Адаптивное измельчение сетки на основе вставок в гидродинамике

Совместная российско-американская конференция пяти лабораторий по вычислительной математике и физике

Вена, Австрия

19-23 июня, 2005 г.

Ричард Пембер (Richard Pember) и Джеффри Грино (Jeffrey Greenough)

Отделение AX Division

Бен Лю (Ben Liu) и Илья Ломов (Ilya Lomov)

Отделение наук о Земле

16-20 июня 2005 г.

Данная работа выполнена при поддержке Министерства энергетики США Ливерморской национальной лабораторией им. Лоуренса Университета штата Калифорния по контракту № W-7405-Eng-48.

Адрес: Lawrence Livermore National Laboratory, P.O. Box 808, Livermore, CA 94551-0808

UCRL-PRES-212694

Содержание

- Начальный курс по адаптивному измельчению сетки (AMR) для газовой динамики однокомпонентной сплошной среды
- Вопросы, относящиеся к многокомпонентным системам, прочности материалов и прочим физическим явлениям.
- Различные результаты

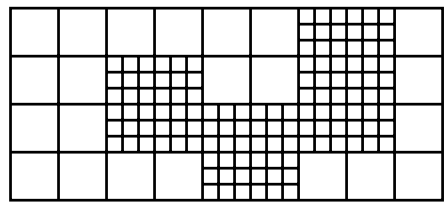
Зачем нужно адаптивное измельчение сетки?

- Цель: сосредоточить вычислительные ресурсы в тех участках области вычислений, где это требуется для достижения точности.

- Зачем это делать?
  - В надежде сэкономить ресурсы ЦПУ
  - В надежде сэкономить ресурсы памяти

В этом докладе мы сосредоточимся на адаптивном измельчении сетки на основе подхода Бергера, Олигера и Колеллы (Berger-Oliger-Colella)

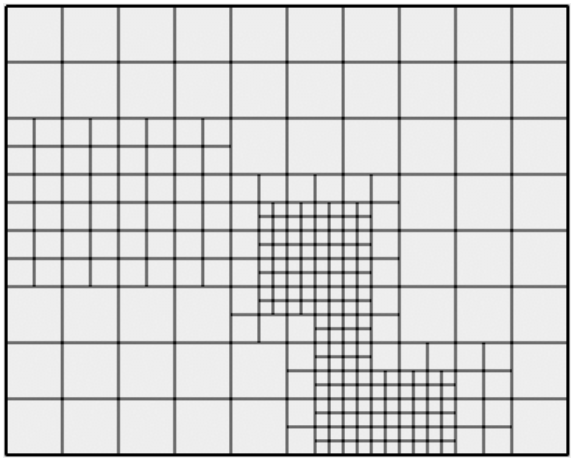
- Особенности подхода Бергера, Олигера и Колеллы:
  - Местное пространственное и временное измельчение вставок в область вычислений.
  - Каждая вставка представляет собой логически прямоугольную структурированную решетку.
  - Вставки “правильно вложены”.
  - Вставка состоит из зон с большой “погрешностью”, объединенных с некоторыми (но немногими) зонами с малой погрешностью.
  - Сетки динамически создаются и разрушаются для того, что обеспечить учет изменяющихся свойств нестационарного потока.
  - Пространственные и временные размеры вставок различны.
  - Субитерации по времени (рекурсивные шаги по времени) возможны, но не необходимы.

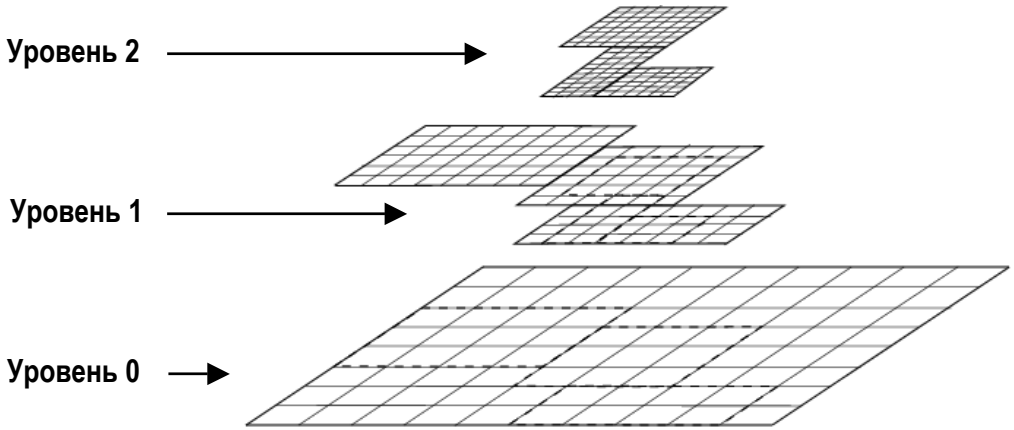


- В данном докладе обсуждаются
  - Физически прямоугольные сетки.
  - Переменные, связанные с центрами ячеек.
  - Переход на следующий уровень по времени проводится в один шаг с помощью явной, прямой эйлеровой, дискретно консервативной структурированной сетки 2-го порядка точности:
    - шаги по времени для структурированной сетки производятся с помощью “фантомных зон”;
    - точность 2-го порядка → линейные интерполяции достаточны.
  - Субитерации по времени.

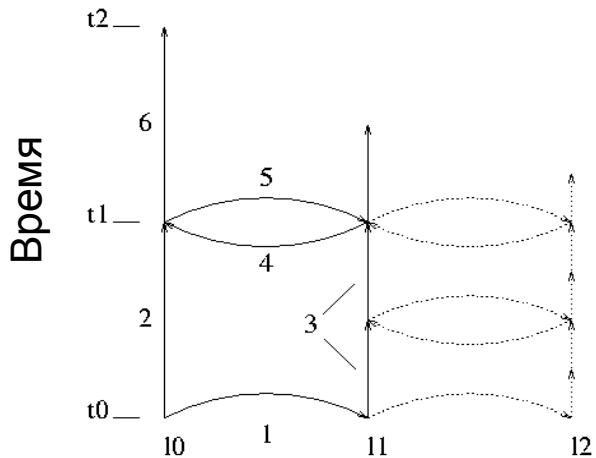
Адаптивное измельчение сетки Бергера, Олигера и Колеллы использует динамическую иерархию сеток и рекурсивный шаг по времени

- **Пространственное измельчение**





- **(Рекурсивный) шаг по времени для крупного уровня:**
  - Следующий крупный уровень
  - Следующий мелкий уровень
  - Синхронизация уровней
  - Перестроение сетки на текущем и всех более мелких уровнях

