в 32 смежных элементах в глобальной памяти на одну нагрузку. Чтобы объяснить процедуру выбора оптимальной конфигурации фрагмента, далее мы приведем пример. Для начала рассмотрим размер плитки 15×16 , как показано на левой стороне рисунка 9. В этом случае данные из двух соседних элементов должны быть загружены для каждого блока потока. Это требует дополнительных 30 запросов или 12,5% больше загрузок из глобальной памяти. Изменение размера плитки до 15×32 уменьшает дополнительную нагрузку на 6,25%. При тщательном рассмотрении замечено, что оптимальный размер для этого контрольного примера составляет 15×128 , что не требует каких-либо дополнительных нагрузок. Однако этот размер превышает предел потоков на один блок, который составляет 1024. Эту проблему можно решить, выбрав блок потоков $16 \times 16 \times 1$ или $16 \times 4 \times 4$, где каждый поток вычисляет 8 значений производной.

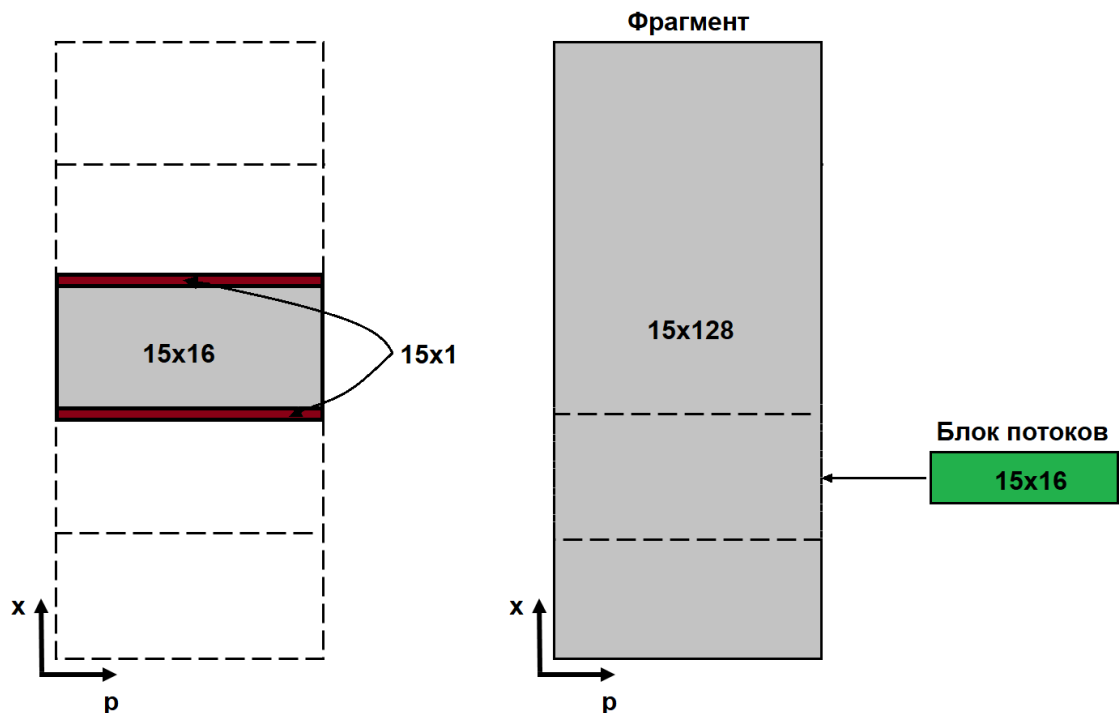


Рисунок 9 – Возможные конфигурации фрагмента для вычисления производной по направлению x при $p=4$

Для трехмерной задачи поступаем аналогичным образом. Отличие состоит лишь в размерности массива:

```

1 I = ( blockIdx %x -1) * blockDim % x + threadIdx % x
2 J = ( blockIdx %y -1) * blockDim % y + threadIdx % y
3 K = ( blockIdx %z -1) * blockDim % z + threadIdx % z
4 DO M=1, Nm
5 DO L=1, Nm
6 STIFF_X= STIFF_X +FH(L, I, J, K) *SMX(L, M)
7 ENDDO
8 DO L=1, Nm
9 FLUXR=FLUXR+0.5_DP*( IMRI(L, M) *FH(L, I, J, K) +IMRE(L, M) *FH(L
, I+1, J, K) )
10 FLUXL=FLUXL+0.5_DP*( IMLI(L, M) *FH(L, I, J, K) +IMLE(L, M) *FH(L
, I-1, J, K) )
11 ENDDO
12 DFDX(M, I, J, K)=2.0_DP*( FLUXR- FLUXL - STIFF_X) / DX(I)
13 ENDDO
14 ENDIF

```

Здесь можно заметить, что количество арифметических операции выполняемое каждым потоком, намного больше чем в двумерном случае. Если в двумерном коде

за один заход каждый поток вычислял только одно значение массива $DFDX$, то в трехмерном случае каждый поток вычисляет N_m элементов. Как мы уже упоминали выше, рост арифметической интенсивности позитивно отражается в производительности параллельного GPU алгоритма.

Таблица 4 – Время работы в секундах GPU для разных случаев использования глобальной памяти.

p	2D распределение			3D распределение		
	Глобальная	Глобальная/ Постоянная	Глобальная/ Разделяемая	Глобальная	Глобальная/ Постоянная	Глобальная/ Разделяемая
1	2.906	2.856	2.782	2.792	2.735	2.805
2	3.838	3.424	3.365	3.192	3.204	3.081
3	6.805	5.577	5.380	4.806	4.514	3.867
4	13.485	11.202	9.254	6.630	7.843	5.711

Следующий вопрос, который необходимо рассмотреть, заключается в том, как указать конфигурацию блока: определение количества потоков в блоке и размерности блоков. В соответствии с руководством CUDA [70], рекомендуется использовать 128/256 потоков на блок. Поэтому для каждой симуляции выбирается 256 потоков для каждого блока. Различные размерности распределения потоков могут выполняться для запуска ядер. Было проверено производительность кода DG с использованием двумерного распределения потоков, где индексы обозначают только координаты; и с помощью трехмерного распределения потоков, где индексы обозначают мод и координаты (p и x,y). В качестве тестовой задачи рассмотрен численный расчет двумерного уравнения Пуассона на фиксированной сетке с разрешением $N_x = 64^2$ и различными полиномиальными степенями $p = 1..4$. Количество итерации зафиксирован и равен 15 000. Таблица 4 показывает время вычисления графического процессора для разных случаев и указывает, что параллельный алгоритм графического процессора работает быстрее, когда глобальная память используется в сочетании с разделяемой памятью и с распределением трехмерных потоков.

3.2 Разработка параллельного кода MC с помощью технологии CUDA

Благодаря независимой реализации каждой частицы MC параллельный алгоритм может быть существенно ускорена графическими процессорами. Процедура MC состоит из пяти этапов: интерполяция данных от элементов DG к позициям частиц, генерация случайных чисел, численное интегрирование стохастических дифференциальных уравнений, подсчет частиц в каждом элементе и усреднение. Решение стохастических дифференциальных уравнений (уравнения (20) - (21)) требует ввода отфильтрованных значений скорости, коэффициента диффузии и градиентов скалярного поля в местах расположения частиц. Эти

значения получаются путем численного решения системы (1) -(2) с помощью DG и последующей интерполяции к местоположению частиц. Процедура интерполяции осуществляется посредством:

$$\hat{U}^l (\hat{X}P^l, \hat{Y}P^l, \hat{Z}P^l) = \sum_{n=1}^{Nm} U_n^{l,J,K} P_n(\hat{X}P^l, \hat{Y}P^l, \hat{Z}P^l) \quad (42)$$

где \hat{U}^l - интерполированное значение частицы l, $\hat{X}P^l$, $\hat{Y}P^l$ и $\hat{Z}P^l$ - положения частиц. $U_n^{l,J,K}$ - моды интерполирующей функции в элементе I, J, K, а P_n - полиномы Лежандра. Частицы движутся в физическом пространстве за счет конвекции из-за отфильтрованной средней скорости потока и диффузии из-за молекулярной и подсеточной диффузии. Следовательно, необходимо определять местоположение каждой частицы в сетке DG на каждом временном шаге. Часть кода, которая выполняет эту задачу представлено следующим образом:

```

1 l = ( blockidx %x -1) * blockdim % x + threadidx % x
2 Shx= XP(1)
3 Shy= YP(1)
4 Shz= ZP(1)
5 I=MY_ID_I(Shx)
6 J=MY_ID_J(Shy)
7 K=MY_ID_K(Shz)
8 DO n=1, Nm
9 legendre_f=P(n, Shx, Shy, Shz)
10 UP(1)= UP(1)+U(n, I, J, K)*legendre_f
11 ENDDO

```

Здесь l определяет номер частиц, XP, YP и ZP определяют местоположение этой частицы. MY_ID_I, MY_ID_J и MY_ID_K - это функции, которые возвращают значения I, J и K в соответствии с XP, YP и ZP, соответственно. Строки 9-10 являются реализацией уравнения (42), где UP - интерполированное значение частицы l, U - моды интерполирующей функции (например, скорости), а P - функция Лежандра. Чтобы рассчитать каждое значение UP, необходимо загрузить из глобальной памяти значения U, XP, YP и ZP. Чтобы уменьшить количество обращений к глобальной памяти, XP, YP и ZP считываются из глобальной памяти и записываются в Shx, Shy и Shz, которые определены в локальной памяти, которая находится в кэше L1. Также можно заметить, что значения U читаются из глобальной памяти $Nq^2 \times NPQ$ и $Nq^3 \times NPQ$ раза в двумерном и трехмерном случаи, соответственно. Это среднее количество частиц на элемент DG. Для эффективного доступа к глобальной памяти можно воспользоваться разделяемой памятью. В этом случае Nm значений U копируются в разделяемую память, затем все потоки блока имеют быстрый доступ к этим данным.



Рисунок 10 – Блок схема процедуры Монте Карло

Численное интегрирование стохастических дифференциальных уравнений (20) - (21) требует ввода N_p случайных чисел в каждом направлении. Эти числа генерируются с помощью библиотеки CURAND. Эта библиотека включает несколько подпрограмм для генерации случайных чисел с равномерным, нормальным и логарифмическим распределениями. Каждая из этих подпрограмм может генерировать числа с одинарной или двойной точностью. Для генерации случайного числа необходимо выполнить следующие действия: создать генератор с помощью функции `curandCreateGenerator()`, установить начальные значения с помощью функции `curandSetPseudoRandomGeneratorSeed()`, сгенерировать случайные числа с желаемым распределением и последним необходимым - уничтожить генератор с помощью функции `curandDestroyGenerator()`.

Для повышения эффективности, массивы случайных чисел предварительно генерируются на каждом временном шаге и временно сохраняются в глобальной памяти для их использования в численном интегрировании стохастических дифференциальных уравнений. Согласно [71], скорость сходимости методов МС не зависит от того, используем ли мы одинарную или двойную точность. Это зависит только от количества сгенерированных случайных чисел. Следовательно, мы можем использовать случайные числа с одинарной точностью и в то же время поддерживать общий порядок точности. Это позволяет значительно сократить время вычислений. В интегрировании доступ к глобальной памяти полностью

объединена. Тем не менее, есть некоторые данные, такие как отфильтрованная средняя скорость и отфильтрованные композиционные скаляры, которые используются несколько раз для каждой ансамблевой области. Здесь подход доступа к памяти такой же, как в процедуре интерполяции. Отфильтрованные величины в квадратурных точках копируются из глобальной памяти в разделяемую память для эффективного повторного использования в ядре.

Построение осредненных значений делится на два этапа. На первом этапе подсчитывается количество частиц в ансамблевом домене, а на втором этапе мы определяем отфильтрованные величины с помощью уравнения 4. И подсчет, и усреднение частиц реализованы с использованием директив ядра CUF. Фрагмент кода представлен ниже:

```
1  NumberOfParticles=0
2  !$cuf kernel do <<<*,*>>>
3  DO M=1,NP
4  I=INT((XP(M)+L*0.5)*NX/L+0.5)
5  J=INT((YP(M)+L*0.5)*NY/L+0.5)
6  K=INT((ZP(M)+L*0.5)*NZ/L+0.5)
7  NumberOfParticles(I,J,K)=NumberOfParticles(I,J,K)+1
8  ENDDO
```

Здесь M – номер частицы, строки 4-6 определяют индексы элемента, к которому принадлежит M -я частица. Строка 2, `cuf` директива, автоматически генерирует ядра из обычного цикла (строки 3-8 написаны для последовательной компиляции, выполняемой на CPU), давая полностью параллельную процедуру. Более того, CUF-ядра могут распознавать операции редукции, делая реализацию еще более упрощенной. На этапе усреднения доступ к глобальной памяти полностью объединен, и никакой оптимизации не требуется.

4 ПАРАМЕТРЫ ТЕЧЕНИЯ

Для того чтобы показать эффективность разработанной численной методологии моделирования крупных вихрей на основе FDF моделирования, рассмотрена задача развивающегося по времени слоя смешения. Этот слой в начальный момент состоит из двух параллельных потоков, текущих в противоположных направлениях с одинаковыми по абсолютной величине скоростями. На рисунке 11 представлена схематическая иллюстрация задачи, где x соответствует потоку жидкости в продольном направлении, y и z – поперечным направлениям. Компоненты скорости в этих направлениях обозначены как u и v , w , соответственно.

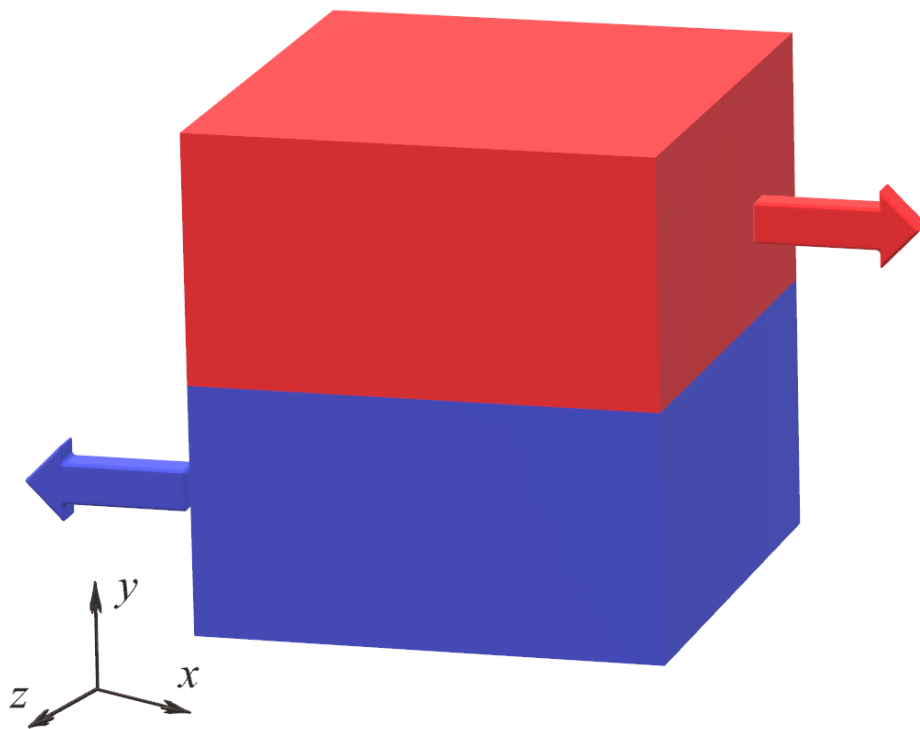


Рисунок 11 – Схематическая иллюстрация начального распределения компонентов смеси. Красным цветом указана компонента А, движущаяся в продольном направлении со скоростью $u = 1$. Синим цветом указана компонента В, движущаяся в противоположном направлении со скоростью $u = -1$

Профиль скорости потока по направлению x задается в виде гиперболической функций $u(x, y, z, t = 0) = \text{th}(\theta y)$, где θ – толщину слоя. Для формирования крупномасштабных когерентных структур и обеспечения трехмерности течения, налагаются возмущения низких амплитуд по всем направлениям [72]. Возмущения имеют следующий вид:

$$\begin{aligned}
u' &= 2\lambda y e^{-y^2} \left(\sin(\alpha x) + \sin\left(\frac{1}{2}\alpha x\right) + \sin\left(\frac{1}{4}\alpha x\right) \right), \\
v' &= \alpha \lambda e^{-y^2} \left(\cos(\alpha x) + \frac{1}{2}\cos\left(\frac{1}{2}\alpha x\right) + \frac{1}{4}\cos\left(\frac{1}{4}\alpha x\right) \right), \\
w' &= 2\lambda y e^{-y^2} \left(\sin(\alpha z) + \sin\left(\frac{1}{2}\alpha z\right) + \sin\left(\frac{1}{4}\alpha z\right) \right),
\end{aligned}$$

где α и λ задаваемые параметры, определяющие частоту и размеры (диаметры) вихрей.

