

Развитие параллельной версии FlowVision HPC

Аксенов А.А.¹, Дядькин А.А.¹, Сушко Г.Б.^{1,2}, Харченко С.А.^{1,3}

¹ ООО «ТЕСИС», Москва; ² МФТИ, Москва; ³ ВЦ РАН, Москва

Введение. Современные задачи вычислительной гидродинамики требуют моделирование нестационарного движения жидкости в расчетных областях, имеющих подвижные границы, а также моделирование контактных границ раздела фаз (свободная поверхность) сложной формы. Для адекватного воспроизведения тонких физических эффектов в геометрически сложных трехмерных областях требуются подробные расчетные сетки, содержащие от сотен тысяч до сотен миллионов расчетных ячеек. Подобные задачи требуют огромных вычислительных ресурсов и могут быть решены только на самой современной параллельной вычислительной технике.

В программном комплексе **FlowVision** используется наиболее универсальный метод расчета задач вычислительной гидродинамики с использованием эйлеровых адаптивных расчетных сеток для областей произвольно сложной формы. Моделирование свободной поверхности осуществляется во **FlowVision** с использованием подхода VOF (volume of fluid). Моделирование течений при наличии в расчетной области подвижных границ требует специальной методологии расчета.

Для уменьшения времени счета проводимые вычисления требуется распараллелить в соответствии с архитектурой используемой вычислительной техники. Доступная в настоящий момент техника обладает как правило архитектурой с неоднородным доступом к памяти. А именно, с одной стороны, имеется набор вычислительных узлов с распределенной памятью, обмен данными между которыми может быть осуществлен по быстрой обменной сетке. С другой стороны, каждый узел представляет собой многопроцессорный/многоядерный компьютер с общим доступом к оперативной памяти.

В работе рассматриваются вопросы организации параллельных вычислений в условиях наличия в расчетной области подвижных границ и свободных поверхностей. Приводятся результаты численных экспериментов по масштабируемости предложенных алгоритмов на примере моделирования индустриальной задачи вычислительной гидродинамики при помощи **FlowVision**.

Эффективная хорошо масштабируемая организация параллельных вычислений при большом общем числе доступных процессоров/ядер предполагает хорошую балансировку вычислений на различных процессорах между собой. Чем больше число задействованных процессоров/ядер, тем более точная балансировка вычислений требуется для поддержания заданного уровня масштабируемости. В работе описывается параллельная реализация алгоритма решения СЛАУ для компьютеров с неоднородным доступом к памяти. Для уменьшения времени вычислений этот алгоритм заранее распараллеливается по распределенной памяти (по узлам вычислений) с использованием стандарта MPI, а по общей памяти узла распараллеливание по процессорам/ядрам осуществляется с динамическим распределением нагрузки на основе технологии Intel® Threading Building Blocks (TBB). Приводятся результаты численных экспериментов по масштабируемости предложенных алгоритмов на СКИФ МГУ «Чебышев» для систем линейных уравнений, возникающих при моделировании во **FlowVision** задач вычислительной гидродинамики.

1. Моделирование при наличии подвижных элементов геометрии

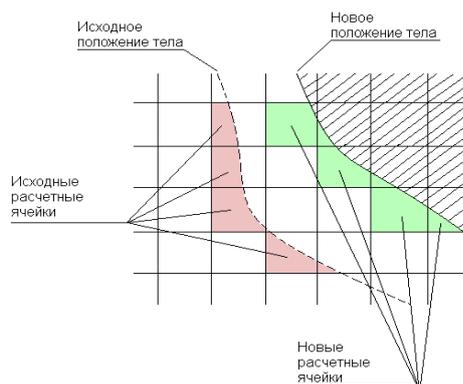


Рис. 1 – Изменение геометрии ячеек сетки и топологии связей между ними при сдвиге подвижного тела