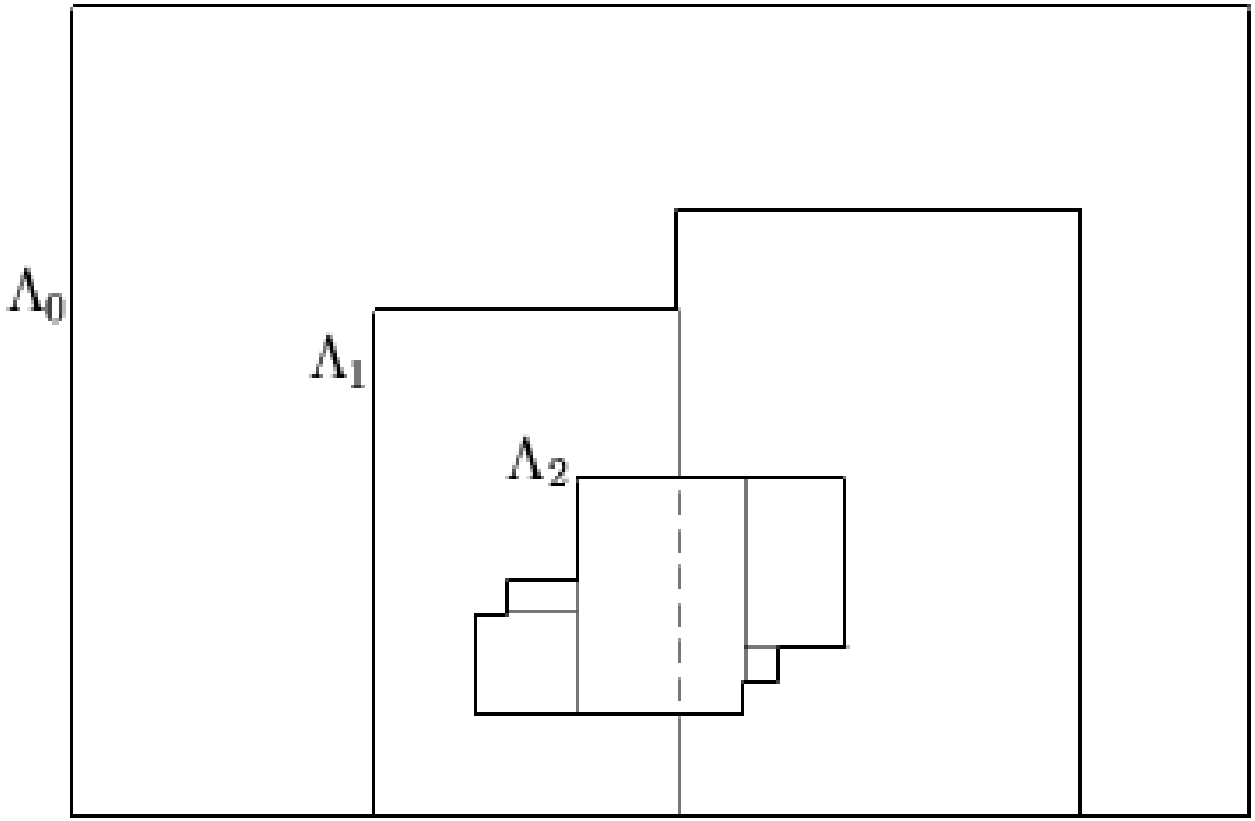


Уровни пространственного измельчения

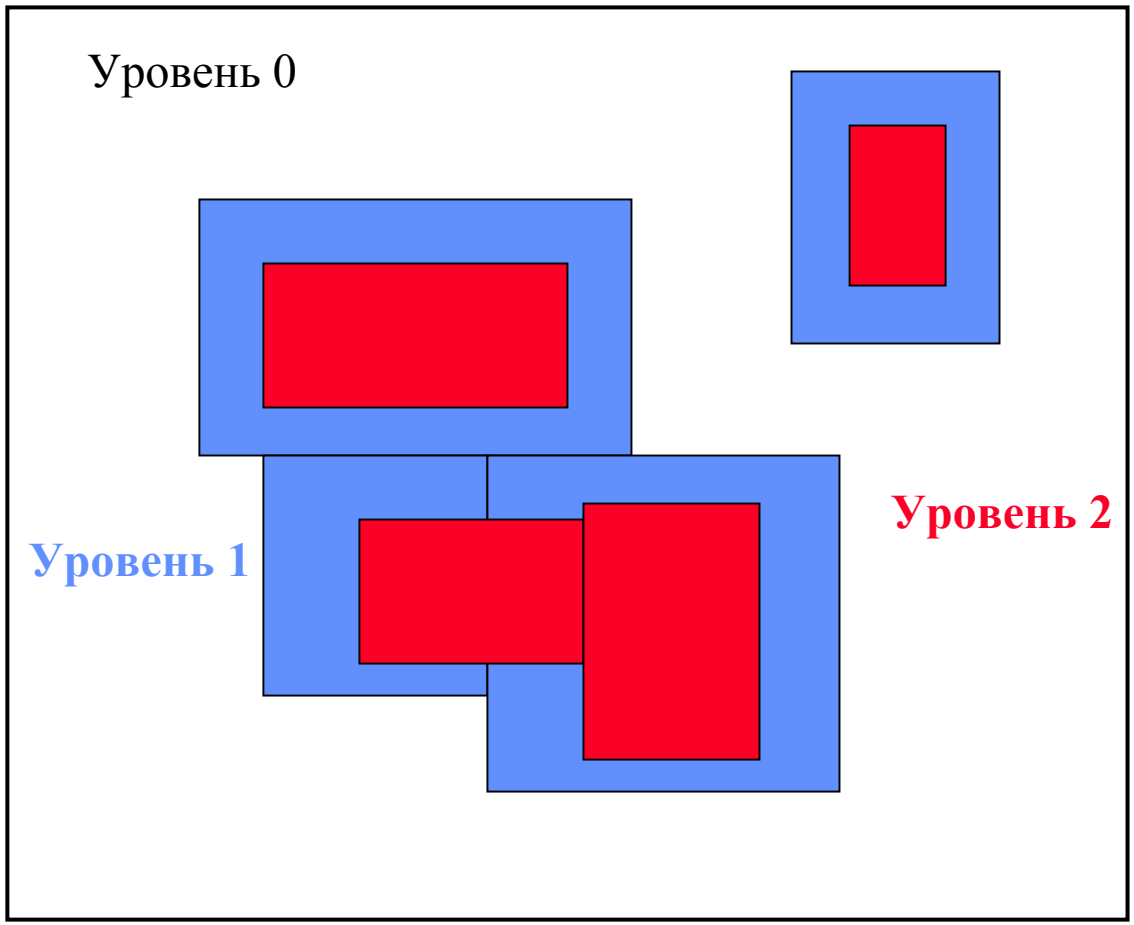
Рекурсивный шаг по времени при адаптивном измельчении сетки

- **Коэффициент измельчения: 2, 2**
- **Измельчение в пространстве *и* времени**

Обеспечивается “правильное вложение” вставок



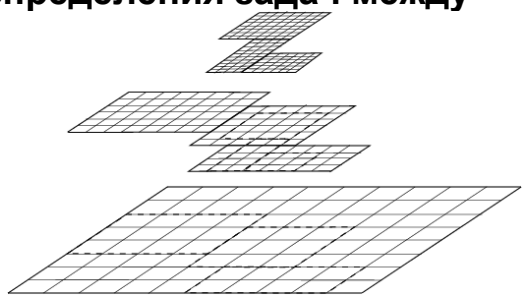
Обеспечивается “правильное вложение” вставок



- Основная вставка соответствует Уровеню 0 и покрывает всю область вычислений.
- Производится местное пространственное измельчение для создания нового уровня. Структуры, погрешности и т.д. помечаются для измельчения.
- Более мелкие уровни строго содержатся внутри следующего более крупного уровня -> правильное вложение
- Решение на некотором уровне (объединении сеток) сохраняется в качестве фундаментального объекта.

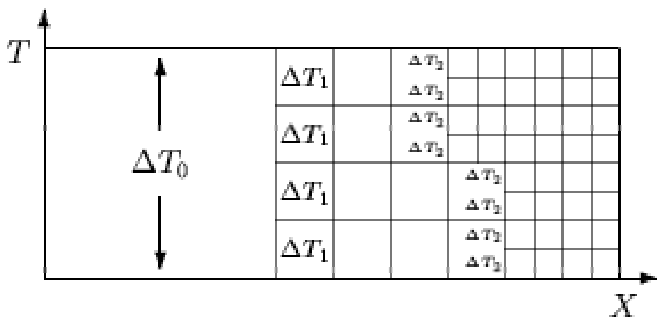
Зачем нужно адаптивное измельчение сетки переменных размеров на основе вставок?

- Некоторые альтернативы
  - Измельчение для отдельных ячеек (напр., Рэйдж (Rage))
    - Преимущества:
      - минимальная память
      - простые структуры данных (4-дерево, 8-дерево)
      - упрощается формирование сеток (не требуется проводить кластеризацию)
    - Возможные недостатки (по моему предположению)
      - обмен данными требует более значительных затрат
      - большее количество “нерегулярных” зон
  - Фиксированный размер вставки (напр., Грэйс (Grace))
    - Преимущества: простота, экономичность обмена данными
    - Возможные недостатки (по моему мнению)
      - Неэффективное использование памяти: процент измельченных зон с малой погрешностью выше, чем при измельчении на основе вставок или для отдельных ячеек
- Действительно, зачем?
  - Первоначальная причина: векторизация на компьютерах марки Cray
  - Обоснование отказа от изменений:
    - Вставки являются удобной единицей для распределения задач между несколькими процессорами
    - Замечание: это - *не* разбиение области вычислений
      - Родительская и дочерняя сетки не обязательно должны быть расположены на одном и том же процессоре.



Общая форма рекурсивного шага по времени для неявной консервативной схемы

- Нахождение шага по времени для уровня l
  - При субитерациях: использовать $min(\Delta t_l, \Delta t_{l-1}/r_l)$
  - Без субитераций: использовать $min_k (\Delta t_k)$
- Переход на уровень l за один шаг по времени
  - Граничные условия с шага l-1 по необходимости (интерполированные во времени и пространстве при необходимости)
  - Игнорируются уровень $l+1$ и более высокие уровни
- Переход на уровне $l+1$ к тому же значению времени
  - Несколько шагов по времени при субитерациях
  - Граничные условия с уровня l по необходимости
- Синхронизация уровней l и $l+1$
  - При перекрывании переопределить решение уровня l осреднением решения уровня $l+1$
  - Если уровни ячеек l находятся за пределами уровня $l+1$, изменить решение уровня l, чтобы исправить несоответствие “потоков”
- Если настало время перестроить сетку, переопределить уровни l и выше
  - Провести рекурсивную интерполяцию с уровня l-1 по необходимости



Пример явной консервативной схемы

- **Записать уравнение в консервативной форме, например:**

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

где

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix} \quad F(U) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u E + p u \end{pmatrix} \quad G(U) = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho v E + p v \end{pmatrix}$$

и $E = e + 1/2\left(u^2 + v^2\right)$

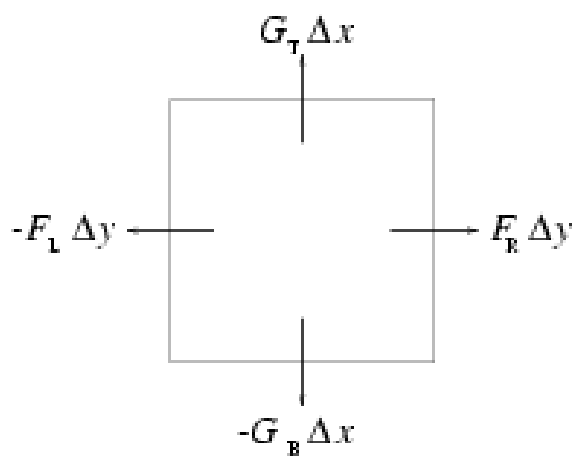
Пример явной консервативной схемы (продолжение)

- **Конечно-разностная схема ≡ аппроксимация** $\nabla \cdot \vec{F} \quad (\vec{F} = (F, G))$
- **Напоминание:**

$$\int \nabla \cdot \vec{F} dV = \int \vec{F} \cdot \vec{n} dA$$

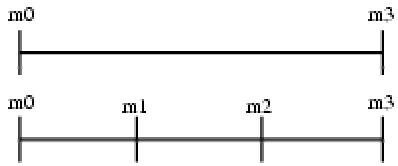
- **Используется теорема дивергенции:**

$$\nabla \cdot \vec{F} \approx \frac{1}{\Delta x \Delta y} \sum_i \vec{F}_i \cdot \vec{n}_i A_i, \quad i = \mathrm{L, R, T, B}$$