В реагирующих течениях, рассматривается необратимая реакция простого типа $\mathcal{A} + \nu \mathcal{B} \rightarrow (1 + \nu) \mathcal{P}$. Скорость образования массовой доли компоненты определяется как $S_A = -KAB$, в которой K является скоростью химической реакции; A и B – массовые доли компонентов \mathcal{A} и \mathcal{B} , соответственно. Предполагается, что компоненты \mathcal{A} , \mathcal{B} , \mathcal{P} являются термодинамически идентичными, а жидкость считается калорически совершенной. Как показано на рисунке 11, реагенты A и B вводятся в верхних и нижних потоках, соответственно. Начальные значения массовых долей реагентов A и B в каждой из точек по z в трехмерном случае идентичны тем что и в двумерном случае.

Переменные течения нормированы относительно характеристических величин. Эталонными величинами в задаче развивающегося по времени слоя смешивания являются величины свободного потока обозначенные индексом r . Длина L_r , которая определяется как $L_r = \delta_v(t=0)/2$, где $\delta_v = \Delta U / |\partial \langle u \rangle / \partial y|_{\max}$ начальная толщина завихренности, $\langle u \rangle$ является осредненным по Рейнольдсу фильтрованной компонентой скорости в продольном направлении и $\Delta U = 2U_r$. Эталонные величины определяют число Рейнольдса $Re = U_r L_r / \nu$. Скорость химической реакции параметризуется числом Дамкёлера $Da = K / (U_r / L_r)$. Безразмерное время определена как $t^* = U_r t / L_r$.

Вычислительная область представляет собой квадрат и куб в двумерном и трехмерном случаях, соответственно. Размер области определяется как: $-L/2 \leq x \leq L/2$, $-L/2 \leq y \leq L/2$ и $-L/2 \leq z \leq L/2$. Длина L выбирается таким образом, что $L = 2^{n_p} \lambda_u$, где n_p является задаваемым количеством последовательных пар вихрей, и λ_u длина волны наиболее неустойчивого режима, которая соответствует длине самой крупной паре вихрей.

Граничные условия задаются следующим образом: на границах по направлениям x и z ставятся условие периодичности; на границах по направлению y ставится условие симметричности.

Проведены численные реализации с DNS и LES расчетами. Основными параметрами течения являются число Рейнольдса и число Дамкёлера. Все DG расчеты проведены на структурированной сетке с постоянным шагом интегрирования по всем пространственным координатам ($\Delta x = \Delta y = \Delta z$). DNS расчеты основаны исключительно на DG решении уравнений (1)-(4), в котором

размеры вычислительной сетки и порядок полинома Лежандра таковы, что достаточны для разрешения всех масштабов потока без необходимости моделирования подсеточных функций. Двумерный DNS расчеты проведены с $Re=400$ на сетке с разрешениями 256×256 и с порядком аппроксимации $p=3$, а трехмерные расчеты с $Re=200$ и на сетке $128 \times 128 \times 128$ и с порядком аппроксимации $p=2$.

FDF и LES-DG расчеты проведены на более грубой сетки с разрешениями 64×64 и $32 \times 32 \times 32$, для двумерного и трехмерного случаев, соответственно, с разными значениями порядка полинома p . В FDF расчетах определяются фильтрованные значения первых и вторых моментов скалярной функции путем решения уравнения переноса FDF (уравнение 19) с помощью лагранжевого метода Монте-Карло. В LES-DG расчетах определяются фильтрованных значений первых и вторых моментов скалярной функции путем решения уравнений (23) и (24). В LES-DG расчетах фильтрованные значения химического источника рассчитываются исключительно с помощью MC. В обоих расчетах, фильтрованные значения скорости и подсеточной вязкости берутся из DG решений уравнений (6)-(7).

В FDF расчетах в начальный момент времени, вычислительная область заполняется частицами MC с равномерным распределением. Количество частиц в каждой ячейке ансамбля равна 16. Ячейка ансамбля находится у каждой квадратурной точки. Количество квадратурных точек в элементе зависит от порядка полинома, и определяется как $N_q = (p + 1)$ в каждом направлении. Для $p=3$, в каждом элементе DG имеется 64 квадратурных точек, соответственно в каждом элементе будут содержаться $N_p=32*16=512$ частиц MC.

Периодичность течения в продольном направлении обеспечивается введением частицы на левой границе со значениями частицы, покидающей область на правой границе и наоборот. В трехмерных расчетах аналогичная процедура используется в направлении z . Зеркальная симметрия течения компонентов на верхней и нижней границах частицы, покинувшие верхнюю или нижнюю границы, возвращаются в область на противоположной границе, а значения берутся таким образом, если, например, частица относится к компоненте A , то присваивается значение компоненты B , и наоборот.

В реагирующих течениях, моделирование проводится с относительно медленными ($Da = 10^{-2}$), умеренными ($Da = 1$) и относительно быстрыми ($Da = 10^2$) реакциями.

5 ВЫСОКОПРОИЗВОДИТЕЛЬНЫЙ DG-МС-LES СИМУЛЯТОР

5.1 Примеры применения

В этом разделе представлены результаты расчетов, которые отражают согласованность, сходимость и прогностические способности разработанной численной методологии DG-МС. Расчеты проведены с использованием тестовой задачи, где моделируется развивающийся по времени слой смешения. Задачи рассмотрены в двумерных и трехмерных случаях.

Для определения согласованности и сходимости решателя МС, рассматриваются обобщенные первые моменты подсеточного масштаба $\langle \phi_\alpha \rangle_L$ и подсеточная дисперсия $\tau(\phi, \phi) = \langle \phi_{(\alpha)}^2 \rangle_L - \langle \phi_\alpha \rangle_L^2$, которые получаются путем интегрирования смоделированного уравнения переноса FDF (22):

$$\frac{\partial \langle \phi_\alpha \rangle_L}{\partial t} + \frac{\partial \langle \phi_\alpha \rangle_L \langle u_j \rangle_L}{\partial x_j} = \frac{\partial}{\partial x_j} \left[(\Gamma + \Gamma_t) \frac{\partial \langle \phi_\alpha \rangle_L}{\partial x_j} \right] + \langle S_\alpha \rangle_L \quad (43)$$

$$\begin{aligned} \frac{\partial \tau(\phi, \phi)}{\partial t} + \frac{\partial \langle u_j \rangle_L \tau(\phi, \phi)}{\partial x_j} &= \frac{\partial}{\partial x_j} \left[(\Gamma + \Gamma_t) \frac{\partial \tau(\phi, \phi)}{\partial x_j} \right] - 2\Omega_m \tau(\phi, \phi) \\ &+ 2(\Gamma + \Gamma_t) \left[\frac{\partial \langle \phi_\alpha \rangle_L}{\partial x_j} \frac{\partial \langle \phi_\alpha \rangle_L}{\partial x_j} \right] + 2(\langle \phi_\alpha S_\alpha \rangle_L - \langle \phi_\alpha \rangle_L \langle S_\alpha \rangle_L) \end{aligned} \quad (44)$$

Общая согласованность гибридной схемы DG-МС, лучше всего достигается путем сравнения значений этих двух первых моментов скалярной функции, полученные с помощью FDF, с моментами, полученные непосредственно через DG на той же сетке. Значения химического источника рассчитываются исключительно с помощью МС, а затем используются в решении DG отфильтрованных скалярных уравнений.

Смоделированные результаты анализируются мгновенно и статистически. В первом случае отображаются контуры моментальных снимков и графики рассеяния скалярной функции. Статистические данные построены путем пространственного усреднения по x и z направлениям. Усредненные переменные обозначены чертой и определяются следующим образом: $\bar{\tau}(a, b) = \overline{\langle ab \rangle} - \langle a \rangle \langle b \rangle$ означает усредненную подсеточную дисперсию, $\bar{R}(a, b) = \overline{\langle a \rangle \langle b \rangle} - \langle a \rangle \langle b \rangle$ означает разрешенную дисперсию $\bar{r}(a, b) = \overline{ab} - \bar{a} \bar{b} = \bar{R}(a, b) + \bar{\tau}(a, b)$ означает полную дисперсию. Расчет этих переменных дает возможность определить какая часть энергии разрешается напрямую, а какая моделируется. С точки зрения метода моделирования с помощью крупных вихрей, $\bar{R}(a, b)$, $\bar{\tau}(a, b)$ и $\bar{r}(a, b)$ относятся к энергии крупных вихрей, мелких вихрей и общей энергии, соответственно. Для

подтверждения согласованности LES-FDF расчетов, их результаты сравниваются с результатами DNS расчетов. DNS расчеты проводятся на более густой сетке, необходимой для захвата всех размеров вихрей начиная с колмогоровского масштаба.

5.1.1 Двумерный развивающийся по времени слой смешения

В этом подразделе представлены результаты расчетов, полученные путем моделирования двумерного слоя смешения. Контур завихренности ($\omega = \partial v / \partial x - \partial u / \partial y$) с наложением вектора скорости в разный момент времени представлены на рисунке 12. Здесь можно наблюдать как развиваются по времени наложенные в начальный момент возмущения.

На рисунке 13 показаны мгновенные контурные графики отфильтрованной скалярной функции $\langle \phi_\alpha \rangle_L$ полученные с помощью DG и MC. Результаты расчетов демонстрирует эффективность моделирования с помощью FDF, поскольку результаты MC наглядно согласуются с результатами, полученными с помощью DG. Это подтверждается количественно графиками рассеяния мгновенных отфильтрованных значений на рисунке 14.

Более строгий способ оценки результатов FDF – рассмотрение результатов, усредненных по Рейнольдсу. На рисунке 15 показано сравнение усредненных по Рейнольдсу первых моментов ($\overline{\langle \phi_\alpha \rangle_L}$) полученные с помощью MC с результатам полученные с помощью DG для $p=2$ и $p=3$. В целом результаты FDF и LES-DG находятся в хорошем согласии. Повышение порядка аппроксимации p не значительно влияет на точность решения. Тем не менее, изменение порядка при расчете подсеточной и разрешенной дисперсии. Как показано на рисунках 16 и 17, с повышением порядка объем подсеточной дисперсии уменьшается, с другой стороны объем разрешенной дисперсии повышается. Эти результаты хорошо согласуются с основной идеей метода моделирования крупных вихрей. С повышением порядка точности и/или разрешения сетки, уменьшается размер фильтра G (уравнение (5)), тем самым разрешается большая часть энергии. Объем полной дисперсии, как отражено на рисунке 18, остается неизменной для $p=2$ и $p=3$.

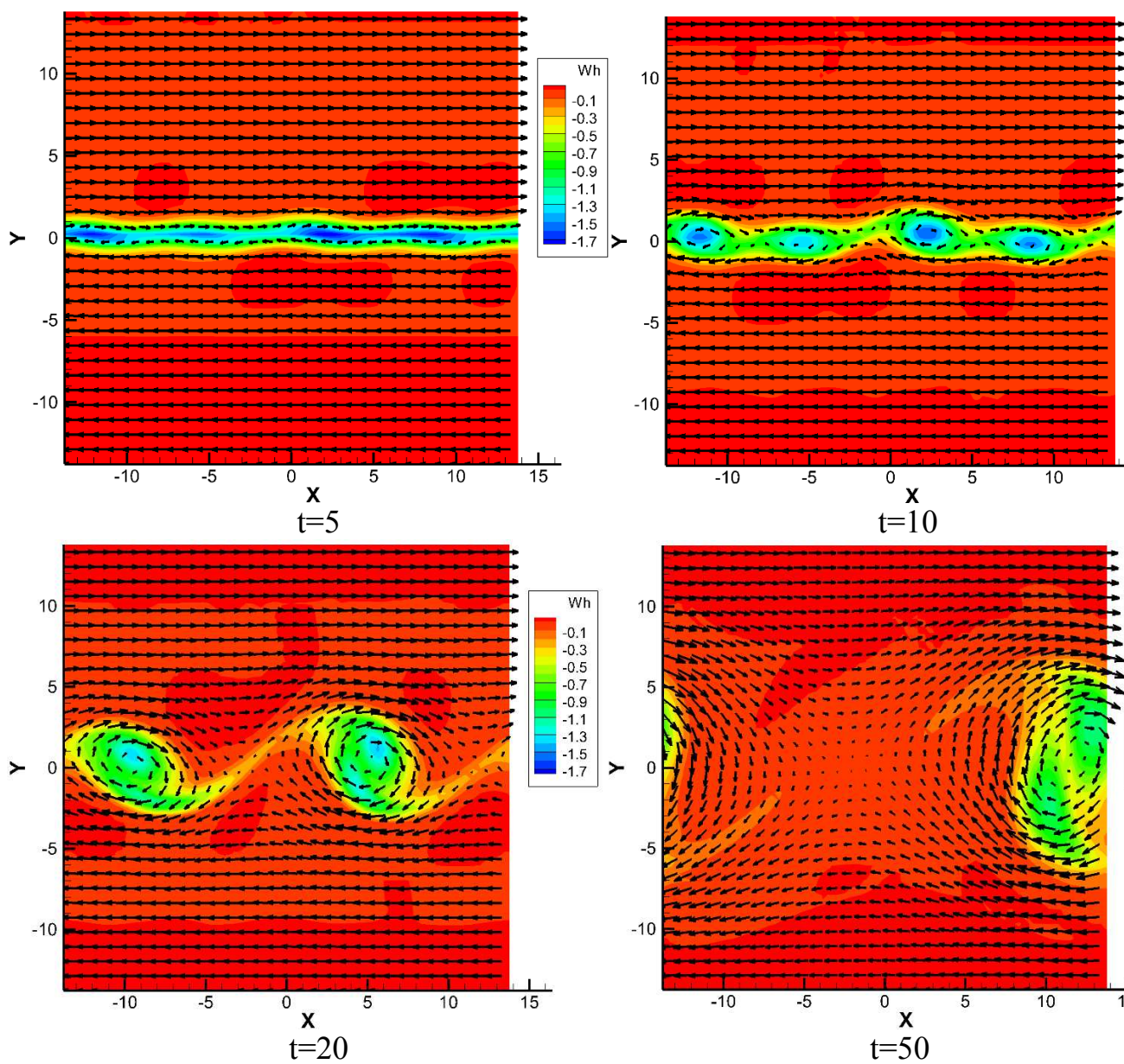


Рисунок 12 – Поле завихренности с наложением вектора скорости в разные моменты времени

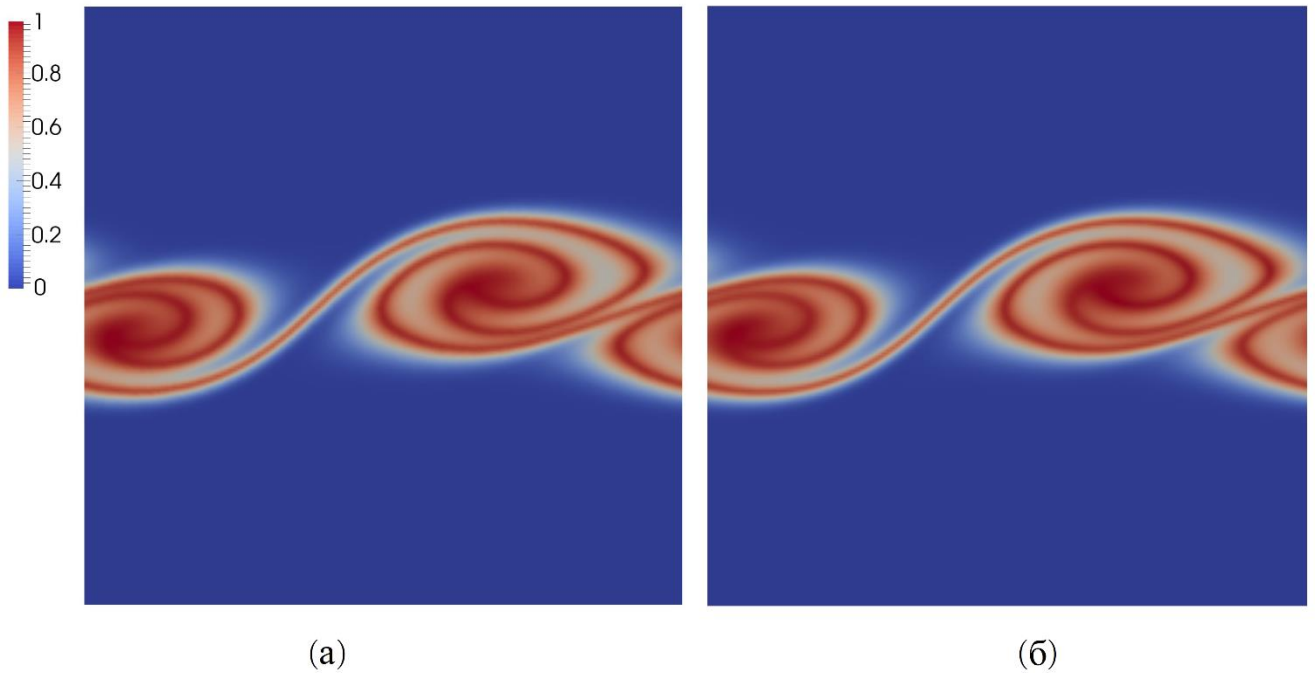


Рисунок 13 – Контуры распределения отфильтрованной скалярной функции $\langle \phi_P \rangle_L$ при $p = 3$ и $t = 25$. (а) DG, (б) MC