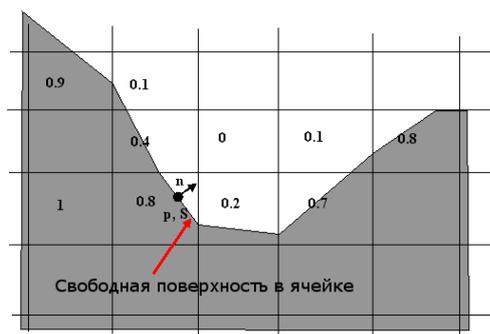


Рис. 2 – Параметр VOF для набора ячеек и свободная поверхность

Во **FlowVision** реализовано моделирование течений жидкости/газа при наличии в расчетной области поддлит к тому (см. Рисунок 1), что часть ячеек, являвшихся расчетными, перестает быть таковыми и попадает внутрь подвижного тела, и наоборот; часть ячеек меняет свою форму, часть ячеек адаптируется, измельчается или укрупняется в соответствии с критериями адаптации, и т.д. Все эти изменения расчетной сетки меняют геометрию ячеек и топологию связей между ними. Кроме того, используются специальные методики, по которым данные из исходных расчетных ячеек преобразовываются в данные новых расчетных ячеек.

Во **FlowVision** также реализовано моделирование так называемой *свободной поверхности* жидкости на границе раздела сред жидкость/газ.

Идея реализации моделирования свободной поверхности состоит в ведении для каждой расчетной ячейки параметра VOF (Volume Of Fluid). Этот параметр представляет собой долю жидкости в заполнении объема текущей ячейки, поэтому $0 \leq VOF \leq 1$. Соответственно, если $VOF = 1$, то текущая ячейка содержит только жидкость. Если $VOF = 0$, то текущая ячейка содержит только газ. Если $0 < VOF < 1$, то в текущей ячейке имеется и жидкость, и газ, и между ними в текущей ячейке имеется граница раздела, именуемая свободной поверхностью. Во **FlowVision** предполагается, что граница раздела в текущей ячейке всегда имеет простейший возможный вид – это плоскость, рассекающая ячейку на две части. Имея значения параметра VOF (см. Рис. 2) для всех расчетных ячеек, можно в области расчета восстановить форму поверхности, разделяющую жидкую и газообразную фазы.

Во **FlowVision** для вычисления геометрии ячеек, через которые проходит свободная поверхность, используется методика подсеточного разрешения геометрии. Это позволяет осуществлять перенос VOF с использованием высокоточного метода. При этом во **FlowVision** уравнения решаются и в ячейках со свободной поверхностью.

2. Параллельные вычисления и подвижные элементы геометрии. Наличие в расчетной области подвижных границ расчетной области или свободной поверхности приводит к необходимости периодически перестраивать расчетную сетку. Это значит, что число ячеек в фиксированной области пространства может меняться с течением времени и, следовательно, необходимо периодически корректировать распределение данных по процессорам для поддержания эффективности параллельных вычислений.

Подвижный элемент геометрии в задачах аэро- гидродинамики часто представляет собой некоторое твердое или деформируемое трехмерное тело (обозначаемое ниже как подвижное тело), которое движется в расчетной области по заданному закону или под действием гидродинамических и/или других сил. С точки зрения вычисления гидродинамических величин, таких как давление жидкости, скорость потока жидкости, и т.д., во внутренней части подвижного тела расчетные ячейки отсутствуют. При этом собственно ячейки геометрии внутри подвижного тела во **FlowVision** всегда строятся и должны сохраняться, поскольку при достаточно сильном сдвиге подвижного тела эти ячейки могут стать расчетными ячейками сетки.

Для параллельных вычислений все расчетные ячейки должны быть балансируемо распределены по процессорам с учетом геометрической локальности данных. Кроме того, при небольших движениях подвижного тела имеет смысл в основном сохранить декомпозицию задачи по процессорам даже в условиях некоторого перестроения расчетной сетки. Тем не менее, при относительно больших движениях подвижного тела наоборот необходимо перестроить декомпозицию задачи по процессорам и перенести все необходимые данные на соответствующие процессоры. Иначе возможен сильный дисбаланс вычислений на разных процессорах. Во **FlowVision** предполагается, что шаг по времени достаточно мал, а потому передвижение подвижного тела за один шаг по времени не приводит к слишком значительному перестроению расчетной сетки. По этой причине представляется естественным при построении новой декомпозиции задачи по процессорам осуществить ее явную привязку к предыдущей декомпозиции.