- **Пересчет:** $U^{n+1} = U^n - \Delta t \nabla \cdot \vec{F}$

Консервативная интерполяция использует ограниченные наклоны и объемные и массовые координаты



- массовые координаты:
  - масса в крупных ячейках: $m_3 - m_0$
  - масса в i-й мелкой ячейке: $m_i - m_{i-1}$
  - центр масс крупной ячейки:

$$M^c = \frac{m_0 + m_3}{2}$$

  - центры масс мелких ячеек:

$$M_i^f = \frac{m_{i-1} + m_i}{2}$$

  - интерполяция:

$$q_i^f = q^c + \delta q \left(M_i^f - M_c \right)$$

  - консервативная форма:

$$\sum \left(m_{k+1} - m_k \right) \left(\frac{m_{k+1} + m_k}{2} - \frac{m_3 + m_0}{2} \right) = 0$$

- объемные координаты: аналогично
- более высокие измерения: применяются итеративно к слоям, полосам и ячейкам
- использование
  - объемных координат для ρ
  - массовых координат для *u, v, E*

- ***Замечание: эта схема – не является консервативной интерполяцией за счет адвекции.***
- ***Это лишь старая добрая линейная (консервативная) интерполяция.***

Интерполяция объемных координат для пространственной прямоугольной сетки – это лишь простая линейная интерполяция

(1) Определение ограниченных центрально-разностных аппроксимаций

$$\left(\Delta_x U^{L-1}\right)_{i,j} / \Delta x_{L-1}, \ \left(\Delta_y U^{L-1}\right)_{i,j} / \Delta y_{L-1},$$

для пространственных производных

$$\partial U^{L-1} / \partial x, \ \partial U^{L-1} / \partial y$$

данных на крупной сетке в точках x_i, y_i, z_k

(2) Данные на мелкой сетке определяются следующим образом:

$$\begin{aligned} U^L_{l,m} &= U^{L-1}_{i,j} + (x_l - x_i)\left(\Delta_x U^{L-1}\right)_{i,j} / \Delta x_{L-1} \\ &+ (y_m - y_j)\left(\Delta_y U^{L-1}\right)_{i,j} / \Delta y_{L-1} \end{aligned}$$

Переход на более крупную сетку – это просто объемно или массово взвешенное осреднение

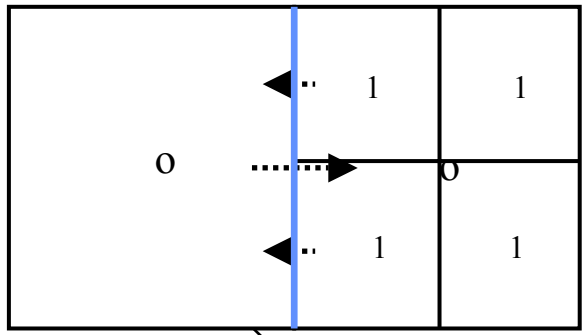
$$\rho_{i,j}^{L} = \sum_{l=r_L i}^{r_L(i+1)-1} \sum_{m=r_L j}^{r_L(j+1)-1} \frac{\Delta x_{L+1} \Delta y_{L+1} \rho_{l,m}^{L+1}}{\Delta x_L \Delta y_L}.$$

$$\rho_{i,j}^{L} q_{i,j}^{L} = \sum_{l=r_L i}^{r_L(i+1)-1} \sum_{m=r_L j}^{r_L(j+1)-1} \frac{\Delta x_{L+1} \Delta y_{L+1} \rho_{l,m}^{L+1} q_{l,m}^{L+1}}{\Delta x_L \Delta y_L}.$$

для $q = u,\ v,\ E$

Поправка для потока меняет значения для крупных ячеек, имеющих общую грань с мелкой ячейкой, но не покрытых мелкими ячейками

- Предположим, что ячейка крупной сетки *i, j, k* на уровне *L* имеет общую правую границу в направлении *x* с границей сетки уровня *L+1*



Эта зона изменяется

- Определяем поправку для потока δF^L_{ijk}:

$$\delta F^L_{i,j,k} = \Delta t \Delta y_L \Delta z_L F^L_{i+1/2,j,k} - \sum_{J=1,r_L} \sum_{m=r_L j}^{r_L(j+1)-1} \sum_{n=r_L k}^{r_L(k+1)-1} \left(\frac{\Delta t}{r_L} \Delta y_{L+1} \Delta z_{L+1} F^{L+1,J}_{l-1/2,m,n} \right).$$

- Пересчитанное решение для ячейки *i, j, k* имеет следующий вид:

$$U^L_{i,j,k} = U^L_{i,j,k} + \frac{\delta F^L_{i,j,k}}{\Delta x_L \Delta y_L \Delta z_L}.$$

- Этот пересчет эквивалентен повторению интегрирования на крупной сетке с использованием суммы потоков в мелкой сетке для пересчета ячейки вместо потока в крупной сетке.
- Поправка в остальных трех случаях определяется аналогично.

Изменение сетки на уровне *L* определяет новые сетки на уровне *L* и выше

- Для уровня ilev = finest_level-1, L, ilev--
  - Пометить ячейки с высокой погрешностью на уровне ilev, предназначенные для измельчения
    - Высокая погрешность определяется
      - — экстраполяцией Ричардсона
        - Оценка погрешности как разности между укрупнением и переходом на следующий шаг по времени, а также между переходом на следующий шаг по времени и укрупнением
      - — определением характеристик (градиентов, поверхностей раздела, температур…)
      - — прочими факторами
    - Пометить зоны в буфере вокруг зон с высокой погрешностью таким образом, чтобы зоны с высокой погрешностью оставались (как следует надеяться) измельченными между шагами по переформированию сетки
      - — Ширина буфера = 1 зона * (интервал переформирования сетки)
    - Если ilev<finest_level-1, спроектировать отмеченные ячейки с уровня ilev+1 на уровень ilev
      - — Обеспечить правильное вложение
  - *Сгруппировать ячейки в кластеры*
  - Обвести вокруг каждого кластера минимально возможные прямоугольники
    - Разбить прямоугольники на меньшие, размер которых позволяет проводить операции с ними (< max_nx X max_ny)
    - Сформировать новые мелкие сетки (тривиально)
  - Сгенерировать данные для мелких сеток
    - Скопировать в точки пересечения значения со старых мелких сеток
    - В противном случае консервативно интерполировать с базовой крупной сетки

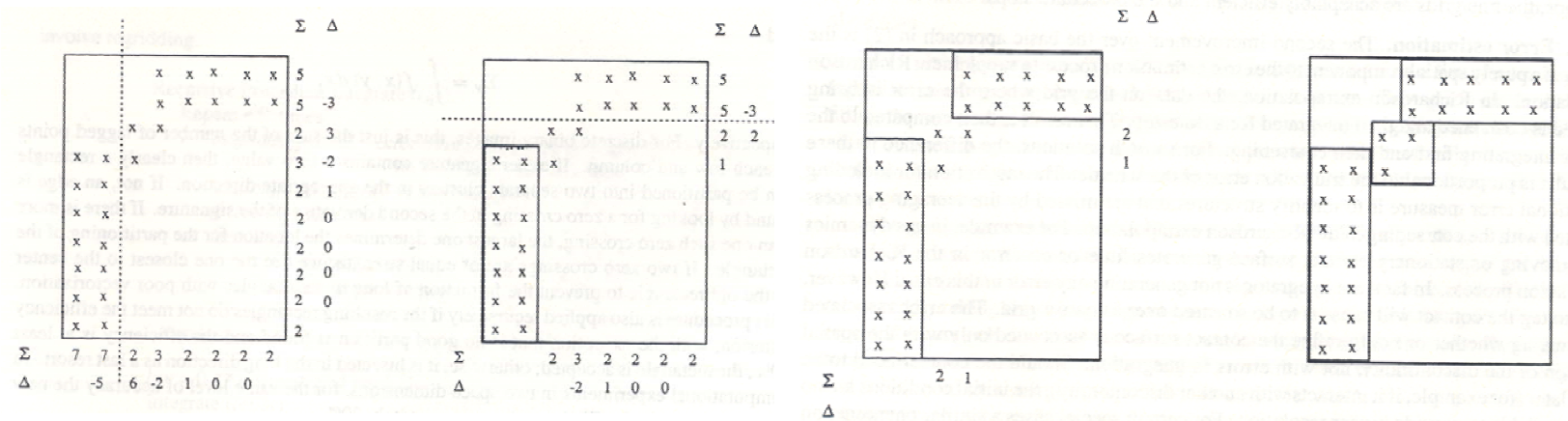
Алгоритм кластеризации является, в основном, алгоритмом интеллектуального разрезания

- Горизонтальные и вертикальные сигнатуры, Σ_X и Σ_Y, функции *f(x, y)* определяются следующим образом:

$$\Sigma_x = \int_y f(x,y)dy \quad \Sigma_y = \int_x f(x,y)dx$$

- Если *f(x, y)* = 1 в отмеченных зонах и 0 в остальных, то Σ_X и Σ_Y – это количество отмеченных зон в каждой строке и каждом столбце
- Края определяются нулевыми значениями сигнатур и вторых производных
- Применяется рекурсивно до тех пор, пока не будет выполнен критерий эффективности (например, 0,7 общего числа зон в прямоугольнике не будет отмечено)
- Если правильное деление не получается
  - если эффективность > 0,5, принять
  - в противном случае провести другой разрез