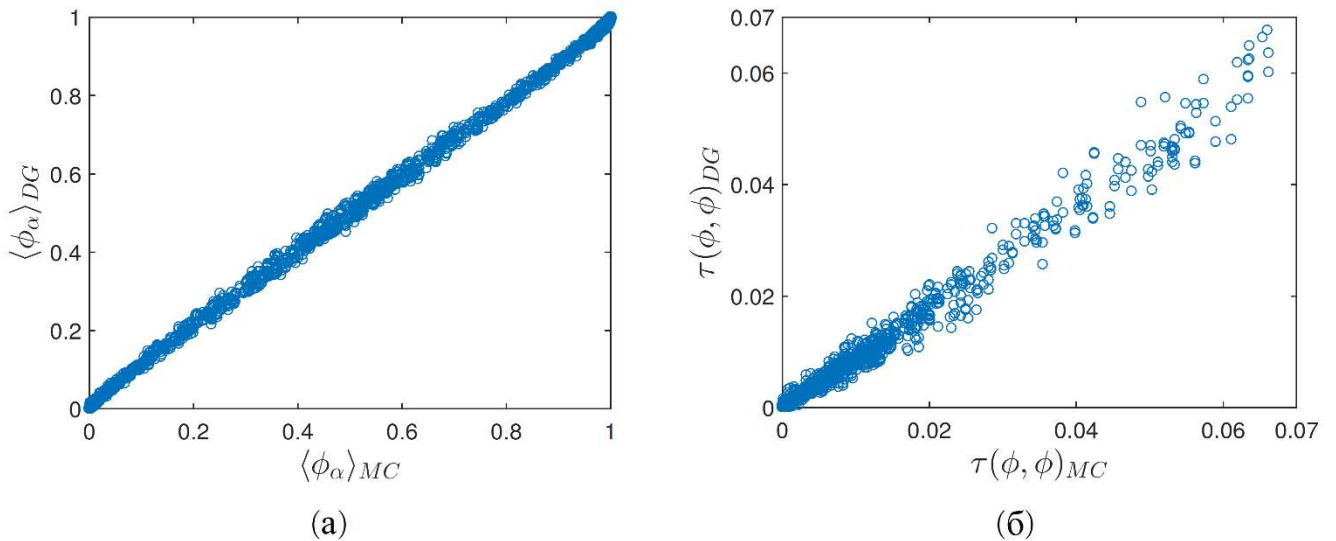


Рисунок 14 – Диаграммы рассеяния между отфильтрованными значениями, полученные с помощью MC и DG при $t = 28$. (а) отфильтрованная скалярная функция, где коэффициент корреляции составляет 0.99967. (б) момент SGS второго порядка, коэффициент корреляции составляет 0.99302

В случае присутствия химической реакции, согласованность FDF расчетов и надежность смоделированных результатов исследуются путем рассмотрения массовых долей частиц в области “фракции смеси” ξ (mixture fraction) [73]. Как

показано на рисунке 19, когда скорость химической реакции относительно мала ($Da=10^{-2}$), значения близки к значениям для случая только перемешивания. Для случая с умеренной скоростью химической реакции ($Da = 10$), композиционная структура аналогична структуре распределенной зоны реакции. Для более высоких скоростей реакции ($Da=10^2$) состав близок к составу бесконечно быстрой реакции.

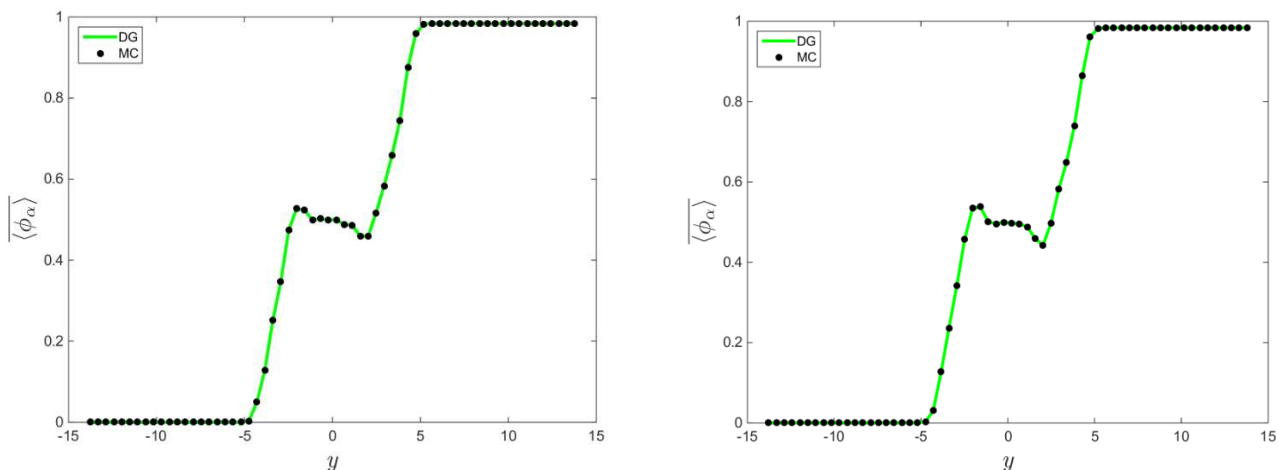


Рисунок 15 – Поперечное изменение осредненных по Рейнольдсу скалярной функции. Слева значения для $p=2$, справа для $p=3$

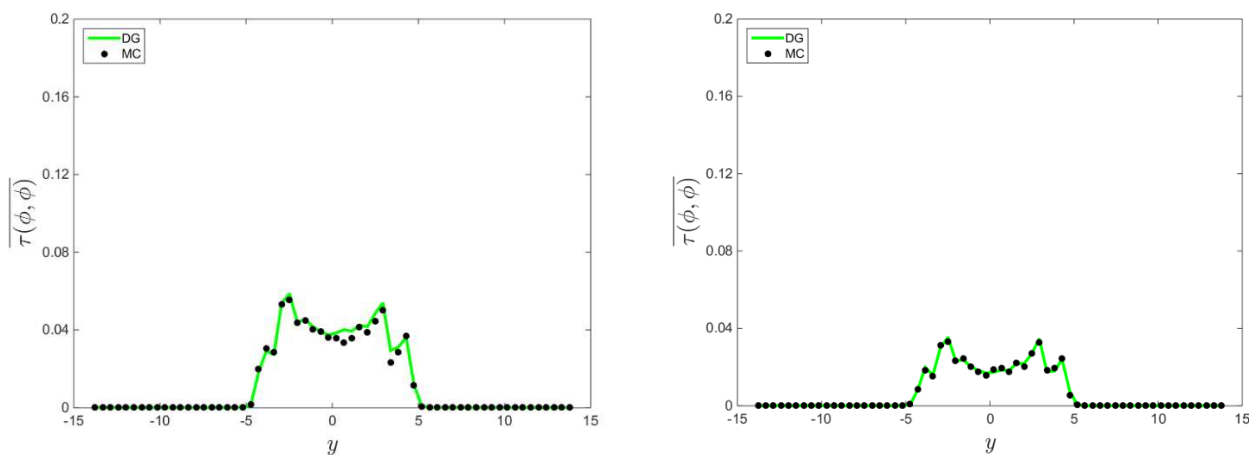


Рисунок 16 – Поперечное изменение осредненных по Рейнольдсу подсеточной дисперсии. Слева значения для $p=2$, справа для $p=3$

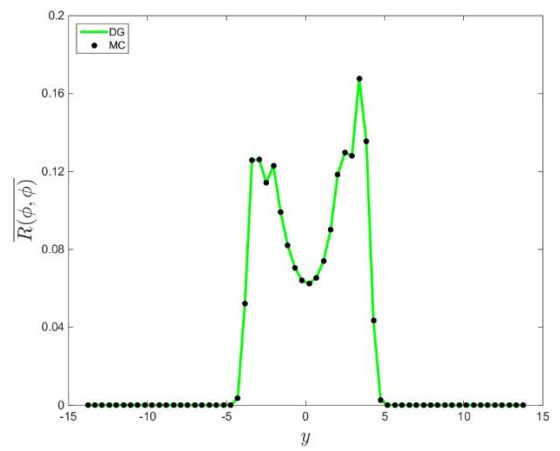
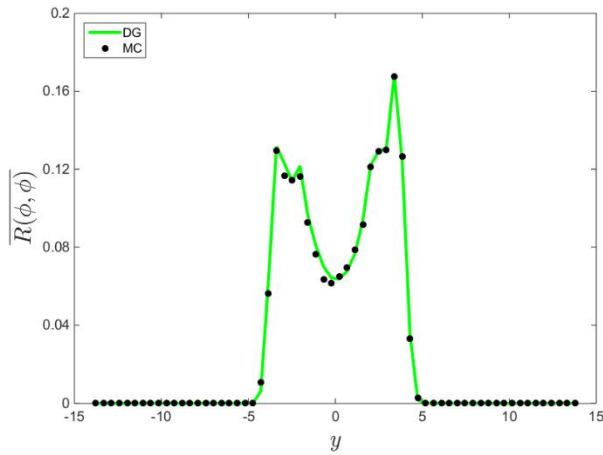


Рисунок 17 – Поперечное изменение осредненных по Рейнольдсу разрешенных дисперсии. Слева значения для $p=2$, справа для $p=3$.

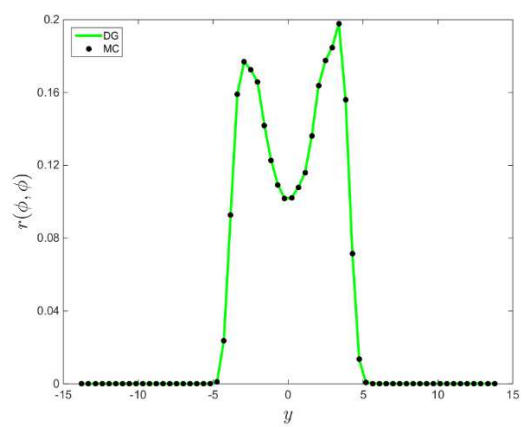
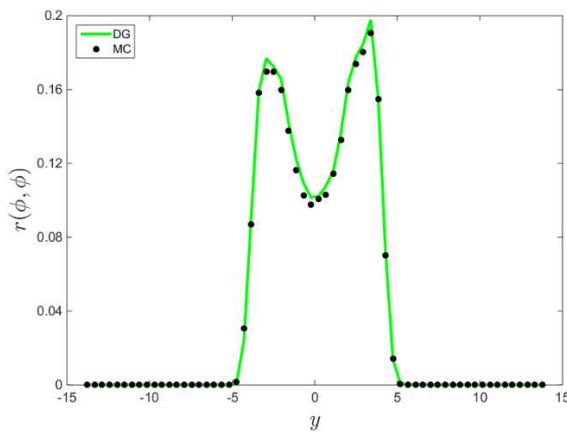


Рисунок 18 – Поперечное изменение осредненных по Рейнольдсу полных дисперсии. Слева значения для $p=2$, справа для $p=3$.

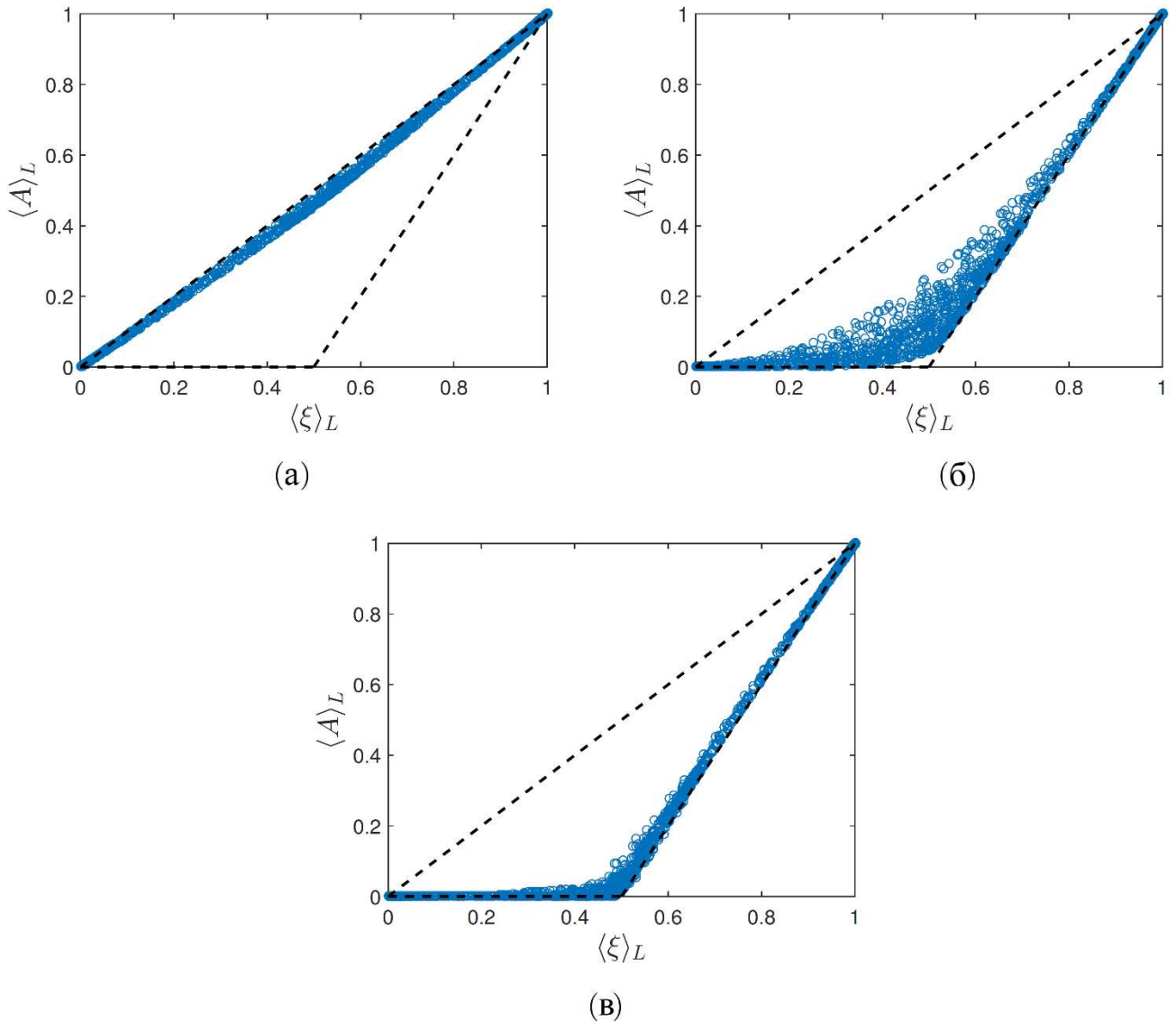


Рисунок 19 – Диаграммы рассеяния между массовыми долями и переменной Шваба-Зельдовича для (а) $Da = 10^{-2}$, (б) $Da = 1$ и (в) $Da = 10^2$. Пунктирные линии обозначают пределы чистого перемешивания без химической реакции и пределы бесконечно быстрой химической реакции

5.1.2 Трехмерный развивающийся по времени слой смешения

На этом подразделе представлены результаты моделирования трехмерного слоя смешения. Гидродинамические параметры течения получаются из системы фильтрованных уравнений (6)-(7) с начальными и граничными условиями. Фильтрованные значения скаляра получаются двумя способами: решение уравнения скаляра (23) с помощью метода DG; решение уравнения переноса FDF (19) с помощью метода MC. В случае присутствия химической реакции фильтрованный источник $\langle S_\alpha \rangle_L$ в уравнении (23) вычисляется с помощью MC.