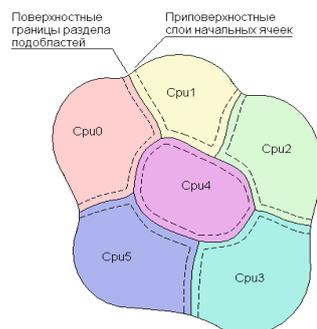


Рис. 3 – Поверхностные границы раздела и приповерхностные слои ячеек

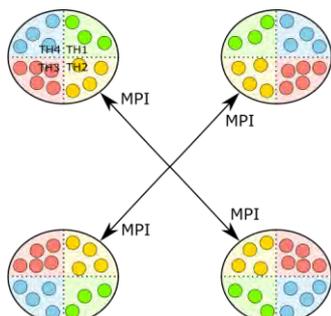


Рис. 4 – Организация вычислений на компьютерах с неоднородным доступом к памяти

Идея алгоритма построения новой декомпозиции состоит в том, чтобы изменить распределение по процессорам некоторой части ячеек, оставив при этом совпадающим с исходным распределением по процессорам большую часть начальных ячеек. Для этого рассмотрим поверхности, разделяющие всю расчетную область на подобласти для разных процессоров, см. Рисунок 3. Все внутренние части подобластей каждого процессора остаются данными этого процессора. Только для нескольких приповерхностных слоев ячеек разрешается изменить их процессорную принадлежность.

Вычисление новой декомпозиции для числа процессоров – степени двойки можно осуществлять последовательным билинием подобласти попо-

лам. А именно, на каждом этапе декомпозиции строим декомпозицию части области на две группы процессоров. Для этого объединяем в одну огромную суперячейку все начальные ячейки каждой из двух групп процессоров, за исключением тех ячеек, что примыкают к границе между этими группами процессоров в предыдущей декомпозиции. Проводим декомпозицию этих приповерхностных начальных ячеек и двух суперячеек на два набора ячеек с учетом балансировки числа реальных ячеек, и с учетом минимизации связей между наборами. При этом две огромные суперячейки должны попасть каждая в свой определенный набор ячеек. Таким образом, с каждым получившимся новым набором начальных ячеек ассоциируется некоторая подгруппа процессоров. И т.д.

3. MPI+threads реализация алгоритма решения систем уравнений.

На компьютерах с неоднородным доступом к памяти возможны различные способы организации вычислений. В данной работе при распараллеливании вычислений между узлами для организации обменов и синхронизаций используется стандарт обмена сообщениями MPI. В каждом узле с общей памятью при распараллеливании по процессорам/ядрам для организации синхронизаций по общей памяти используется технология Intel® Threading Building Blocks (TBB). При этом предполагается, что на каждом узле имеется только один MPI процесс, который затем порождает на этом узле нужное количество одновременно работающих потоков вычислений. При этом все вычисления заранее распределяются по узлам таким же образом, как это делалось при работе только с распределенной памятью. Затем, каждое вычисление в узле динамически распараллеливается по процессорам/ядрам с использованием эффекта самобалансировки вычислений внутри узла. Для этого полное число объектов, с которыми предполагается проводить вычисления внутри одного узла, должно быть достаточно большим. Это необходимо для обеспечения возможности организации вычислений по принципу клиент/сервер с динамическим выбором того ядра, на котором будет проводиться текущее вычисление.

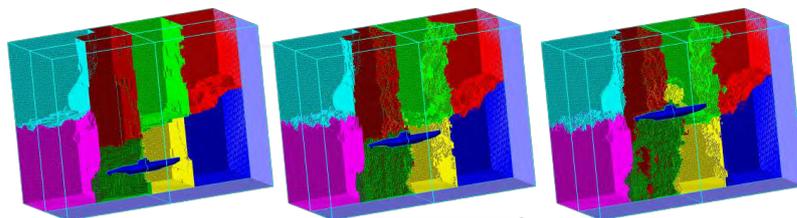


Рис. 5 – Декомпозиция задачи I на 8 процессоров для последовательности положений подводной лодки

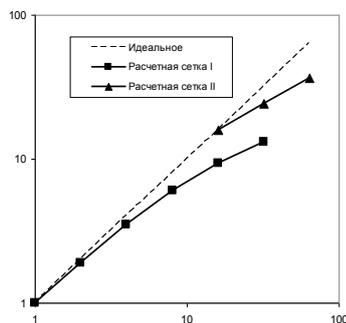


Рис. 6. Ускорение при решении задачи о подводной лодке

4. Результаты численных экспериментов.