См. пример в статье Белла, Бергера, Зальцмана и Уэлкома, “Журнал научного программирования Общества промышленной и прикладной математики” (Bell, Berger, Saltzman, Welcome, SIAM J. Sci. Comput.), том 15, стр. 127-138, 1994



Вопросы, относящиеся к задачам с несколькими материалами и прочности материалов

- Контроль за поверхностями раздела
  - Всегда измельчать поверхности раздела: выполнено
  - В противном случае возникают следующие вопросы
    - Формулировка шага поправки потоков
    - Вопрос с операцией, обратной измельчению, и дальнейшим измельчением: потеря информации
  - Современный подход:
    - Всегда измельчать поверхность раздела или
    - Измельчать поверхность раздела, когда она более не находится в покое
      - Не проводить операции, обратной измельчению
      - Замечание: поправка к потоку не требуется, поскольку поток через границы крупных и мелких ячеек, являющиеся поверхностями раздела различных материалов, отсутствует.
- Прочность материалов
  - Изменяющиеся по времени переменные
  - Тензоры деформаций
    - Современный подход для обоих: ничего специального не делать
  - Поправка к потоку: тензор сдвиговых деформаций рассматривается в недивергентной постановке
    - Альтернативный вариант (см. Миллер и Колелла (Miller & Colella)): полностью консервативная постановка

Вопросы, связанные с другими физическими явлениями

- Физические явления нулевой размерности: вопросов нет
- Явные физические явления, подобные гидродинамическим: вопросов нет
- Нелокальные физические явления (диффузия, перенос)
  - Шаг синхронизации включает нелокальные (параболические, эллиптические) решения
    - Затраты на вычисления и разработку
  - Альтернативное решение: переделать шаг по времени
    - Последовательность для шага по времени:
      - «Гидродинамические» явления на всех уровнях
      - Диффузия на всех уровнях (посредством многоуровневого сопряженного решения)
      - …
    - Преимущества:
      - избегается шаг синхронизации
      - обеспечивается возможность использования имеющегося пакета для неструктурированной сетки
    - Недостатки:
      - нет субитераций: всюду используется единственный шаг по времени (на мелком уровне)
      - необходимо проводить пересчет двух структур данных друг в друга

Вопросы по разработке для других физических явлений

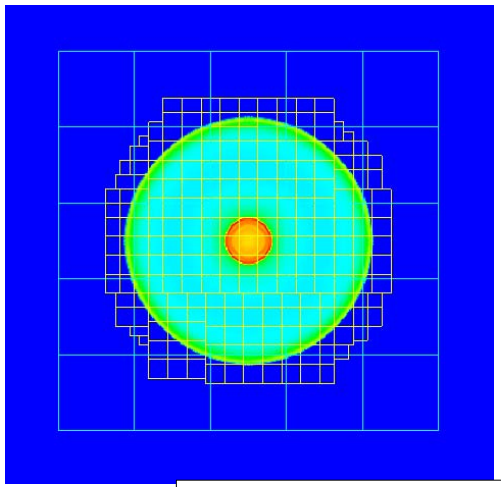
- Сегодняшний подход: в основном, “писать с нуля”
  - Исключение: многие разработчики программ адаптивного измельчения сетки используют инфраструктуру:
    - Boxlib, SAMRAI, Chombo, AMRCLAW, ChomboCLAW, Overture, AMRita, AMROC, GrACE (прочие?)
- В идеале нужен подход, использующий имеющиеся программы и оставляющий их без изменений (в максимально возможной степени)
- Рабочая группа произвольных лагранжево-эйлеровых методов и адаптивного измельчения сетки (ALE-AMR) в Центре прикладных научных вычислений (CASC) использует введение адаптивного измельчения сетки в написанные на языке C программы произвольных лагранжево-эйлеровых методов (CALE), с помощью инфраструктуры SAMRAI, не внося при этом изменений в сами программы
  - В настоящее время имеется надстройка к программам CALE для адаптивного измельчения сетки применительно к лагранжевым или эйлеровым задачам с несколькими материалами, задачам прочности и сгоранию ВВ
  - Используется идея, близкая к написанию сопрограмм
    - Сопрограммы могут *моделироваться* с помощью потоков с явным графиком (потоки как средство разработки программного обеспечения)
  - Накладные расходы на разработку:
    - Порядка 100 строк (225 кбайт) программ, включаемых в оригинальные исходные тексты
    - Порядка 10000 новых строк “отдельной” программы, написанной с помощью инфрастуктуры SAMRAI
  - Предстоит большая работа

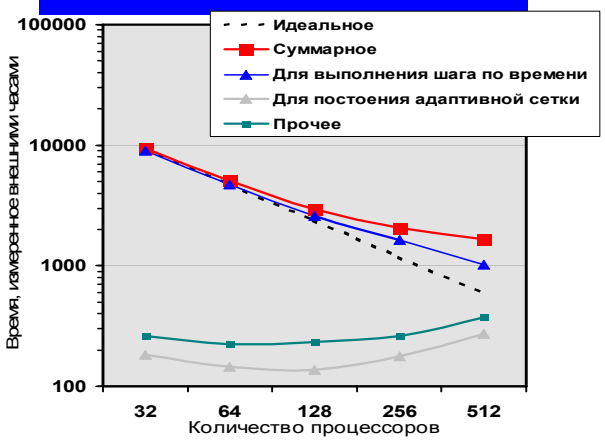
Примеры с использованием инфраструктуры SAMRAI демонстрируют как хорошее масштабирование, так и низкие накладные расходы на адаптивное измельчение сетки

Тест без масштабирования

4-уровневая задача Седова

Компьютер ASCI IBM Blue Pacific

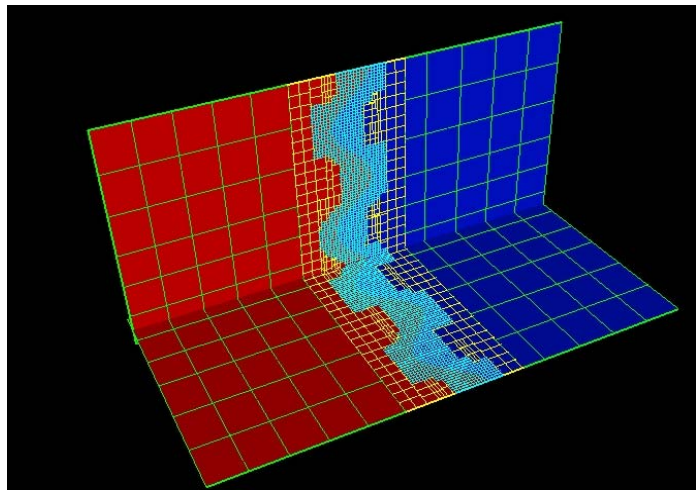


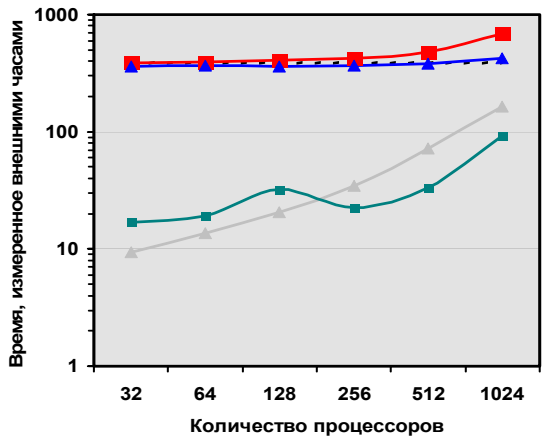


Тест с масштабированием

3-уровневая линейная адвекция

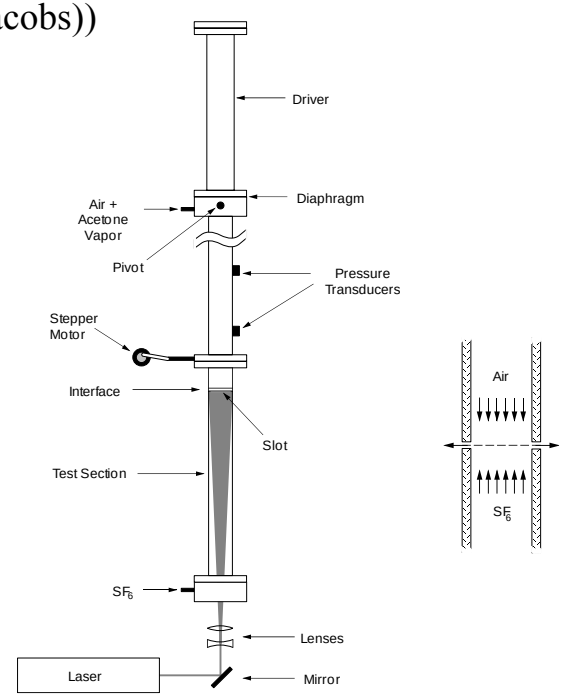
Компьютер Linux MCR Cluster





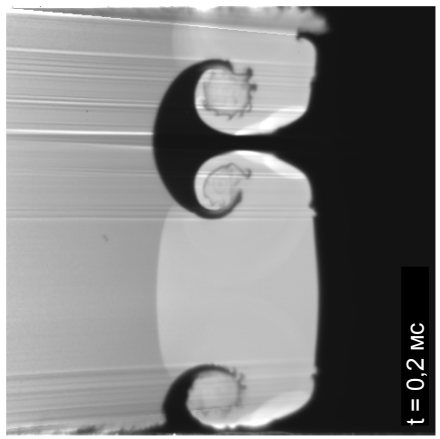
Расчет отраженной ударной волны с помощью программы Raptor (Boxlib) при M=1,3, воздух/SF_6