На рисунках 20-22 представлены результаты прямого численного моделирования для $Re=200$ при $t = 102$. Вычисления проводились на сетке с разрешением $128 \times 128 \times 128$ и с порядком полинома $p = 2$. Результаты представлены в виде векторных полей скоростей и линии тока во всех трех плоскостях. Векторное поле и линии тока визуально демонстрируют, что движение является трехмерным. Также увеличив векторное поле скоростей, можно заметить вихревую структуру течения во всех трех направлениях пространства.

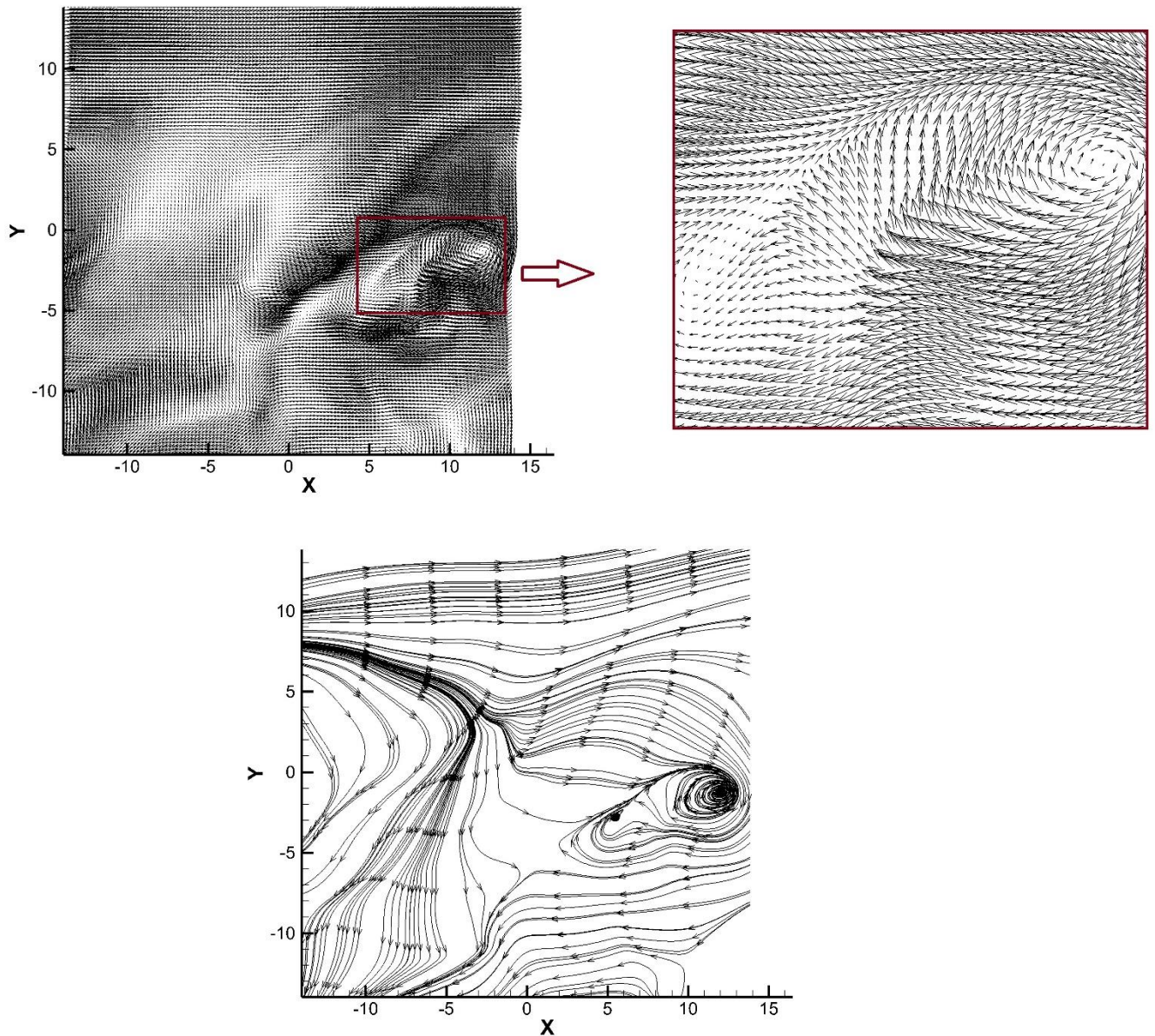


Рисунок 20 – Векторное поле скоростей и линии тока в сечении плоскости x-y при $t=102$

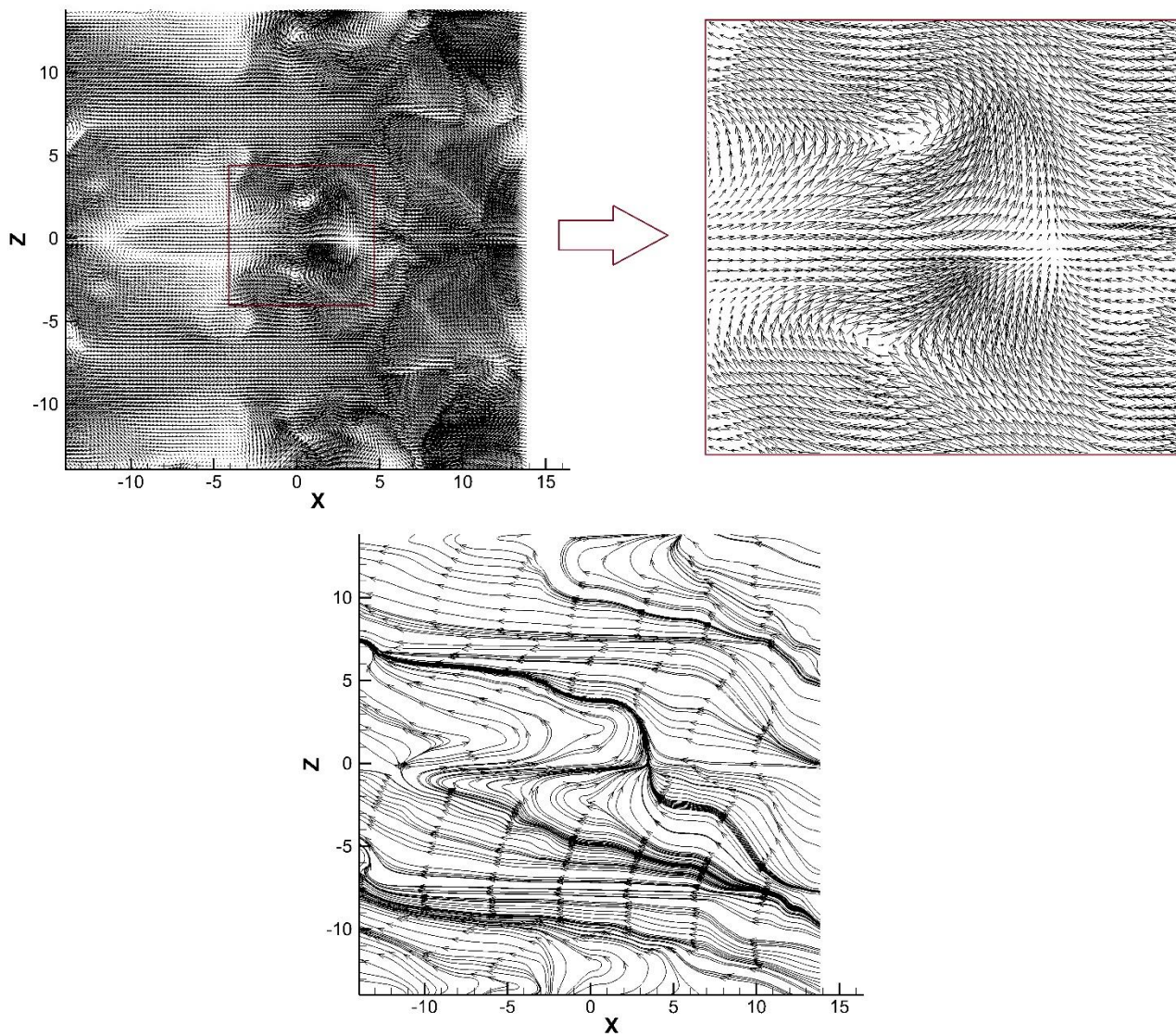


Рисунок 21 – Векторное поле скоростей и линии тока в сечении плоскости $x-z$ при $t=102$

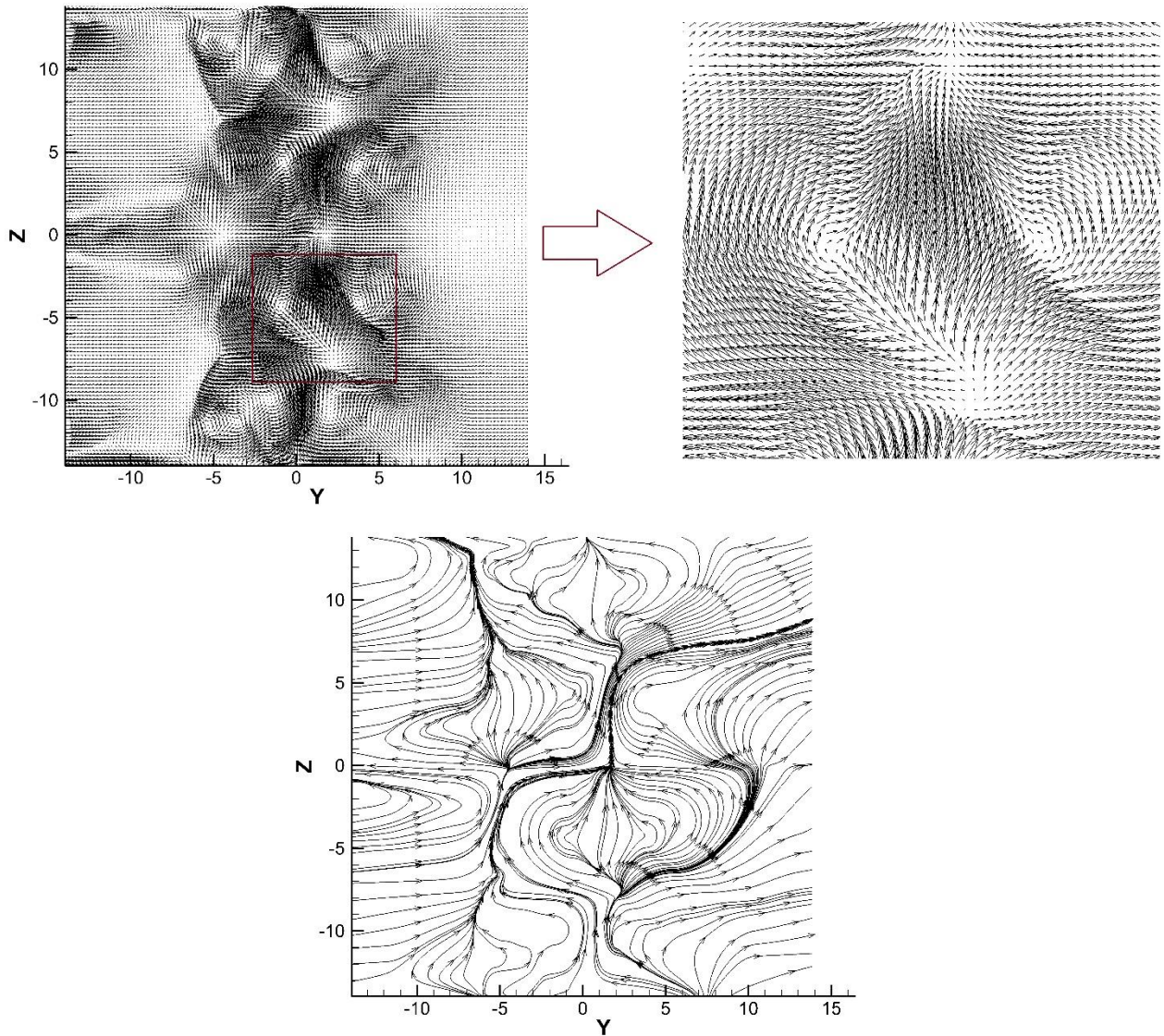


Рисунок 22 – Векторное поле скоростей и линии тока в сечении плоскости y - z при $t=102$

На рисунке 23 представлены контуры распределения скалярной функции ϕ_α в разных сечениях в плоскостях x - y , y - z и x - z при $t=102$. Результаты демонстрируют что смешивание имеет место во всех трех направлениях, что также показывает трехмерность течения. Когерентная структура течения наглядно выражена с помощью вихрей относительно малых и крупных размеров. Цвет контура соответствует значению ϕ_α , где красный цвет является максимальным значением равному 1, и темно-синий является минимальным значением равному 0. Промежуточные значения можно посмотреть на индикаторе, изображенный на первом рисунке 23.

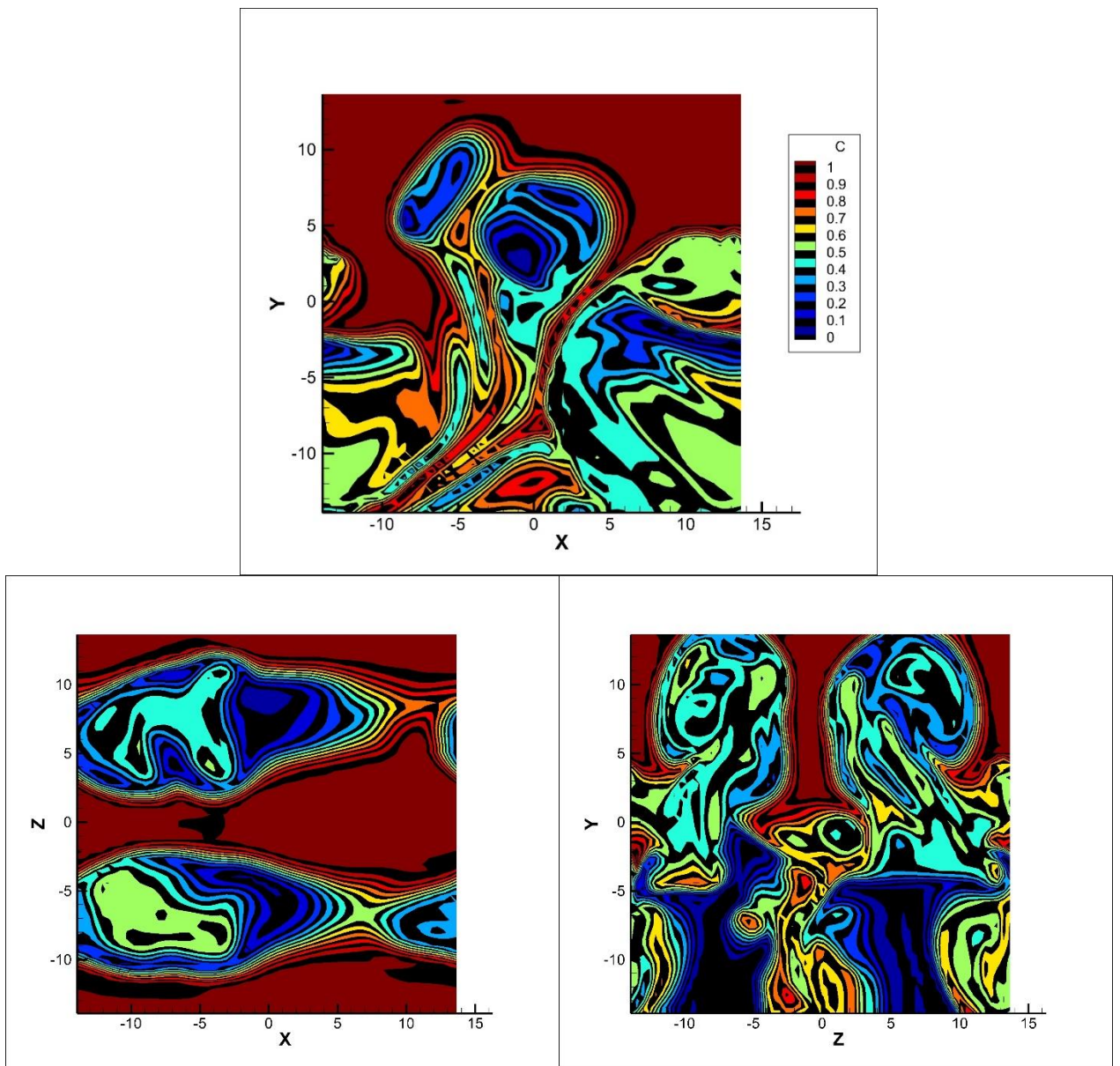


Рисунок 23 – Контуры распределения фильтрованных значений скалярной функции ϕ_A в плоскостях x-y, y-z и x-z при $t=102$

Общая согласованность симулятора, как и в двумерном случае, лучше всего достигается путем сравнения первых двух моментов, полученных из FDF с помощью MC, с моментами, моделируемыми непосредственно через DG на той же сетке. На рисунках 32 и 25 сравниваются контуры и изоповерхности отфильтрованных значений первого фильтрованного момента ($\langle \phi_\alpha \rangle_L$) полученные с помощью MC с результатами полученные с помощью DG. Эти рисунки наглядно демонстрируют согласованность моделирования FDF, поскольку результаты MC согласуются с результатами, полученными в DG. Здесь также можно заметить, как

мы уже утверждали ранее, по сравнению с DG, результаты MC не имеют численную диффузию.