Для проведения численных экспериментов была выбрана задача о всплытии подводной лодки модельной геометрии. Лодка движется с горизонтальной скоростью 4 м/с, и всплывает со скоростью 0.1 м/с. При решении данной задачи требуется определить силу сопротивления воды, а также структуру гидродинамического течения вокруг лодки.

Моделирование тестовой задачи проводилось на двух расчетных сетках, сетке I и сетке II, с адаптацией к поверхности лодки. Расчетные сетки отличались между собой шагом по пространству начальной сетки, сетка II более подробная по сравнению с сеткой I. Полное число расчетных ячеек в задаче для сетки I при начальном положении лодки было равно 1.522.292 ячеек, для сетки II – 7.052.948. По мере всплытия лодки расчетные сетки и декомпозиция перестраивались. При этом адаптация сетки снова производилась к текущей поверхности лодки, а снизу всплывающей лодки проводилась разадаптация ячеек сетки, поэтому полное число ячеек не существенно изменялось при всплытии. Декомпозиция расчетной области перевычислялась при каждом изменении положения лодки.

Декомпозиция расчетной области для сетки I на 8 процессоров для последовательности положений лодки приведена на Рисунке 5. На Рисунке 6 хорошо видна связь между декомпозициями задачи при сдвиге положения лодки. На Рисунке 6 показана масштабируемость при решении тестовой задачи на сетках I и II вплоть до 64 процессоров. Результаты по масштабируемости при решении тестовой задачи получены на суперкомпьютере СКИФ МГУ “Чебышев” в режиме расчета 2 процессора на узел.

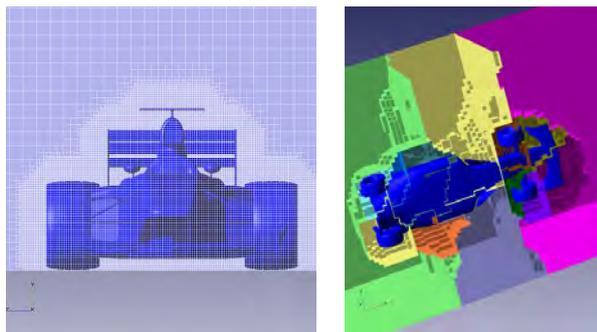


Рис. 7 – Адаптация сетки и декомпозиция расчетной области на 16 процессоров

Для проведения численных экспериментов по проверке масштабируемости комбинированного MPI+threads алгоритма решения систем уравнений была выбрана задача об обтекании спортивного автомобиля. Адаптивная расчетная сетка для этой задачи содержит 2.994.497 расчетных ячеек. Структура адаптации сетки вокруг автомобиля и объемная декомпозиция расчетной сетки на 16 процессоров показана на Рисунке 7. На Рисунке 8 показана структура разреженности матрицы зависимости блочных вычислений для 16 процессоров/ядер. Численные эксперименты проводились на СКИФ МГУ “Чебышев”. В Таблице 1 представлены

результаты численных экспериментов.

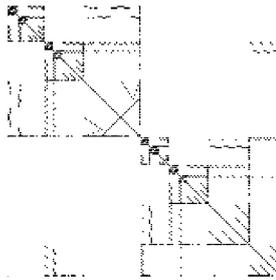


Рис. 8 Структура матрицы зависимостей блочных вычислений для 16-ти процессоров/ядер

Таблица 1. Результаты численных экспериментов.

| | 2 узла по 4 ядра | | 4 узла по 2 ядра | | 4 узла по 4 ядра | | 4 узла по 8 ядер | |
|------|------------------|---------|------------------|---------|------------------|---------|------------------|---------|
| | MPI | MPI+TBV | MPI | MPI+TBV | MPI | MPI+TBV | MPI | MPI+TBV |
| Fct | 14,4 | 9,32 | 9,71 | 9,16 | 7,63 | 5,01 | --- | 3,98 |
| Iter | 102,27 | 54,13 | 51,69 | 54,39 | 53,08 | 29,14 | --- | 31,74 |

Заключение

В работе представлены параллельные алгоритмы, позволяющие эффективно организовать вычисления в случае если в расчетной области присутствуют подвижные тела и свободные поверхности. Рассмотрена также комбинированная MPI+threads параллельная реализация алгоритма решения систем уравнений. На примере тестовых гидродинамических задач показана высокая эффективность предложенных в работе алгоритмов.