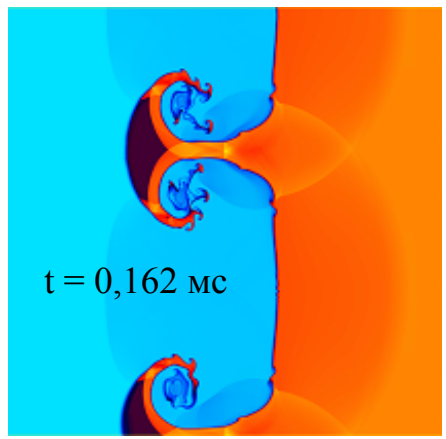
Вертикальная ударная труба в Университете штата Аризона (проф. Джефф Джейкобс (Jeff Jacobs))



Адаптивное измельчение сетки хорошо приспособлено для расчета отраженной ударной волны

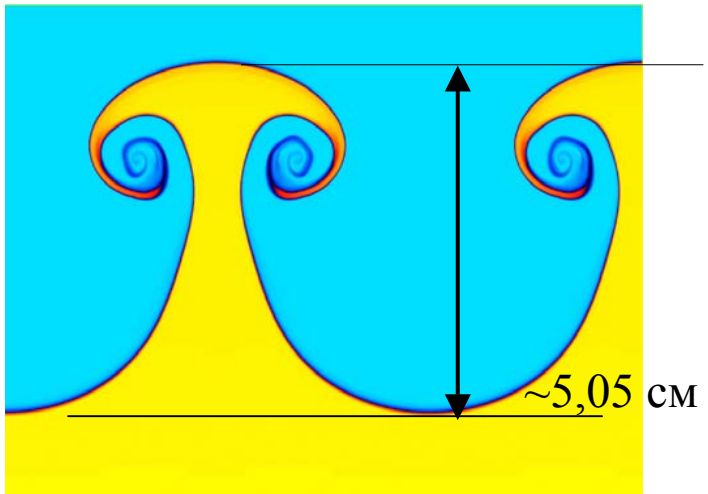
• Не обращайте внимания на небольшое смещение по времени из-за экспериментального фальш-днища



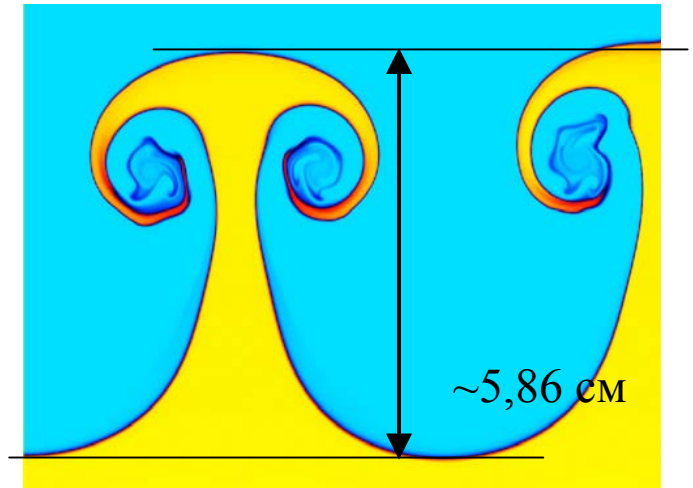


Исследование чувствительности измельчения сетки к начальной амплитуде при M=1,2 с помощью программы Raptor

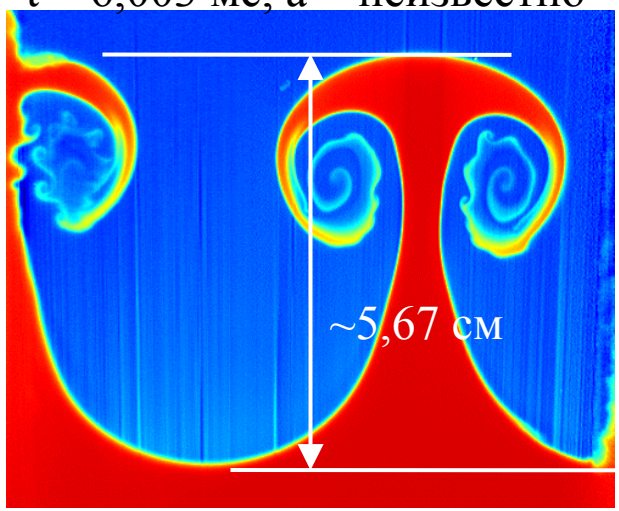
t = 6,276 мс, a = 0,221



t = 6,276 мс, a = 0,3



t = 6,003 мс, a = неизвестно

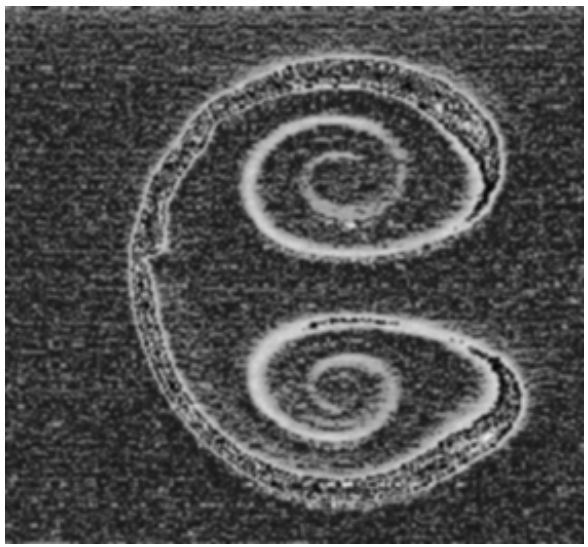


Рисунки предоставлены проф. Джеффом Джейкобсом, Университет штата Аризона

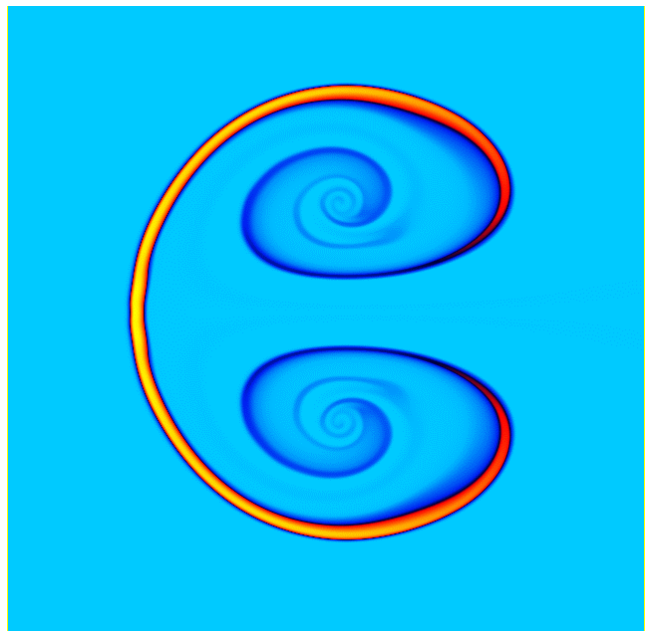
Эксперимент по взаимодействию ударной волны со струей SF_6. Джейкобс, “Физика жидкостей, А” (Jacobs, Phys. Fluids A), т. 5, № 9, 1993

- Слабая ударная волна с M=1,095 пропускается через цилиндрический столб SF_6
- Проводится визуализация поперечного сечения с помощью плоскостной лазерной флуоресценции (PLIF)