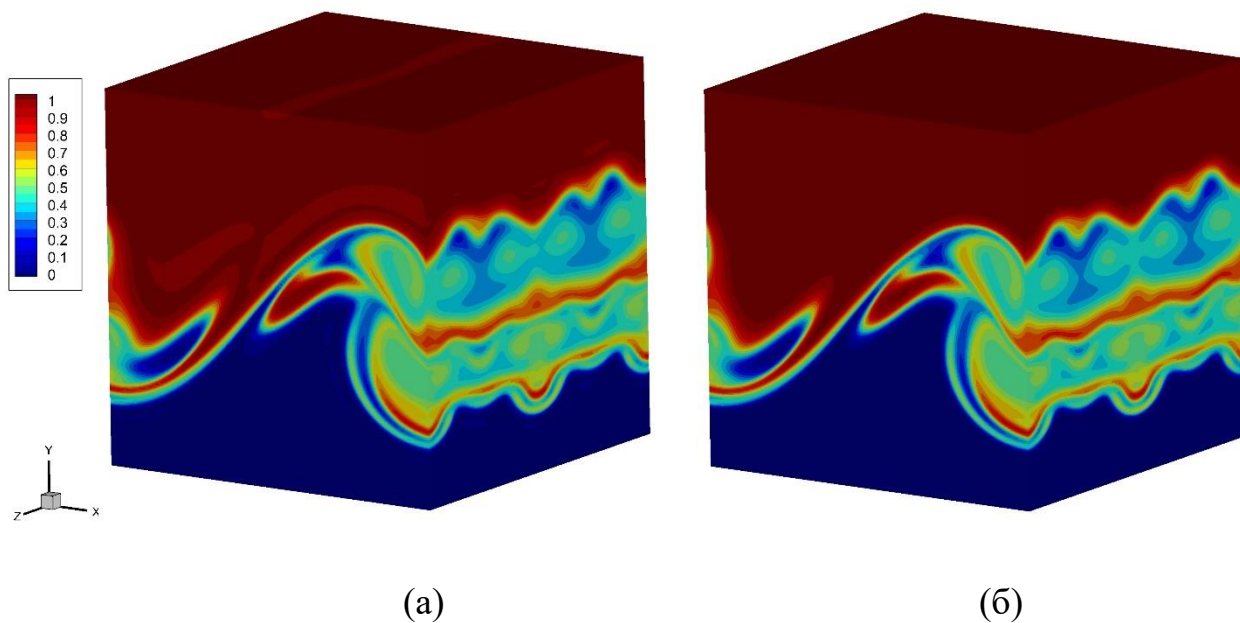


Рисунок 24 – Контуры распределения отфильтрованных значений скалярной функции $\langle \phi_A \rangle_L$ при $p=2$ и $t=45$. (а) DG, (б) MC

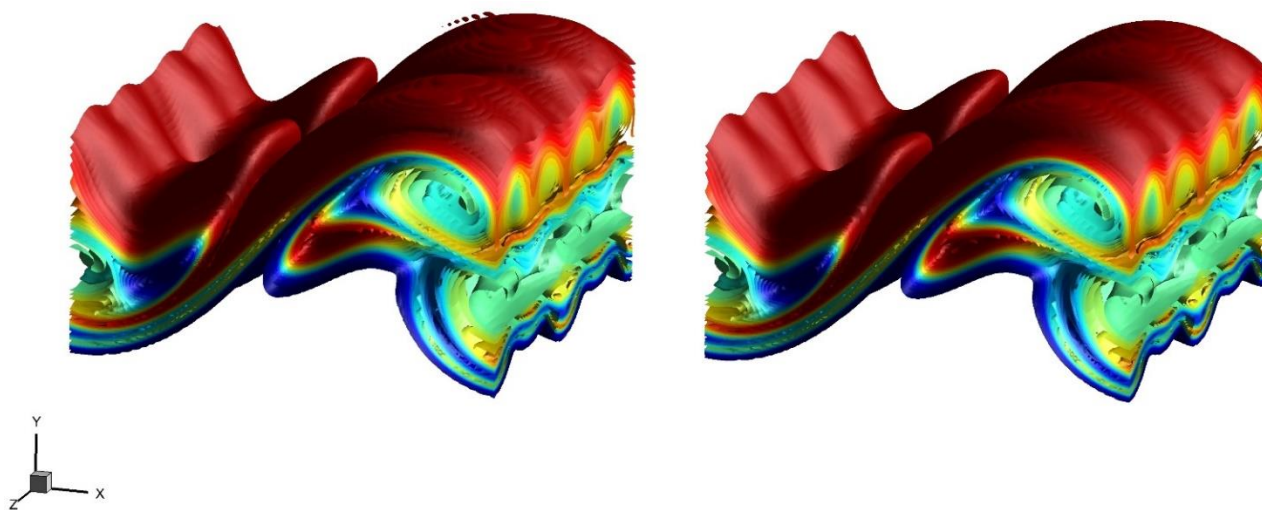


Рисунок 25 – Изоповерхности отфильтрованных значений скалярной функции $\langle \phi_A \rangle_L$ при $p=2$ и $t=45$. (а) DG, (б) MC

Количественное сравнение первых двух моментов полученные при помощи DG и MC представлены на рисунке 26 через корреляционные диаграммы. Для обеих моментов, коэффициент корреляции очень близок к единице, что говорит о высокой

согласованности FDF и LES-DG расчетов. Стоит учесть, что такую согласованность для вторых моментов трудно добиться с использованием FD методов. При реагирующих течениях, члены химического источника вычисляются исключительно через MC и затем используются в DG путем интерполирования.

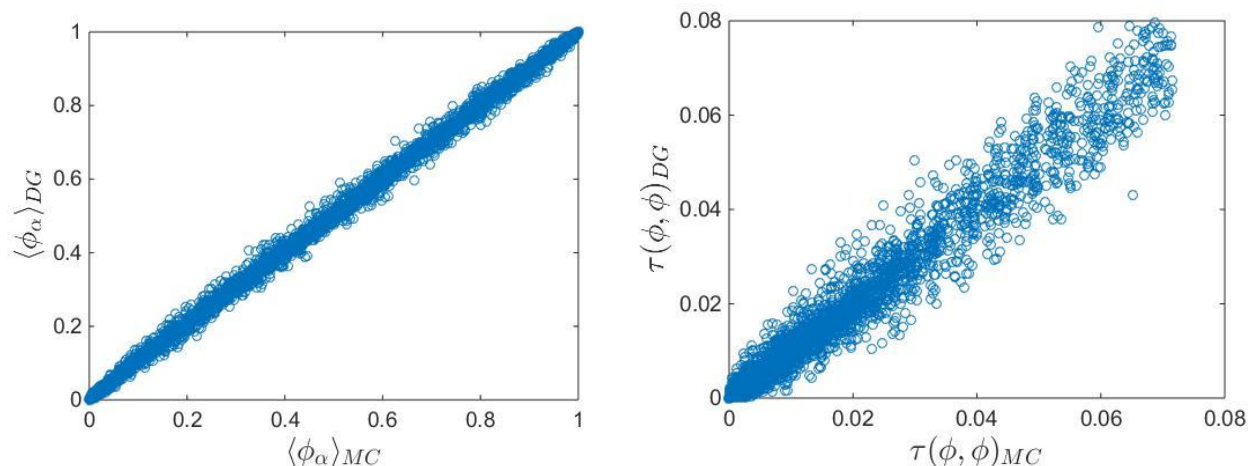


Рисунок 26 – Консистенция отфильтрованных значений MC при NPQ=32 и DG с $p = 2$ при $t = 40$. (а) отфильтрованная скалярная функция, коэффициент корреляции составляет 0.9997. (б) дисперсия подсеточного масштаба, коэффициент корреляции составляет 0.9921

На рисунке приведено количественное сравнение осредненной скалярной функции ($\langle \phi_\alpha \rangle_L$) с результатами полученные в работе [28]. Результаты получены с учетом химической реакции. В этой работе, как и в нашей, значения химического источника рассчитываются исключительно с помощью MC, а затем используются в решении DG. По корреляционной диаграмме можно наглядно увидеть, что результаты сходятся с высокой точностью; коэффициент корреляции составляет 0.9998.

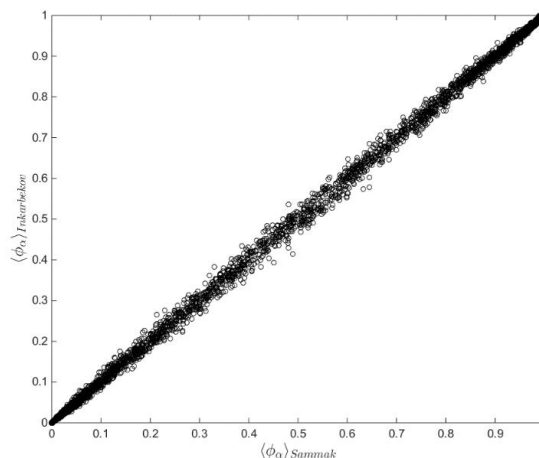


Рисунок 27 – Сравнение с результатами [28]

Расчеты осредненных по Рейнольдсу подсеточной дисперсии, разрешенной дисперсии и полной дисперсии приведены на рисунке 28. Анализ результатов приводит к выводу, что около 78% полной энергии разрешается напрямую, а оставшая часть моделируется. Полная энергия течения определяется с высокой точностью даже при значении порядка полинома аппроксимации $p=2$. Этому свидетельствует третий рисунок, где сравниваются результаты, полученные с помощью FDF-MC и LES-DG с результатами DNS. Такое свойство схемы является особенно притягательной, когда прогноз развития полной энергии имеет первостепенное значение.

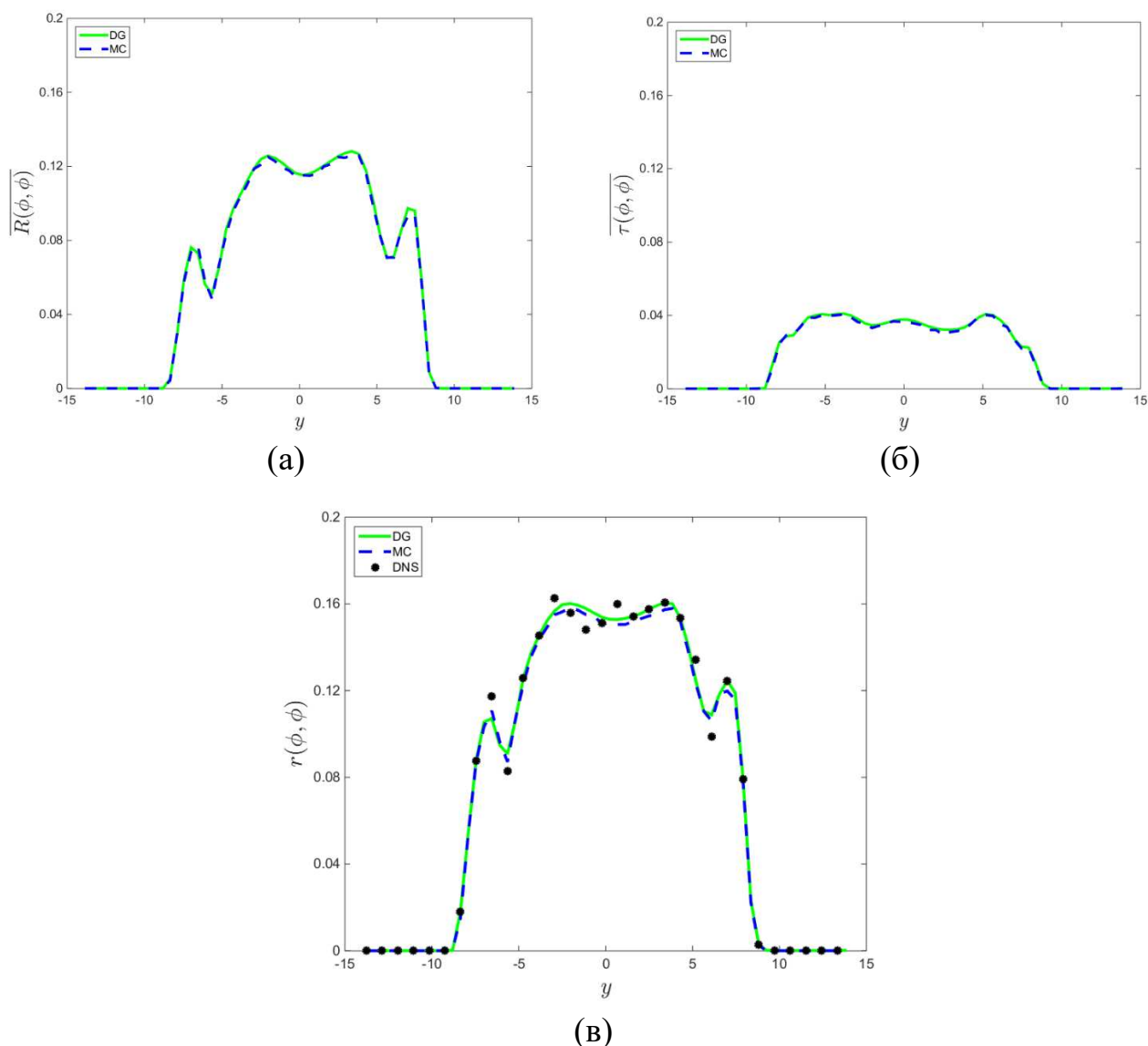


Рисунок 28 – Поперечное изменение усредненных значений по Рейнольдсу для $p = 2$ и $NPQ = 64$. (а) SGS скалярная дисперсия, (б) разрешенная скалярная дисперсия и (в) полная дисперсия

В случае отсутствия химической реакции, согласованность FDF расчетов и надежность смоделированных результатов, как и в двумерном случае, исследуются путем рассмотрения массовых долей частиц в области “массовой доли смеси” ξ . Результаты расчетов приведены на рисунке 29. При относительно малых значениях скорости химической реакции ($Da=10^{-2}$), значения близки к значениям для случая только перемешивания. При умеренной скорости химической реакции ($Da = 10$), композиционная структура аналогична структуре распределенной зоны реакции. Для более высоких скоростей реакции ($Da=10^2$) состав близок к составу бесконечно быстрой реакции.