Эксперимент

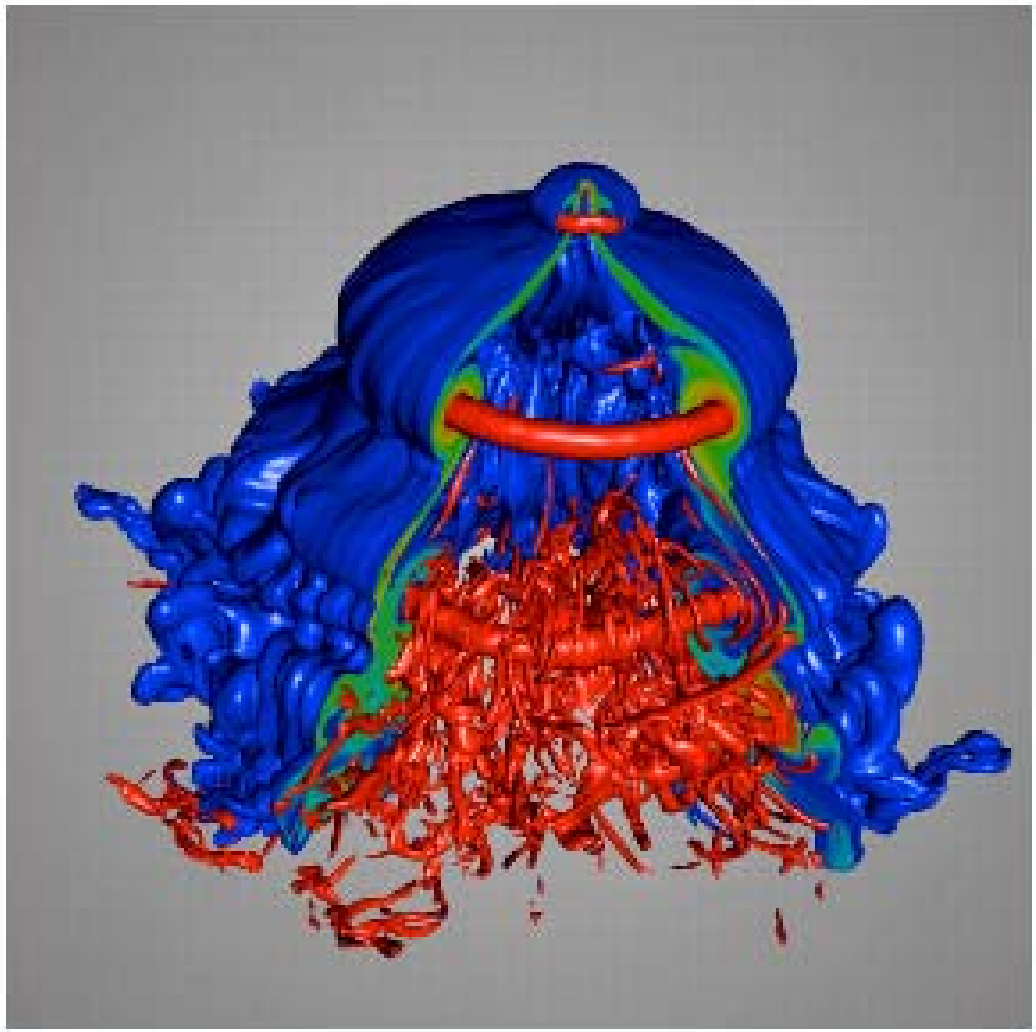


Расчет с помощью программы Raptor



Трехмерный расчет взаимодействия пузыря с ударной волной с помощью программы Raptor был выполнен за 7 часов на компьютере ALC Linux cluster

- Ударная волна с M= 2,88 падает на пузырек аргона
- R_{120}
- Разбиение за зоны с шагом около 200 мкм
- красный: величина завихренности
- синий: концентрация аргона
- разрез: концентрация мыльной пленки

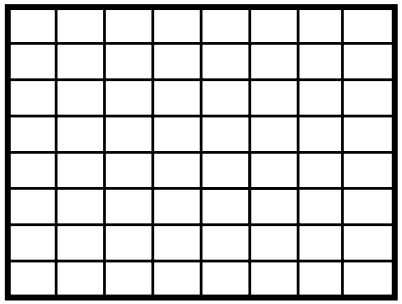


Найдерхаус, Оэкли, Андерсон, Рэнджэн, Бонацца и Грино, «Физика жидкостей» (Niederhaus, Oakley, Anderson, Ranjan, Bonazza, and Greenough, Physics of Fluids), на рецензии

Основная архитектура

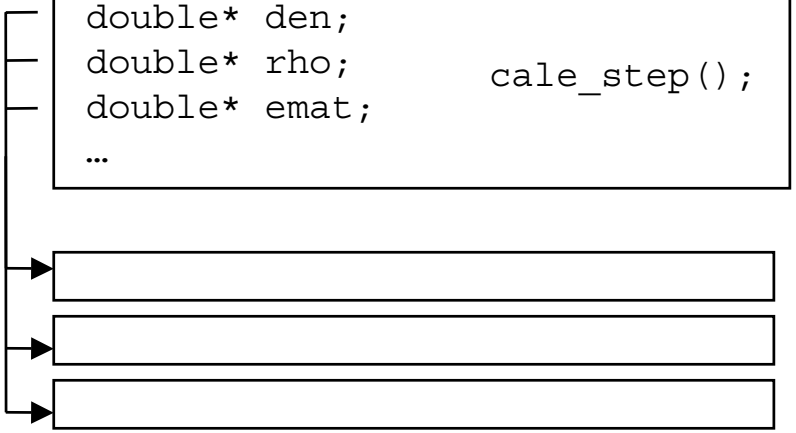
CALE

Единый блок
Структурированная сетка
1 ЦПУ



```
double* den;
double* rho;
double* emat;
…
```

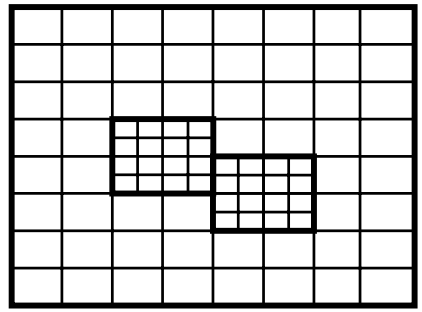
```
cale_step();
```



```
timestep loop {
  cale_step();
}
```

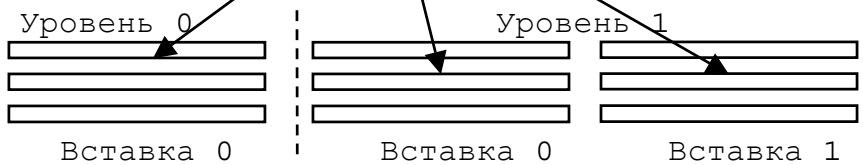
CALE с адаптивным измельчением сетки

Несколько вставок
Несколько уровней
N ЦПУ



```
double* den;
double* rho;
double* temp;
…
```

```
cale_step();
```




```
timestep loop {
  communicate_bdry(); // <- SAMRAI
  patch loop {
    set_patch_environment();
    cale_step();
  }
}
```

Зависимости от внутренних данных CALE

При разборе исходных текстов CALE обнаружена трудная задача:

```
timestep loop {
  communicate_bdry(); // <- SAMRAI
  patch loop {
    set_patch_environment();
    cale_step {
      … deeply nested with lots of stack-based state
      compute_new_u();
      communicate_new_u?(); // new u doesn't exist yet!
      compute_q(new_u); // <- requires bdry new_u
      …
  }
}
```

- Введенные Кнутом (1963) “сопрограммы” хорошо решают эту задачу
  - Позволяют выполнять “запуск” и “останов” по желанию без “утраты точки нахождения”
  - Требуются для поддержки нескольких вставок на одном процессоре
- Ни C, ни C++ не поддерживают сопрограмм
- Сопрограммы могут *моделироваться* с помощью потоков с явным графиком (потоки как средство разработки программного обеспечения)

Синхронизация, обеспечиваемая сопрограммами

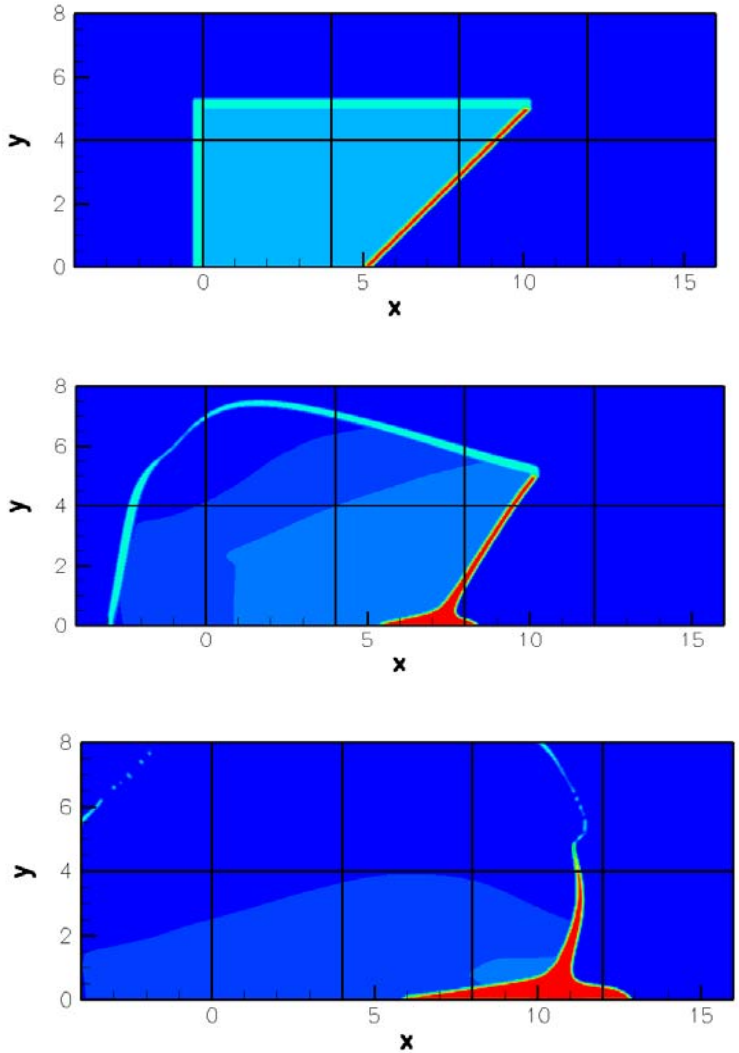
```
xcale {
   laghydro {
     accel();
     coroutine_yield();
     strain();
     strsheat();
     work();
   }
   coroutine_yield();
   advec();
…
```

```
for_each(patch) {
  set_patch_environment(n);
  coroutine_spawn(n);
}
// sync vel for stress/strain calc
communicate_bdry(rvel,zvel);
for_each(patch) {
  set_patch_environment(n);
  coroutine_yield(n);
}
// sync soln for advection
communicate_bdry(full_soln);
…
```

- Мощное средство синхронизации программы, первоначально написанной как последовательная
  - Минимальное вмешательство в имеющийся основной исходный текст
  - Полная общая передача управления
  - Среда, обеспечивающая исполнение без потерь
    - стек, временные данные и т.д.