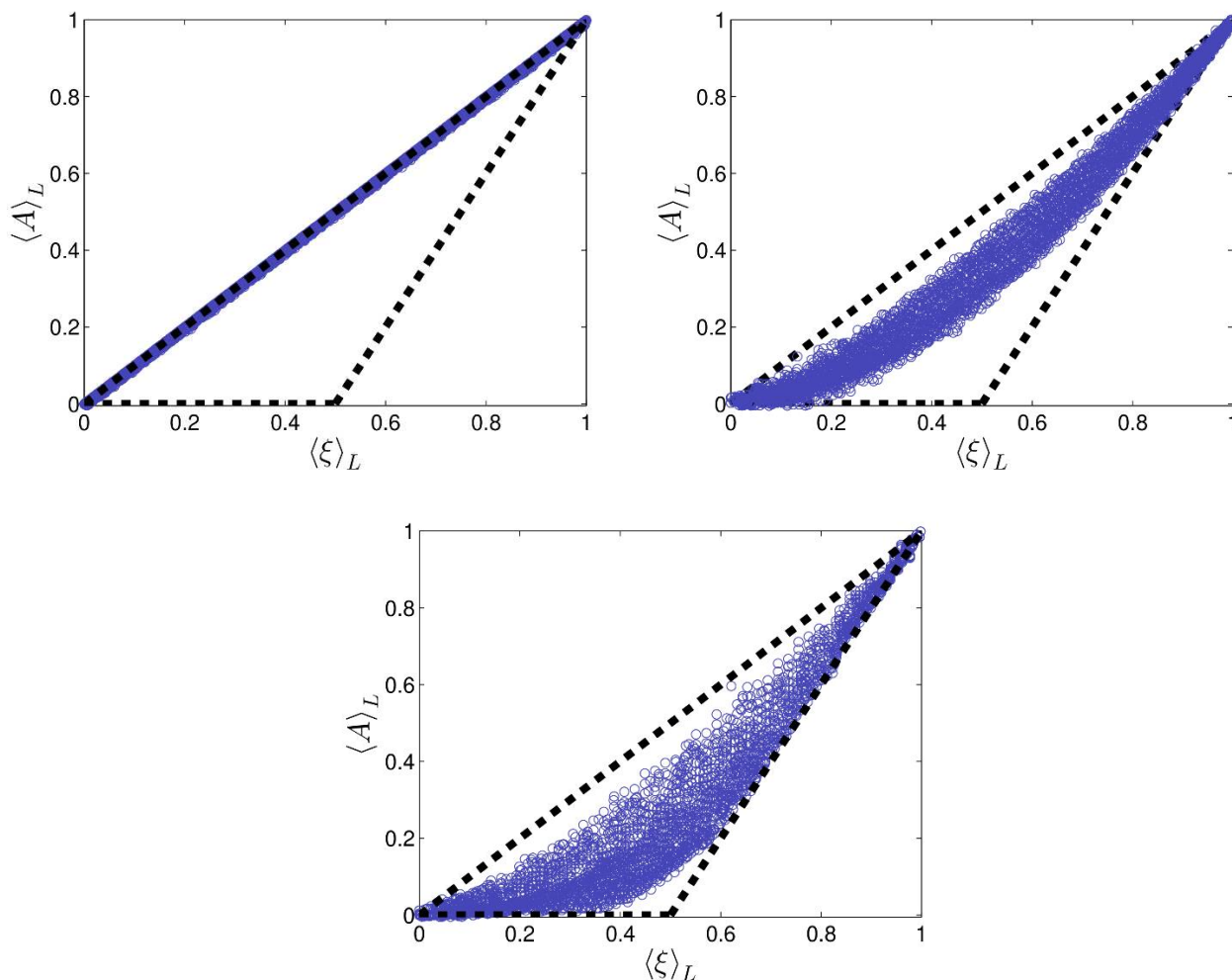


Рисунок 29 – Диаграммы рассеяния между массовыми долями и переменной Шваба-Зельдовича для (а) $Da = 10^{-2}$, (б) $Da = 1$ и (в) $Da = 10^2$. Пунктирные линии обозначают пределы чистого перемешивания без химической реакции и пределы бесконечно быстрой химической реакции

5.1 Вычислительная производительность

В этом подразделе демонстрируются вычислительные возможности параллельного решателя DG-МС. Результаты двумерных тестов (таблицы 5-6,

рисунки 30-36), полученные с помощью параллельных вычислений на GPU, оцениваются путем сравнения с результатами последовательных вычислений на CPU. Для этого соблюдается, чтобы количество выполняемых арифметических операций, как и в параллельном коде, так и в последовательном коде были одинаковы. Для трехмерных вычислений (рисунки 37-38), производительность параллельного алгоритма оценивается путем вычислений среднего количества операций с плавающей запятой в секунду (FLOPS).

Таблица 5 – Вычислительная производительность решателя MC при фиксированных $N_e=256^2$ и $NPQ=32$

p	CPU Runtime	GPU Runtime	Ускорение
1	3.31	0.049	67.24
2	7.12	0.090	78.91
3	12.17	0.099	123.28
4	20.49	0.149	143.86

В Таблице 5 показаны рабочие характеристики MC для различных полиномиальных степеней при фиксированном количестве элементов и количестве частиц в квадратурной точке. К счастью, с увеличением полиномиального порядка число частиц увеличивается, что приводит к увеличению арифметической интенсивности, и, следовательно, увеличивается коэффициент ускорения. Как упоминалось ранее, процедура MC состоит из пяти этапов, каждый из которых выполняет различные виды операций. Анализ производительности этих шагов MC для CPU и GPU представлен в Таблице 6. Здесь во втором и четвертом столбцах представлены времена работы CPU и GPU, в третьем и пятом столбцах представлен относительный процент работы каждого этапа в CPU и GPU, а в последнем столбце представляет скорость графического процессора по сравнению с реализацией процессора. Как указано, этапы интерполяции, интегрирования и усреднения показывают наибольшие коэффициенты ускорения, в то время как этап подсчета частиц показывает наименьшее ускорение. Причина низкой производительности этапа подсчета частиц может быть объяснена тем фактом, что это своего рода операция сокращения, когда потоки в пределах одной основы получают доступ к данным, охватывающим многие сегменты памяти. К счастью, эта часть вычислений занимает всего 14% от общего вычислительного времени и, следовательно, незначительно влияет на общую производительность. Визуальная иллюстрация фактора ускорения каждой стадии и доля распределения времени в модуле Монте Карло, показано на рисунке 30 с помощью графика и круговую диаграмму, соответственно

Таблица 6 – Время вычисления каждой стадии МС при $p = 4$ для двумерного моделирования

Стадия МС	CPU Runtime	%	GPU Runtime	%	Speedup
Интерполяция	12.29	59.98	0.065	48.15	179.74
Генерация случайных чисел	2.73	13.30	0.027	18.52	103.81
Интегрирование	2.36	11.54	0.015	10.49	158.43
Подсчет частиц	1.01	4.93	0.308	14.20	50.09
Осреднение	2.10	10.25	0.012	8.64	171.17

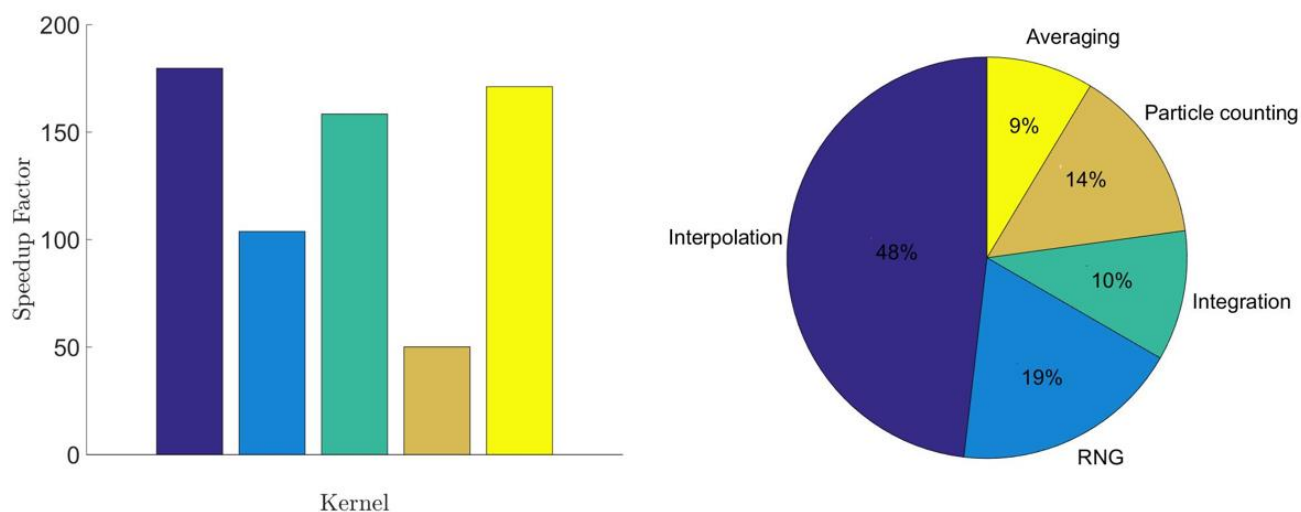


Рисунок 30 – Распределение времени в модуле МС

На рисунках 31-34 показаны сравнения производительности GPU и CPU для разных размеров сетки и полиномиальных степеней. Наблюдается, что общий коэффициент ускорения для комбинированных МС и DG увеличивается с увеличением количества элементов. Рассматривая только часть DG, можно заметить, что ее коэффициент ускорения увеличивается с увеличением количества элементов для всех p . Однако наклон увеличения вначале крутой, а затем плавно достигает своего плато. Для $p = 1$ плато достигается после $N_e = 512^2$, и в то же время коэффициент ускорения составляет около 80. Для $p = 2$ и $p = 4$ плато ускорения достигаются при $N_e = 256^2$, а коэффициент ускорения составляет 122 и 153 соответственно. Следует отметить, что из-за ограничений памяти графического процессора максимально возможный размер ячейки для $p = 4$ ограничен 384^2 элементами. Для МС верхнее значение для коэффициента ускорения для $p = 1$ достигается при $N_e = 512^2$, а для остальных $p = 2$ и $p = 4$ при $N_e = 256^2$. После этих N_e факторы ускорения слегка уменьшаются для всех p . Следует отметить, что из-за ограничений глобальной памяти GPU, наибольшим размером проблемы, который

мы могли бы рассмотреть, является конфигурация потока с $N_e = 384^2$ элементами, $p = 4$ и $NPQ = N_e * N_q^2 * NPQ = 75,497,472$ общего числа частиц. Глобальное распределение памяти в этом случае составляет 18,3% и 81,7% для DG и MC соответственно. Процент распределения памяти в DG и MC по сравнению с полиномиальным порядком приведен на рисунке 31.

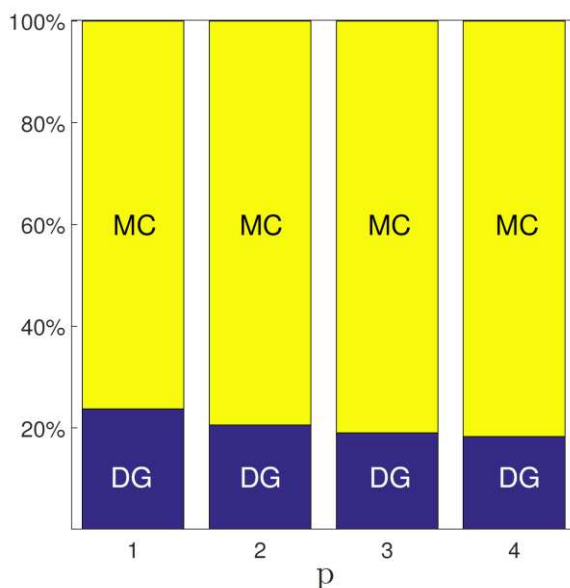


Рисунок 31 – Процент распределения памяти в DG и MC в зависимости от полиномиального порядка при фиксированном $N_e = 384^2$ и $NPQ = 16$

На рисунке 35 показано изменение коэффициента ускорения относительно степени p полинома для MC, DG и общего решателя. Наблюдается, что, когда значение p выше, значение факторов ускорения увеличивается. Коэффициент ускорения является неоднородным, и максимальное значение достигается при $p = 4$, в то время как коэффициент ускорения MC монотонно увеличивается, пока $p = 6$. Максимальный коэффициент ускорения равным 156 достигается в DG, а коэффициент ускорения равным 212 достигается в расчетах MC. На рисунке 36 показана арифметическая пропускная способность для решателя GPU с несколькими размерами ячеек при $p = 3$ и $NPQ = 16$ с двойной точностью. Здесь результаты показаны для операций с плавающей запятой в секунду (FLOPS), учитывая, что теоретическая пиковая производительность устройства с графическим процессором составляет 354 GFLOPS [69]. Как показано, для данных вычислительных параметров достигается 215 GFLOPS, что составляет 60,7% от общей арифметической пропускной способности устройства.

Далее представлены производительности параллельных алгоритмов моделирования трехмерных задач. На рисунке 37 отражены результаты тестов, где размер сетки N_e и количество частиц у каждой квадратурной точки NPQ зафиксированы, а значение порядка полинома p изменяется от 1 до 4. Наблюдается,

что, когда значение p выше, производительность алгоритма увеличивается монотонно. Среднее количество выполняемых операций GFLOPs является неоднородным для DG и MC. Максимальное значение GFLOPs для DG имеет место при $p = 3$ и составляет 203.08, в то же время, для MC максимум зафиксирован при $p = 4$ и составляет 255.11. Можно также заметить, что плато общей производительности достигается после $p = 3$.

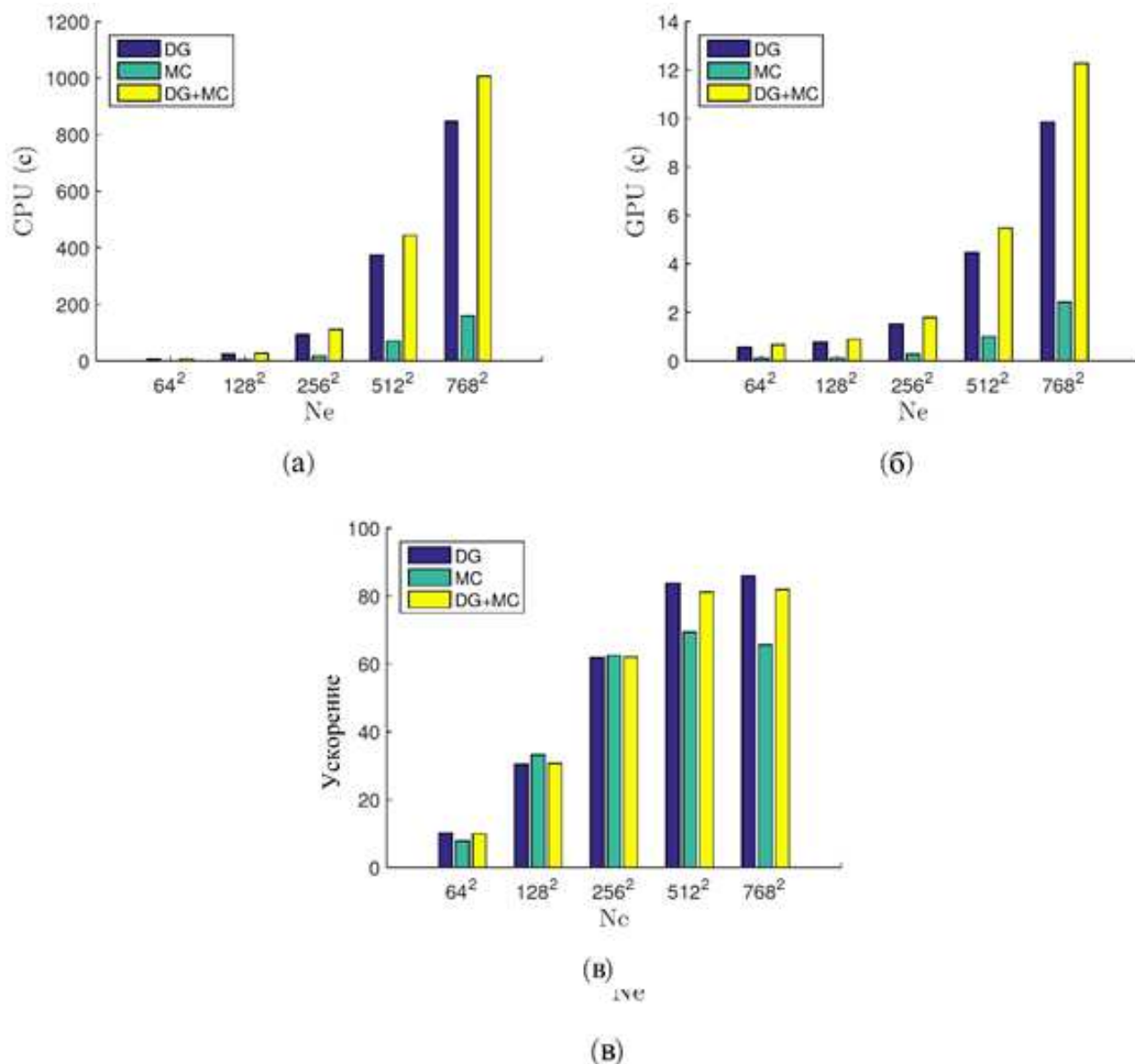
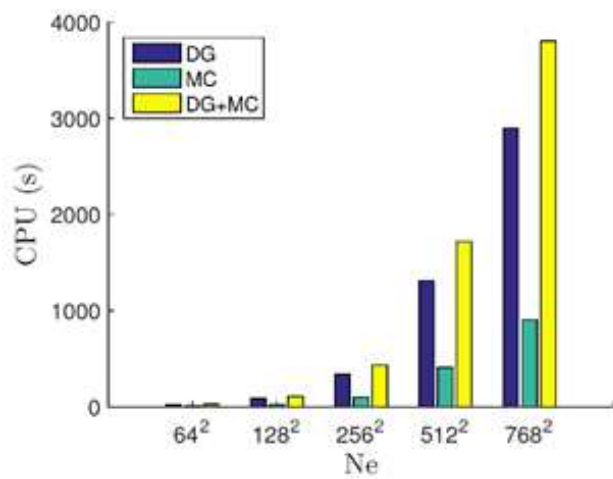
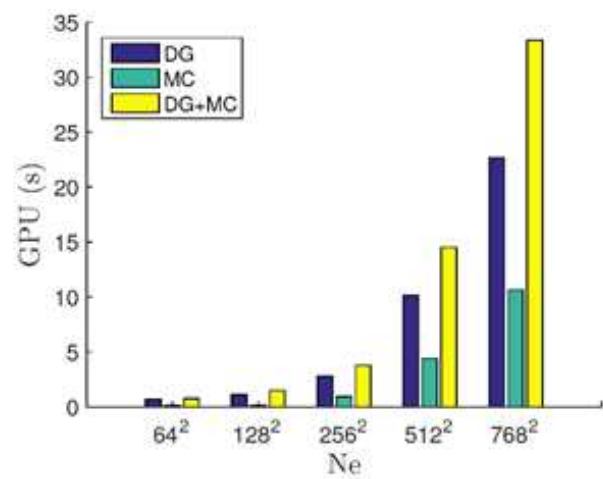


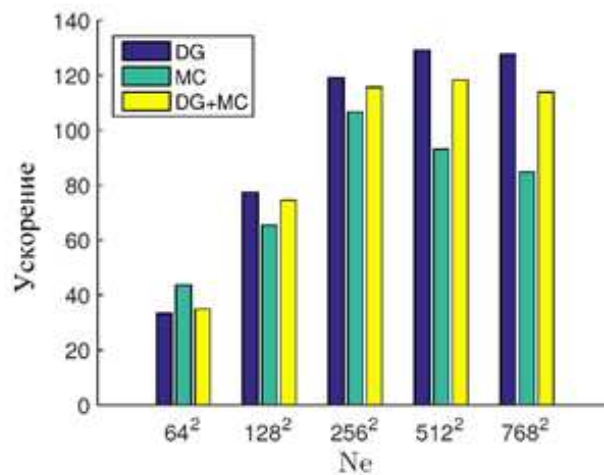
Рисунок 32 – Время вычислений для MC с $NPQ = 32$ и DG с $p = 1$. (а) Время CPU. (б) Время GPU. (в) Коэффициент ускорения



(a)

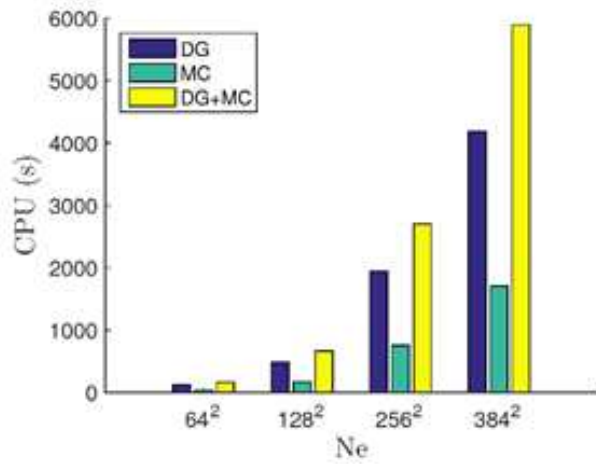


(б)

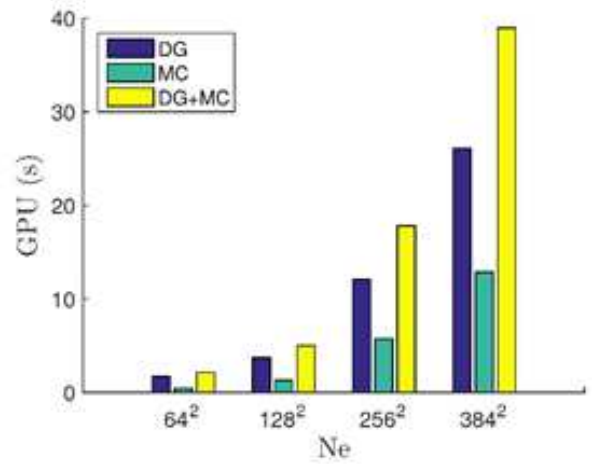


(в)

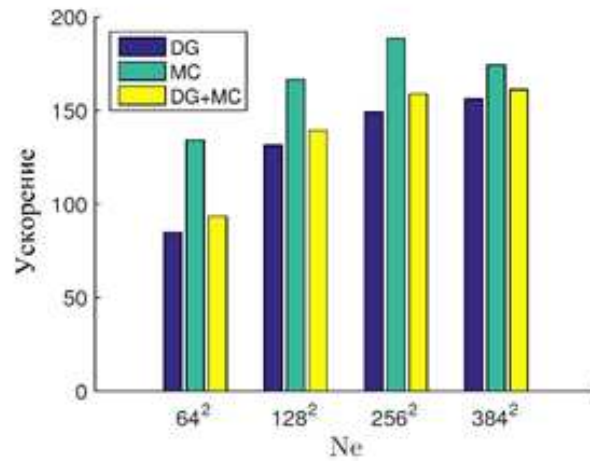
Рисунок 33 – Время вычислений для MC с $NPQ = 32$ и DG с $p = 2$. (a) Время CPU. (б) Время GPU. (в) Коэффициент ускорения



(a)



(б)



(в)

Рисунок 34 – Время вычислений для MC с $NPQ = 32$ и DG с $p = 4$. (а) Время CPU. (б) Время GPU. (в) Коэффициент ускорения

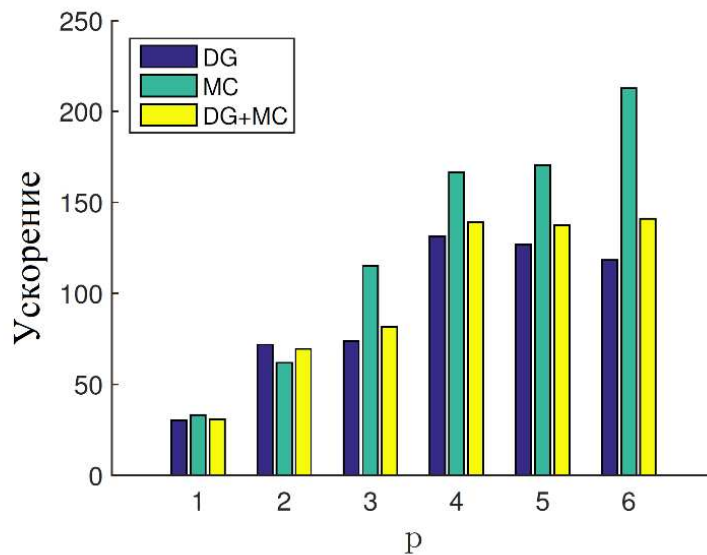


Рисунок 35 – Влияние степени полинома p на коэффициент ускорения при фиксированном $N_e = 128 \times 128$ и $NPQ = 32$

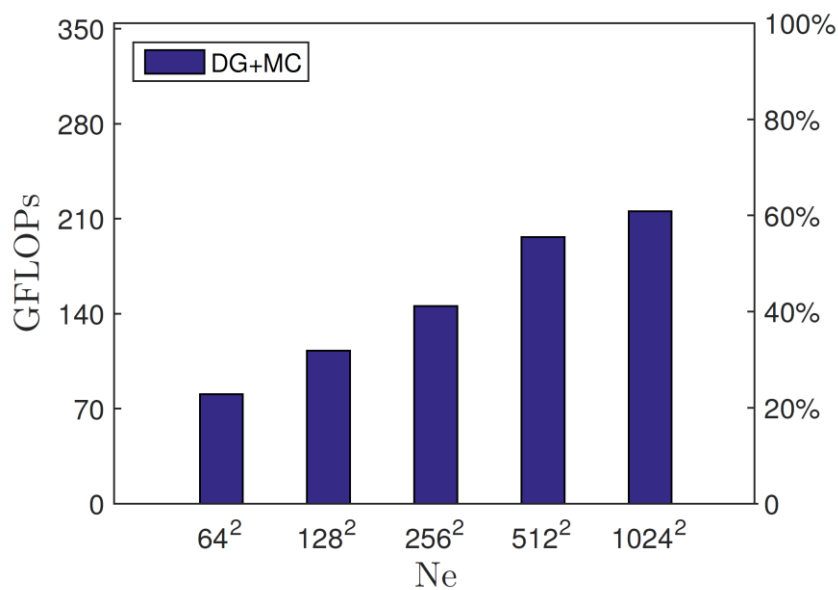


Рисунок 36 – Общая вычислительная производительность решателя для различного числа элементов с фиксированным $p = 3$ и $NPQ = 16$. Расчеты выполнены на GeForce GTX 1080 Ti. Максимальная теоретическая производительность составляет 354 GFLOPS

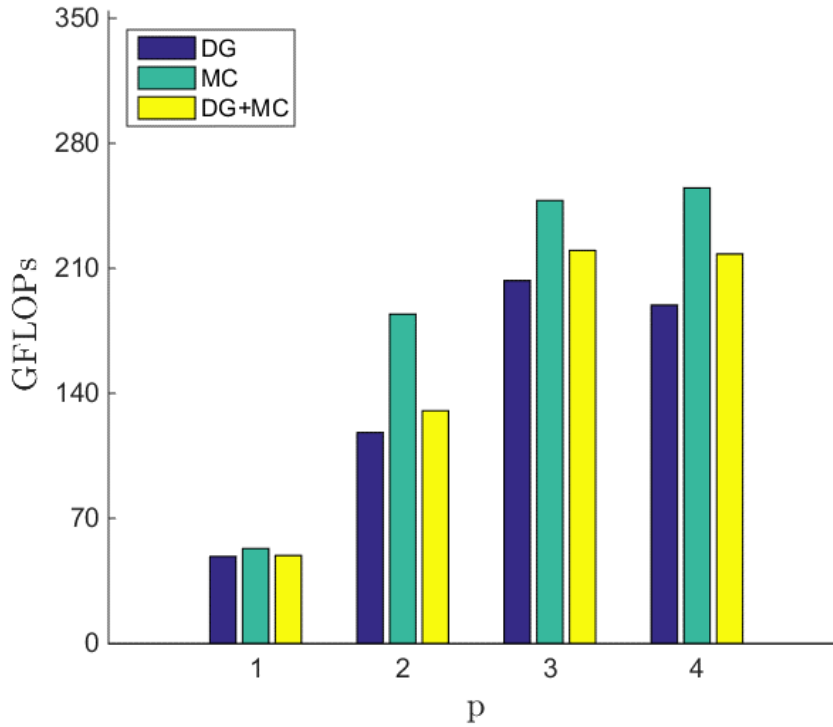


Рисунок 37 – Влияние степени полинома p на производительность при фиксированном $N_e = 32 \times 32 \times 32$ и $NPQ = 32$

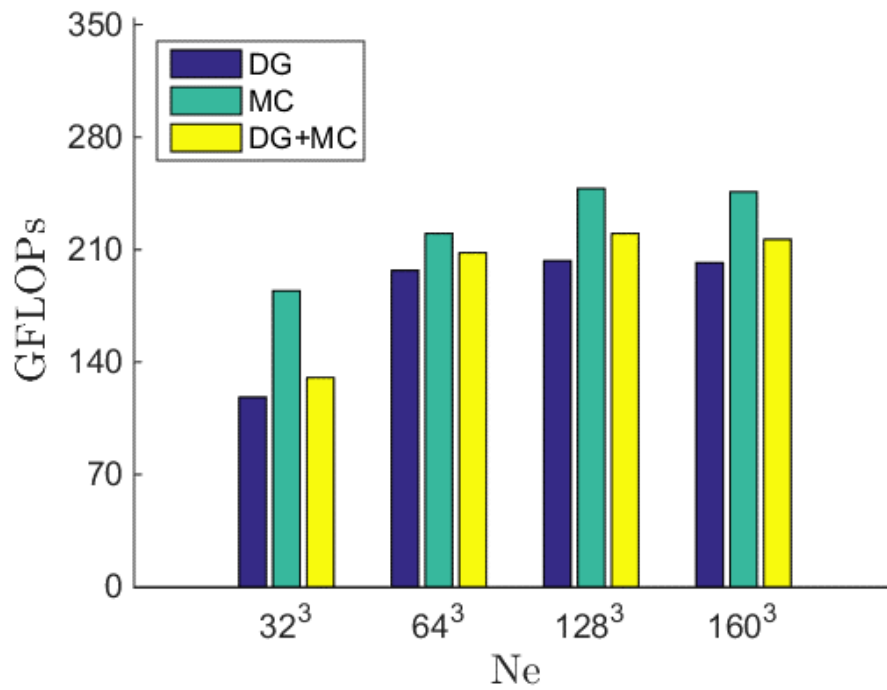


Рисунок 38 – Влияние разрешения сетки N_e на производительность при фиксированном $p = 2$ и $NPQ = 32$

ЗАКЛЮЧЕНИЕ

В диссертационной работе разработана новая **высокопроизводительная вычислительная методология** для моделирования реагирующих турбулентных течений несжимаемой жидкости методом крупных вихрей. Для численного решения основных фильтрованных уравнений течения разработана схема на основе разрывного метода Галеркина (DG), позволяющая с высокой точностью определить такие динамические величины течения, как скорость и давление. Для замыкания источника химической реакции применяется метод фильтрованной функции плотности (FDF). Смоделированное уравнение переноса FDF решается численно с помощью лагранжевого метода Монте-Карло (MC). Гибридная схема численного решения DG-MC построена таким образом, чтобы её можно было реализовать с использованием параллельных вычислений. Параллельная версия кода реализована с помощью технологии CUDA для проведения расчетов на вычислительных системах, основанных на графических устройствах, что позволяет проводить вычисления с высокой производительностью. Это существенно расширяет исследователям круг доступных для численного исследования задач реагирующих турбулентных течений.

Оценка общей производительности методологии оценивается путем численного моделирования и компьютерной реализации дву- и трехмерных задач слоев смешения как нейтральных, так и химически реагирующих компонент. Прогностическая способность продемонстрирована путем сравнения статистических данных усредненных по Рейнольдсу, с данными прямого численного моделирования (DNS).

Результаты выполненной работы показывает, что комбинирование метода FDF и гибридной численной схемы DG-MC является очень эффективным инструментом для моделирования реагирующих турбулентных течений методом крупных вихрей.

Основные особенности и преимущества численной методологии:

- i. Разрывный метод Галеркина сочетает в себе преимущества метода конечных объемов и спектральных методов: преимущество первого – в способности решать задачи в геометрически сложных областях, в то же время преимущество второго – в обеспечении высокой точности аппроксимации.
- ii. Высокую численную эффективность и устойчивость схемы разрывного метода Галеркина можно многократно усилить путем использования высоких порядков полинома аппроксимации p . Использование высоких порядков полиномов ($p \sim 4-6$) приводят к очень незначительным численным диссипациям, что очень важно в моделировании методом крупных вихрей.
- iii. Полная энергия течения определяется с высокой точностью даже при низких значениях порядка полинома аппроксимации p . Такое свойство схемы является особенно притягательной, когда прогноз развития полной энергии имеет первостепенное значение.

- iv. Существенным преимуществом разработанной методологии моделирования турбулентных течений методом LES является возможность достижения точности DNS в пределе путем повышения порядка полинома аппроксимации p . Показано что, с точки зрения вычислительной стоимости, такая процедура является намного экономичней по сравнению с процедурой сгущения сетки, которая является типичной в традиционных подходах аппроксимации.
- v. Показана эффективность использования DG методов в сочетании с лагранжевым методом MC на основе частиц. Значения частиц с легкостью определяются с помощью DG в любой точке элемента в силу того, что эти переменные представляются с помощью простых полиномов. Такое преимущество дает возможность избежать потери точности, как это происходит в обычных интерполяциях с использованием приближении низкого порядка.
- vi. Разработанная схема, за счет высокого порядка полиномов аппроксимации, позволяет решать задачи на более грубой сетке по сравнению с обычной дискретизацией, где во многих случаях для повышения точности аппроксимации используется метод сгущения сеток. Следовательно, это позволяет значительно уменьшить выделение оперативной памяти вычислительного устройства.
- vii. Существенное преимущество гибридной схемы DG-MC состоит в том, что она является явной и позволяет распараллеливать вычисления. Схема DG также имеет свойства локальности и является арифметически интенсивной. Отдельно стоит отметить, что значения частицы в элементе зависят только от значения переменных того же элемента. Все эти свойства схемы дают возможность решать задачи на массивно-параллельных вычислительных машинах.
- viii. Способность методологии производить параллельные вычисления на графических процессорах дает возможность исследователям решать ресурсоемкие задачи с высокой производительностью на настольных компьютерах, не прибегая к дорогостоящим вычислительным кластерам. Это позволит расширить как круг исследователей турбулентных реагирующих течений, так и круг решаемых задач.