Демонстрация параллельных вычислений: кумулятивный заряд

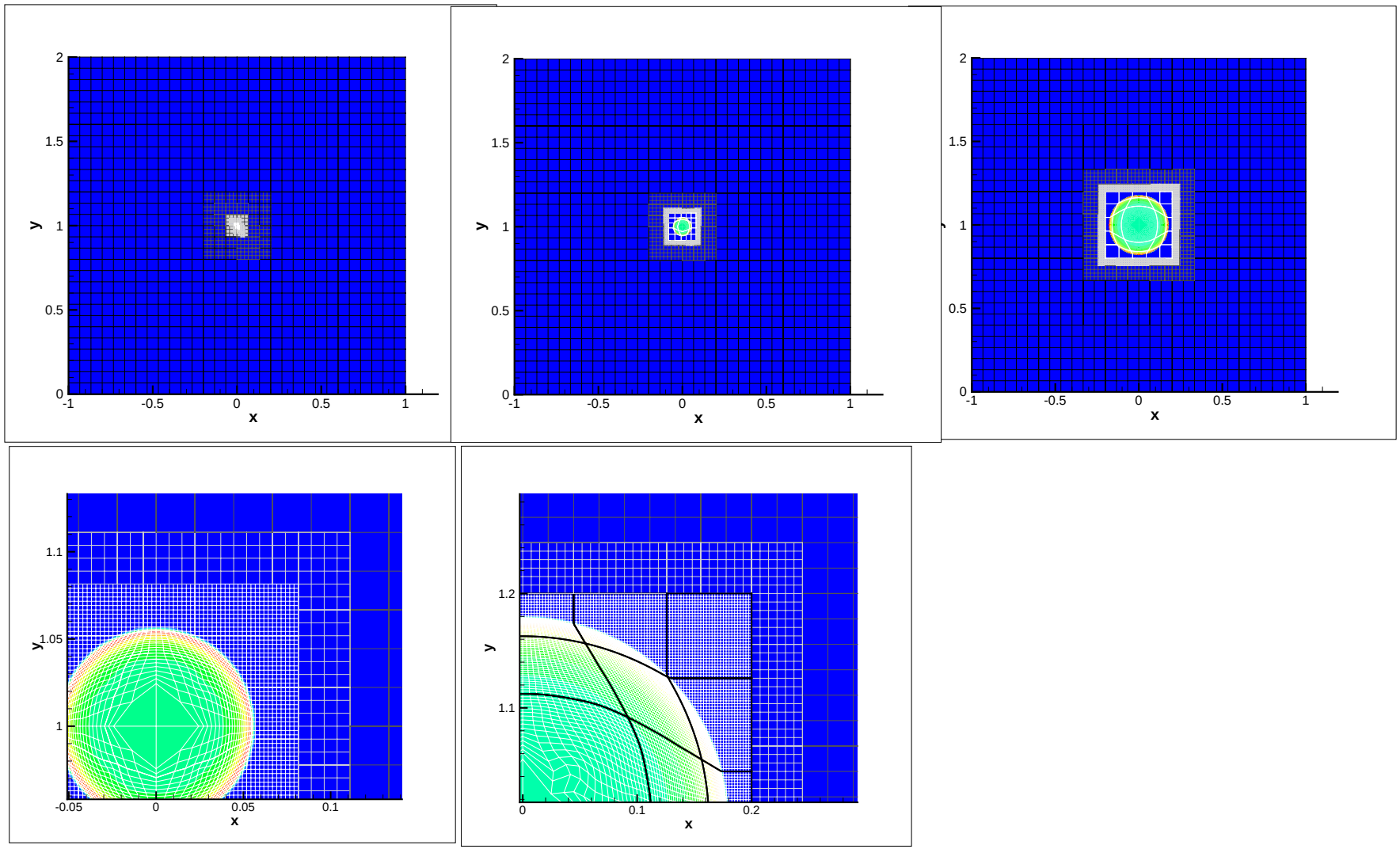
- Физические явления, учитываемые в CALE:
  - Эйлерова гидродинамическая модель
  - Смешанные материалы
    - Al, Cu, ВВ, воздух
  - Прочность
  - Сгорание ВВ
- Обычная входная колода CALE
  - jet02
- Мультипрограммный ресурс (mcr) в составе 10 процессоров



Демонстрация параллельного адаптивного измельчения сетки

- Лагранжева гидродинамическая модель CALE: простая взрывная волна
  - < 50 строк измененной программы CALE
- Обычная входная колода CALE без изменений
  - Дополнена входной колодой SAMRAI для параллельных вычислений и параметрами адаптивного измельчения сетки
- 16 процессоров (мультипрограммный ресурс)
  - динамическое выравнивание объема вычислений при каждом переформировании сетки
- 4 уровня сетки, коэффициент 3x3
  - Основная сетка 30x30
  - На самом мелком уровне сетка эквивалентна 810x810
- Критерий измельчения: конечная разность 2-го порядка для давления

Демонстрация параллельного адаптивного измельчения сетки

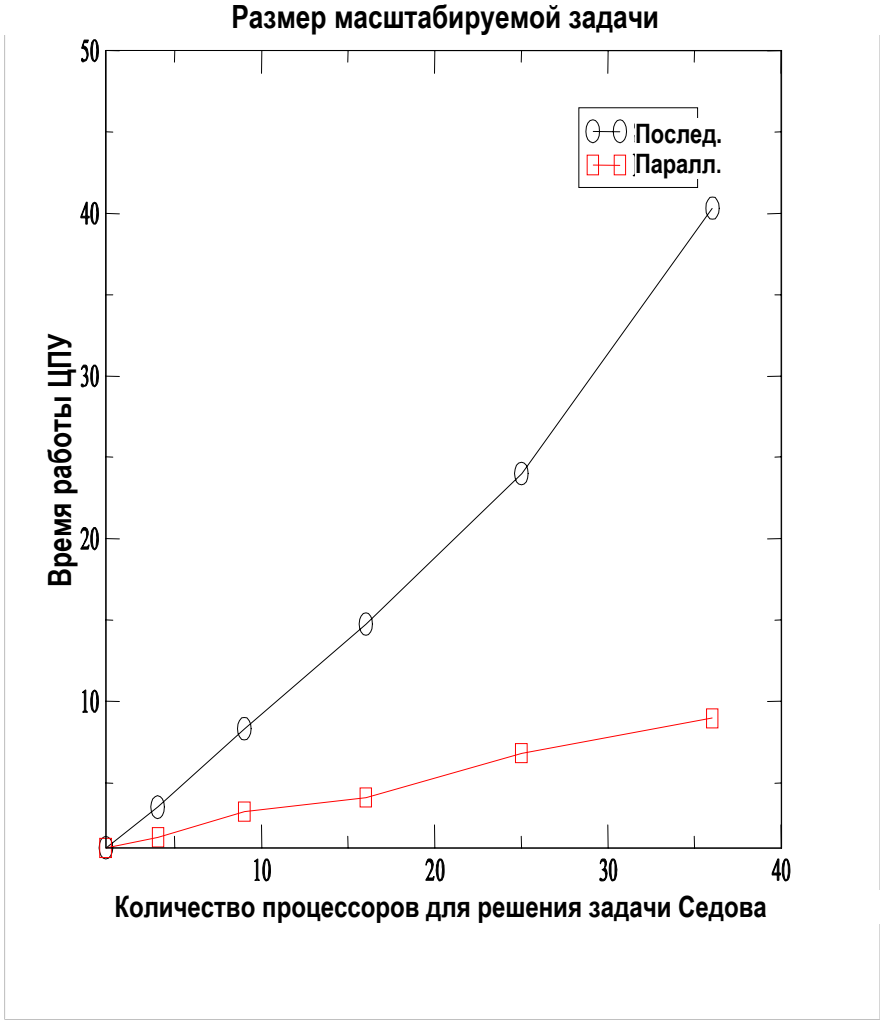


Накладные расходы на выполнение вычислений

- Накладные расходы при последовательных вычислениях: базовая проверка сопряжений в библиотеке
  - задача Седова: 650x325 для 50 шагов
  - CALE: 30 с, 97% физических явлений
  - CALE с адаптивным измельчением сетки: 31 с
    - Накладные расходы на контекстное переключение (сопрограммы) близки к 0
  - Память: удваивается в начальный момент, затем близко к 0
- Накладные расходы при параллельных вычислениях:
  - Обмен данными между зонами требует осторожности и некоторой оптимизации
  - Инфраструктура SAMRAI не представляет преграды для эффективного обмена данными.
- Накладные расходы на разработку: порядка 100 строк (225 кбайт) программ, включаемых в оригинальные исходные тексты
  - Для передачи точек синхронизации, фиксированные контуры для “фантомных зон”
  - Порядка 10000 новых строк “отдельной” программы, относящихся к инфраструктуре SAMRAI
- Ускорение параллельных вычислений?