Эффективность, высокая точность и надежность разработанного DG-MC FDF симулятора, как показали результаты проведенных численных исследований, гарантирует его дальнейшего расширения и применения к более сложным задачам моделирования турбулентных течений, включая турбулентное горение.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Reynolds O. An experimental investigation of the circumstances which determine whether the motion of water shall be direct or sinuous, and of the law of resistance in parallel channels // Philosophical Transactions of the Royal Society of London. – 1883. – Vol. 174. – P. 935 – 982.
- 2 Givi P. Spectral and random vortex methods in turbulent reacting flows // in: Turbulent Reacting Flows, Libby P. A. and Williams F. A., Ред., London, England: Academic Press. – 1994. – P. 475 – 572.
- 3 Menter F.R. Methoden, Möglichkeiten, Grenzen numerischer Strömungsberechnungen // in: Kurzlehrgang NUMET 2002: Numerische Methoden zur Berechnung von Strömungs- und Wärmeübertragungsproblemen. Erlangen, Deutschland – 2002. – P. 1.
- 4 Fox D.G., Lilly D.K. Numerical simulation of turbulent flows // Reviews of Geophysics. – 1972. – Vol. 10. – P. 51 – 72.
- 5 Jones W. Turbulence modelling and numerical solution methods for variable density and combusting flows // in: Turbulent Reacting Flows, Academic Press. – 1994. – P. 309 – 374.
- 6 Spiegel E. A., Veronis G. On the Boussinesq approximation for a compressible fluid // Astrophysical Journal. – 1960. – P. 442 – 447.
- 7 Parente A., Górlé C., van Beeck J., Benocci C. Improved $k-\varepsilon$ model and wall function formulation for the RANS simulation of ABL flows // Journal of Wind Engineering and Industrial Aerodynamics. – 2011. – Vol. 99(4). – P. 267 – 278.
- 8 Rogallo R. S., Moin P. Numerical simulation of turbulent flows // Annual Review of Fluid Mechanics. – 1984. – Vol. 16. – P. 99 – 137.
- 9 Givi P. Filtered density function for subgrid scale modeling of turbulent combustion // AIAA J. – 2006. – Vol. 44(1). – P. 16 – 23.
- 10 Инкарбеков М.К. Исследование фильтрованной функции плотности для моделирования крупных вихрей сложных турбулентных течений // Вестник КазНУ им. аль-Фараби: Вычислительные технологии. – 2001. – Т. 20, № 3. – С. 231 – 241.
- 11 Germano M., Piomelli U., Moin P. and Cabot W. H. A dynamic subgrid-scale eddy viscosity model // Physics of Fluids. – 1991. – Vol. 3(7). – P. 1760 – 1765.
- 12 Colucci P. J., Jaber F. A., Givi P. and Pope S. Filtered density function for large eddy simulation of turbulent reacting flows // Physics of Fluids. – 1998. – Vol. 10(2). – P. 499 – 515.
- 13 Nik M., Givi P., Madnia C. and Pope S. B. EPVS-FMDF for LES of high-speed turbulent flows // in: 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Nashville, Tennessee. – 2012. – P. 1 – 12.

- 14 Ansari N., Jaber F. A., Sheikhi M. R. H. and Givi P. Filtered density function as a modern cfd tool // Engineering Applications of Computational Fluid Dynamics. – 2011. – Vol. 1. – P. 1 – 22.
- 15 Banaeizadeh A., Li Z. and Jaber F. A. Compressible scalar filtered density function model for high-speed turbulent flows // AIAA J. – 2011. – Vol. 49, – P. 2130-2143.
- 16 Carrara M. D., Des Jardin P. E. A filtered mass density function approach for modeling separated two-phase flows for LES I: Mathematical formulation // Int. J. Multiphas. Flow. – 2006. – Vol. 32. – P. 365-384.
- 17 Cha C. M. and Trouillet P. A subgrid-scale mixing model for large-eddy simulation of turbulent reacting flows using the filtered density function // Phys. Fluids, – 2003. – Vol. 15. – P. 1496 –1504.
- 18 Drozda T.G., Sheikhi M.R.H., Madnia C. K. and Givi P. Developments in formulation and application of the filtered density function // Flow Turbul. Combust. – 2007. – Vol. 78, – P. 35 – 67.
- 19 Zhou X.Y. and Pereira J. C. F. Large eddy simulation (2d) of a reacting plane mixing layer using filtered density function closure // Flow Turbul. Combust. – 2000. – Vol. 64, – P. 279 – 300.
- 20 Zhao W., Zhang C. and Chen C. Large eddy simulation of bluff–body stabilized flames using a multi – environment filtered density function model // Proc. Combust. Inst. – 2011. – Vol. 33. – P. 1347 – 1353.
- 21 Wang D. and Tong C. Conditionally Filtered Scalar Dissipation, Scalar Diffusion, and Velocity in a Turbulent Jet // Phys. Fluids. – 2002. – Vol. 14. – P. 2170 – 2185.
- 22 L. Y. M. Gicquel , P. Givi, F. A. Jaber and S. B. Pope Velocity filtered density function for large eddy simulation of turbulent flows // Physics of Fluids. – 2002. – Vol. 14(3). – P. 1196 – 1213.
- 23 Barlow R. S. and Frank J. H. Effects of Turbulence on Species Mass Fractions in Methane/Air Jet Flames // in: Symposium (International) on Combustion. – 1998. – Vol. 27(1). – P. 1087 – 1095.
- 24 Nouri A. G., Nik M. B., Givi P., Livescu D. and Pope S. B. Self-contained filtered density function // Phys. Rev. Fluids. – 2017. – Vol. 2(9). – P. 1 – 20.
- 25 Темирханов А.Б., Аджан Б., Инкарбеков М.К. Моделирования крупных вихрей турбулентных течений с помощью узлового разрывного метода Галеркина// Colloquium-journal. – 2019. – С. 22 – 25.
- 26 Sammak S., Ren Zh., Givi P. Modern Developments in Filtered Density Function // in: Modeling and Simulation of Turbulent Mixing and Reaction. – 2020. – P. 181 – 200.

- 27 Grigoriu M. Applied non-gaussian processes // Englewood Cliffs, NJ: PTR Prentice Hall, 1995, – 442 p.
- 28 Kloeden P. E., Platen E. and Schurz H., Numerical solution of sde through computer experiments // Springer-Verlag Berlin Heidelberg, 1994, – 312 p.
- 29 Sammak S., Brazell M. J., Givi P. and Mavriplis D. J. A hybrid DG-Monte Carlo FDF simulator // Comput. Fluids. – 2016. – Vol. 140. – P. 158 – 166.
- 30 Yilmaz S. L., Ansari N., Pisciueneri P. H., Nik M. B., Otis C. C. and Givi P. Advances in FDF modeling and simulation // in 47th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, San. – 2011. – P. 1 – 11.
- 31 Pisciueneri P. H., Yilmaz S. L., Strakey P. A. and Givi P. An irregularly portioned FDF simulator // SIAM J. Sci. Comput. – 2013. – Vol. 35. – P. 438 – 452.
- 32 Owens J., Houston M., Luebke D., Green S., Stone J. E. and Phillips J. C. GPU computing // Proceedings of the IEEE. – 2008. – Vol. 96. – P. 879 – 899.
- 33 Pisciueneri P. H., Meneses E. and Givi P. Dynamic Load Balancing Strategies for Parallel Reacting Flow Simulations // in: Bulletin of the American Physical Society, San. – 2014. – P. R31. 001
- 34 Dumbser M., Zanotti O., Loubère R. and Diot S. A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws // Journal of Computational Physics. – 2014. – Vol. 278, – P. 47 – 75.
- 35 Collis S. S. Discontinuous Galerkin Methods for Turbulence Simulation // in: Proceedings of the 2002 Center for Turbulence Research Summer Program. – 2002. – P. 155 – 167.
- 36 Landmann B., Kessler M., Wagner S. and Krämer E. A parallel, high-order discontinuous Galerkin code for laminar and turbulent flows // Computers & Fluids. – 2008. – Vol. 37, – P. 427 – 438.
- 37 Klöckner A., Warburton T. and Hesthaven J. S. High-order discontinuous Galerkin methods by GPU metaprogramming // in GPU Solutions to Multi-scale Problems in Science and Engineering, Berlin, Germany: Springer. – 2013. – P. 353 – 374.
- 38 Klöckner A., Warburton T., Bridge J. and Hesthaven J. S. Nodal discontinuous Galerkin methods on graphics processors // Journal of Computational Physics. – 2009. – Vol. 228, – P. 7863 – 7882.
- 39 Gandham R., Medina D. and Warburton T. GPU accelerated discontinuous Galerkin methods for shallow water equations // Communications in Computational Physics. – 2015. – Vol. 18, – P. 37 – 64.
- 40 Fuhry M., Giuliani A. and Krivodonova L. Discontinuous Galerkin methods on graphics processing units for nonlinear hyperbolic conservation laws // International Journal for Numerical Methods in Fluids. – 2014. – Vol. 76, – P. 982 – 1003.

41 Высокопроизводительный 3D симулятор фильтрованной функции плотности для моделирования крупных вихрей турбулентных реагирующих течений, основанный на методах разрывного Галеркина и Монте Карло: отчет о НИР (промежуточный) / АО «Нац. центр научно-техн. информ.»: рук. Калтаев А.; исполн.: Инкарбеков М. К. – Алматы, 2018. – 55 с. – № ГР 0118РК00564.

42 Высокопроизводительный 3D симулятор фильтрованной функции плотности для моделирования крупных вихрей турбулентных реагирующих течений, основанный на методах разрывного Галеркина и Монте Карло: отчет о НИР (промежуточный) / АО «Нац. центр научно-техн. информ.»: рук. Калтаев А.; исполн.: Инкарбеков М. К. – Алматы, 2019. – 77 с. – № ГР 0118РК00564.

43 Inkarbekov M., Aitzhan A., Kaltayev A. and Sammak S. A GPU-accelerated filtered density function simulator of turbulent reacting flows // International Journal of Computational Fluid Dynamics. – 2020. – Vol. 34(6) – P. 381 – 396.

44 Инкарбеков М.К., Айтжан А., Калтаев А. Использование комбинированного DG-МС для моделирования крупных вихрей реагирующих турбулентных течений // Совместный выпуск Вестник КазНУ им. аль-Фараби и Вычислительные технологии Института Вычислительных технологий СО РАН. – 2016. – № 5. – P. 459 – 466.

45 Temirkhanov A., Inkarbekov M., Assilbekov B. Nodal discontinuous Galerkin method for large eddy simulation of turbulent flows // Вестник КазНТУ имени К. И. Сатпаева. – 2019. – Vol. 20(6), – P. 703 –710.

46 Аджан Б.З., Инкарбеков М.К. Модальный метод разрывного Галеркина для моделирования крупных вихрей турбулентных течений // Вестник Satbayev University. – 2018. – № 6, – P. 736-741.

47 Inkarbekov M., Aitzhan A., Sammak Sh., Koldas A., Kaltayev A., Givi P. A gpu accelerated DG-FDF simulator for large eddy simulation of reacting turbulent flows // in Bulletin of APS March Meeting 2017, New Orleans. – 2017. – P. 1373.

48 Inkarbekov M., Aitzhan A., Sammak S., Givi P., Kaltayev A. GPU accelerated DG-FDF large eddy simulator // in: Bulletin of 70th Annual Meeting of the APS Division of Fluid Dynamics, Denver, Colorado. – 2017. – P. 436.

49 Inkarbekov M., Aitzhan A., Sammak S., Kaltayev A., Givi P. A gpu accelerated FDF simulator // in: Abstract of Seventeenth International Conference on Numerical Combustion, Aachen, Germany. – 2019. – P. 15.

50 Inkarbekov M., Kaltayev A. A gpu accelerated discontinuous Galerkin-Monte Carlo large eddy simulator // in: Program of Inverse Problems in Finance, Economics and Life Sciences, Almaty. – 2019.

- 51 Aitzhan A., Inkarbekov M.K., Koldas A., Kaltayev A. Implementation of scalar filtered density function for large eddy simulation of turbulent reacting flows using a high – order discontinuous Galerkin method // in: The Abstract Book ICAAM. – 2016.
- 52 Sagaut P. Large eddy simulation for incompressible flows, New, York: Springer, 2010. – 558 p.
- 53 Pope S. B. Computations of turbulent combustion: Progress and challenges // Proc. Combust. Inst. – 1990. – Vol. 23, – P. 591 – 612.
- 54 Sagaut P., Large Eddy Simulation for Incompressible Flows, Springer – Verlag Berlin Heidelberg. – 2001. – XV, 319 p.
- 55 Smagorinsky J. General circulation experiments with the primitive equations. I. The Basic Experiment // Monthly Weather Review. – 1963. – Vol. 91, – P. 99 – 164.
- 56 Im H. G., Lund T. S. and Ferziger J. H. Large eddy simulation of turbulent front propagation with dynamic subgrid models // Phys. Fluids. – 1997. – Vol. 9, – P. 3826 – 3833.
- 57 Schumann U. Large eddy simulation of turbulent diffusion with chemical reactions in the convective boundary layer // Atmos. Environ. – 1989. – Vol. 23, – P. 1713 – 1726.
- 58 Libby P. A. and Williams F. A. Turbulent Reacting Flows // Heidelberg: Springer – Verlag. – 1980. – 246 p.
- 59 Colucci P. J., Jaber F. A., Givi P. and Pope S. B. Filtered density function for large eddy simulation of turbulent reacting flows // Phys. Fluids. – 1998. – Vol. 10, – P. 499 – 515.
- 60 Jaber F. A., Colucci P. J., James S., Givi P. and Pope S. B. Filtered mass density function for large eddy simulation of turbulent reacting flows // J. Fluid Mech. – 1999. – Vol. 401. – P. 85 – 121.
- 61 Karlin S. and Taylor H. M. A second course in stochastic processes, New, York: Academic Press. – 1981. – 542 p.
- 62 Risken H. The Fokker-Planck equation, methods of solution and applications, New-York: Springer – Verlag. – 1989. – 490 p.
- 63 Thomas J. Numerical partial differential equations: finite difference methods, – Vol. 22, New York: Springer – Verlag. – 1995. – 437 p.
- 64 Kolditz O. Finite Volume Method // in: Computational Methods in Environmental Fluid Mechanics, Berlin, Springer, Berlin, Heidelberg, 2002, – P. 173 – 190.
- 65 Reed W.H. and Hill T.R. Triangular mesh methods for the neutron transport equation // in: Materials of National topical meeting on mathematical models and computational techniques for analysis of nuclear systems, United States, 1973.
- 66 Chorin A. J. A numerical method for solving incompressible viscous flow problems // Journal of Computational Physics. – 1967. – Vol. 2(1). – P. 12 – 26.

- 67 Pope S. B. Lagrangian PDF methods for turbulent flows // Annu. Rev. Fluid Mech. – 1994. – Vol. 26, – P. 23 – 63.
- 68 Afzal A., Ansari Z., Faizabadi A.R., Ramis M. K. Parallelization strategies for computational fluid dynamics software: State of the art review // Archives of Computational Methods in Engineering. – 2017. – Vol. 24(2) – P. 337 – 363.
- 69 GeForce GTX 1080 Ti Graphics Card. <https://www.nvidia.com/en-sg/geforce/products/10series/geforce-gtx-1080-ti>. 18.09.2019.
- 70 NVIDIA CUDA C Programming Guide v4.2. https://developer.download.nvidia.com/compute/DevZone/docs/html/C/doc/CUDA_C_Programming_Guide.pdf. 20.04.2017.
- 71 Ruetsch G. and Fatica M., CUDA Fortran for Scientists and Engineers: Best Practices for Efficient CUDA Fortran Programming // San, Francisco: Morgan Kaufmann Publishers Inc. – 2013. – 338 p.
- 72 Vreman B., Geurts B. and Kuerten H. Realizability Conditions for the Turbulent Stress Tensor in Large – Eddy Simulation // J. Fluid Mech. – 1994. – Vol. 278. – P. 351 – 362.
- 73 Bilger R. W. Future Progress in Turbulent Combustion Research // Progress in Energy and Combustion Science. – 2000. – Vol. 26. – P. 367 – 380.
- 74 Bardina J., Ferziger J. H. and Reynolds W. C. Improved Turbulence Models Based on Large Eddy Simulation of Homogeneous, Incompressible Turbulent Flows // National Aeronautics and Space Administration NASA. – 1983. – 194 p.