Накладные расходы на выполнение вычислений

Параллельное масштабирование: гидродинамическая модель для задач с несколькими материалами, прочности и сгорания ВВ



Текущее положение дел

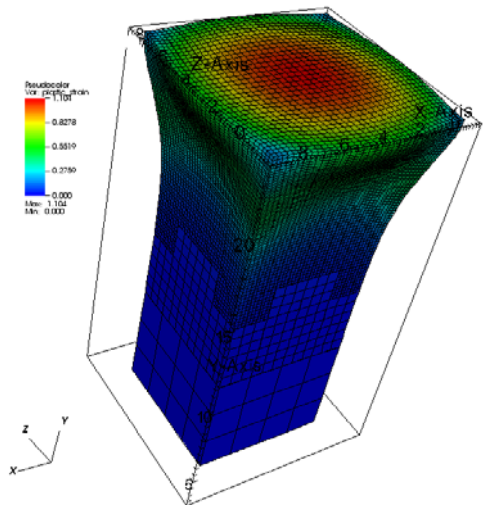
- Проверено с абсолютной точностью (дифференцируемое решение):
  - При параллельных вычислениях:
    - чистая лагранжева гидродинамическая модель для задач с несколькими материалами
    - смешанная лагранжева гидродинамическая модель (t=0)
- Проверено в плане “общей правильности”:
  - входная колода jet02 (кумулятивный заряд)
    - эйлеров подход для задач с несколькими материалами (фиксированная сетка)
    - прочность
    - сгорание ВВ
  - Остается вопрос с аппроксимацией 1-го порядка для адвекции на границах вставок
- Возможности по адаптивному измельчению сетки – пространственное измельчение
  - относительно простые межуровневые операторы
  - чистая лагранжева гидродинамическая модель

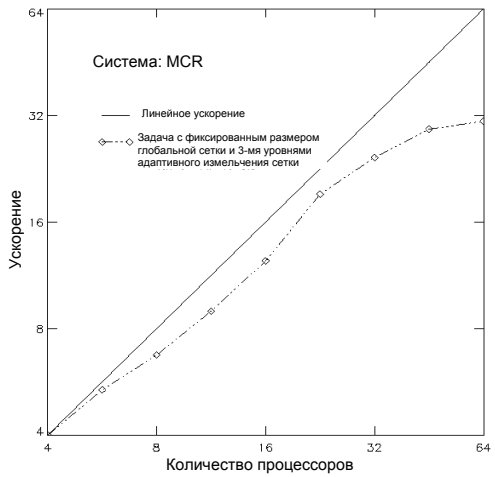
Текущие ограничения

- Отсутствуют возможности для:
  - параллельной формулировки задач
  - попеременного лагранжева или эйлерова подхода (либо лагранжев, либо эйлеров)
  - учета линий скольжения
  - редактирования
  - интерактивной графики (?)
  - включения прочих физических явлений нелокального характера: магнитной гидродинамики и т.д.
- С другой стороны, многое практически “бесплатно”
  - уравнения состояния
  - сгорание ВВ
  - источники
  - учет физических явлений локального характера в целом тривиален
  - Программное обеспечение VisIt для визуализации

Эластопластичные материалы

- Условие фон Мизеса для сдвига
- Новые соображения для адаптивного измельчения сетки:
  - межуровневые операторы для тензорных величин
  - физические соображения, связанные с интерполяцией и укрупнением сетки:
- Текущие вопросы для проведения исследований
  - интерполяция вблизи эластопластического перехода
  - нарушение условий сдвига
  - отрицательные пластические деформации
- Включение библиотеки моделирования материалов Р. Беккера (R. Becker)





Цели для разработки операторов межуровневой передачи данных (укрупнение и измельчение)

- Сохранение плотности, скорости и внутренней энергии в свободной струе
- 2-й порядок точности (линейная реконструкция)
- Монотонность
- Местная консервативность
- Проведение путем укрупнения преобразования, в точности обратного проведенному при измельчении

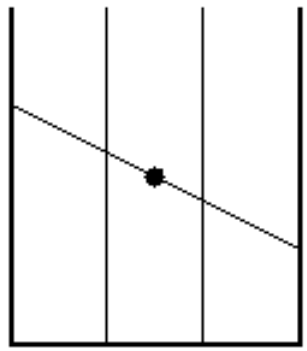
При измельчении используется линейная интерполяция в примитивных и объемно-массовых системах координат

- Обеспечивает
  - Консервативность
  - Монотонность и сохранение примитивных переменных вдоль свободной струи
- Общие одномерные формы:

$$\rho_k = \rho_o + \frac{\Delta\rho_o}{V_o}\left(\bar{v}_i - \frac{1}{2}V_o\right)$$

$$u_k = u_o + \frac{\Delta u_o}{\sum \hat{m}_i}\left(\hat{\bar{m}}_i - \frac{1}{2}\sum \hat{m}_i\right)$$

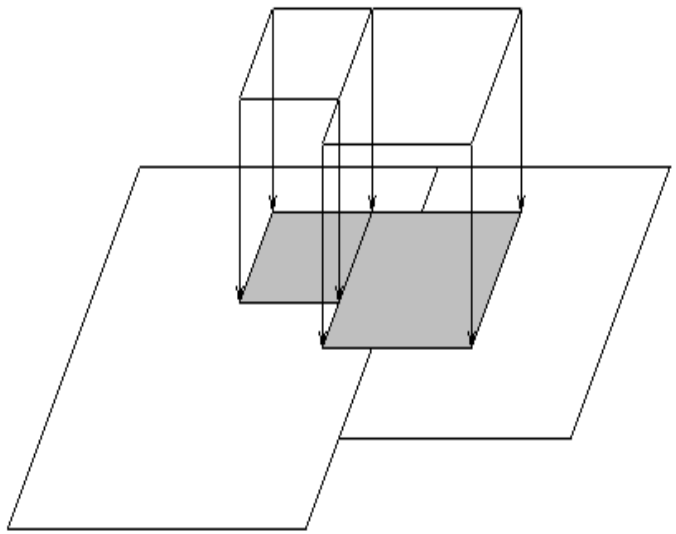
$$e_k = e_o + \frac{\Delta e_o}{M_o}\left(\bar{m}_i - \frac{1}{2}M_o\right)$$



- Расщепленные или не расщепленные по пространственным переменным расширения на два и три измерения

При укрупнении используется проекция узлов и простое взвешенное объемное или массовое осреднение

- Сетка формируется выбором каждой r-й точки сетки
- Взвешенное осреднение характеристик потока устанавливается таким образом, чтобы определить оператор, обратный оператору измельчения:

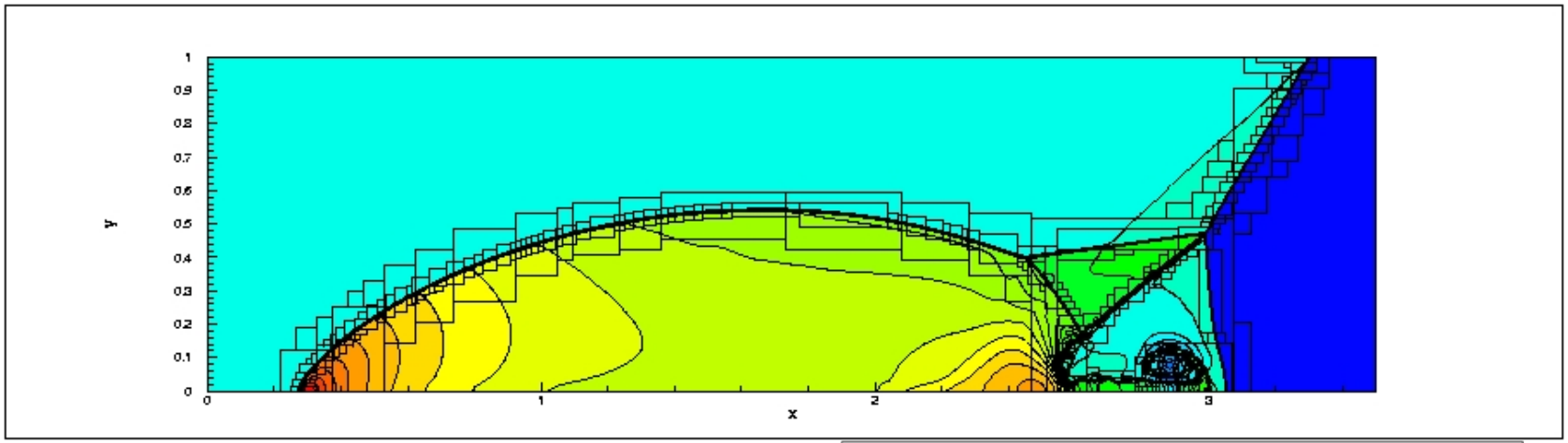
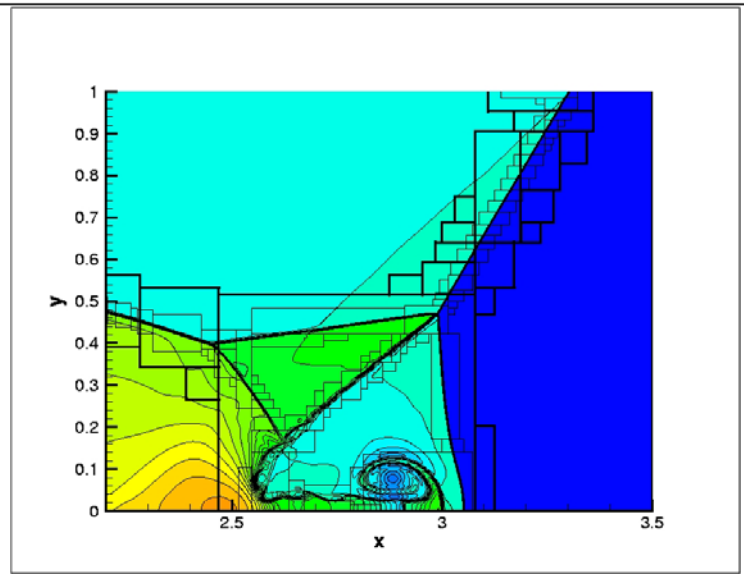


$$\rho_o = \frac{\sum \rho_i V_i}{\sum V_i}$$

$$u_o = \frac{\sum u_i \hat{m}_i}{\sum \hat{m}_i}$$

$$e_o = \frac{\sum e_i m_i}{\sum m_i}$$

Эйлеров расчет двойного маховского отражения при числе Маха 10 с помощью произвольной лагранжево-эйлеровой схемы с измельчением сетки (ALE-AMR)



- **Показана: плотность при *t*=0,21 для уровня 3 расчетов двойного маховского отражения при числе Маха 10 с помощью произвольной лагранжево-эйлеровой схемы с измельчением сетки (ALE-AMR)**