

Распараллеливание пространственной модели переноса примеси на системах с распределенной памятью с помощью одно- и двумерной декомпозиции области*

Д.А. Беликов, А.В. Старченко

В работе представлено распараллеливание вычислительного алгоритма решения системы адвективно-диффузионно-кинетических уравнений математической модели переноса малых составляющих примеси с учетом их химических взаимодействий в приземном слое атмосферы. Распараллеливание основано на одно- или двумерной декомпозиции расчетной области с использованием синхронных операций для межпроцессорной передачи данных. Производительность разработанного параллельного алгоритма при различных способах декомпозиции и организации обменов оценена экспериментально на кластере Томского государственного университета СКИФ Cyberia.

1. Введение

Подробное моделирование протекания сложных атмосферных процессов в региональных масштабах требует значительных вычислительных затрат. При этом затраты со временем будут только возрастать, поскольку развитие моделей подобного класса идет по пути повышения пространственного разрешения и включения в рассмотрение более широкого спектра атмосферных явлений [1].

Эффективным способом сокращения времени расчета, активно развиваемым в настоящее время, является применение вычислительных систем с параллельной архитектурой. Наиболее общим подходом при создании параллельных программ на системах с распределенной памятью является стандарт MPI (Message Passing Interface), разработанный в 1993 году в результате обобщения и стандартизации различных библиотек передачи сообщений [2]. Однако его практическое применение сопровождается необходимостью решения не всегда тривиальных задач, а для достижения положительного результата (получения параллельных программ с эффективностью, близкой к единице) требуется учитывать особенности алгоритма решения задачи и вычислительной системы.

Универсальным подходом при реализации моделей атмосферного пограничного слоя и переноса примеси на вычислительных системах с параллельной архитектурой является распараллеливание по данным. При этом область исследования разделяется на подобласти в соответствии с числом доступных процессоров, и каждый процессор одновременно с другими производит вычисления в своей подобласти. Тогда объемы вычислений для всех процессов примерно равны и напрямую зависят от равномерности декомпозиции области исследования. Хотя при расчете химических реакций уровни нагрузки на процессы, работающие с подобластями, характеризующимися большим количеством источников примеси (урбанизированная территория), могут быть выше, чем у процессов, имеющих дело с подобластью с меньшим количеством источников (сельская местность) [1].

При использовании парадигмы распараллеливания по данным для моделирования пространственных процессов в атмосфере декомпозиция расчетной области может быть проведена по трем координатным направлениям. Однако разбиение области исследования по вертикальной координате представляется менее эффективной, поскольку в мезомасштабных метеорологических моделях или фотохимических моделях качества атмосферного воздуха необходимо вычислять интегральные характеристики в вертикальном столбе воздуха [1,4,5], что потребует межпроцессорных пересылок и негативно скажется на ускорении параллельного алгоритма. Поэтому двумерная декомпозиция по двум горизонтальным координатам является оптималь-

* Работа выполнена при поддержке РФФИ грант № 07-05-01126-а.

ным способом распараллеливания, поскольку обеспечивает масштабируемость и гибкость параллельной программы при проведении вычислений на современных многопроцессорных кластерных системах. Заметим, однако, что при небольшом числе используемых процессоров эффективность параллельных программ, построенных с применением одномерной и двумерной декомпозиции области исследования, отличается незначительно [6]. Это следует также принимать в расчет, поскольку трудоемкость создания параллельных программ для решения сеточных уравнений с использованием более простой одномерной декомпозиции значительно ниже.

Данная работа посвящена теоретическому анализу и апробации параллельных алгоритмов, основанных на одномерной и двумерной декомпозиции сеточной области, для численного решения уравнений пространственной прогностической модели переноса примеси с учетом химических реакций.

2. Прогностическая пространственная модель переноса примеси с учетом химических и фотохимических реакций

Для расчета концентрации компонентов примеси с учетом химических взаимодействий между ними применяется эйлерова модель турбулентной диффузии, включающая транспортные уравнения с описанием адвекции, турбулентной диффузии и химических реакций [5]:

$$\frac{\partial C_i}{\partial t} + \frac{\partial UC_i}{\partial x} + \frac{\partial VC_i}{\partial y} + \frac{\partial WC_i}{\partial z} = -\frac{\partial \langle c_i u \rangle}{\partial x} - \frac{\partial \langle c_i v \rangle}{\partial y} - \frac{\partial \langle c_i w \rangle}{\partial z} + S_i + R_i; \quad i = 1, 2, \dots, n_s \quad (1)$$

Здесь C_i, c_i – осредненная и пульсационная составляющие концентрации i -ой компоненты примеси; W, w – осредненная и пульсационная составляющие вертикальной компоненты скорости примеси; U, V, u, v – осредненные и пульсационные составляющие горизонтальной скорости; $\langle \rangle$ – осреднение по Рейнольдсу, S_i – источниковый член, представляющий выбросы компонентов примеси в атмосферу; R_i описывает образование и трансформацию вещества за счет химических и фотохимических реакций с участием компонентов примеси; n_s – количество химических компонентов примеси, принятых к рассмотрению в выбранной схеме химических реакций.

Уравнения (1) незамкнуты, так как в них кроме концентрации C_i , которую необходимо рассчитать, присутствуют дополнительные неизвестные – корреляции $\langle c_i u \rangle$, $\langle c_i v \rangle$, $\langle c_i w \rangle$, моделирующие турбулентную диффузию примеси. В данной работе для их определения используются замыкающие соотношения, которые получаются в рамках равновесного приближения для дифференциальных уравнений переноса турбулентных потоков массы в условиях локальной однородности атмосферного пограничного слоя [7].

В настоящее время в рассматриваемой фотохимической модели адаптировано три сокращенные кинетические схемы [5]: модуль AIRCHEM ($n_s = 10$) [8], механизм GRS ($n_s = 11$) [9], RADM ($n_s = 11$) [10].

3. Начальные и граничные условия

На нижней границе области исследования ставятся граничные условия, моделирующие сухое осаждение компонентов примеси [11]. На верхней границе для концентраций C_i применяются простые градиентные условия. На боковых границах расчетной области используются условия радиационного типа, обеспечивающие выход возмущений, сгенерированных в области (погрешность метода и округления), за её пределы без отражения. При инициализации модели считается, что во всей расчетной области концентрации рассматриваемых компонентов примеси равны начальным фоновым.

Расчет полей метеорологических характеристик осуществляется при помощи мезомасштабной метеорологической модели [4] или модели атмосферного пограничного слоя в одно-родном приближении [12].

4. Метод расчета

Дискретизация дифференциальных уравнений модели переноса примеси (1) осуществлена с использованием метода конечного объема на основе схем второго порядка аппроксимации для диффузионных (центрально-разностная схема) и адвективных (MLU-схема Ван-Лиры [13]) членов уравнений переноса на равномерной в горизонтальных направлениях (Ox и Oy) и неравномерной, сгущающейся к поверхности конечно-разностной сетке.

Дискретный аналог уравнения переноса (1) при фиксированном индексе $1 \leq i \leq n_s$, $S = S_i$, $R = R_i$, $\Phi = C_i$ можно записать в виде:

$$\begin{aligned} & \left[a_P^0 - (R^-)_P \Delta x \Delta y \Delta z + D_t + D_b \right] \Phi_P^{k+1} = D_t \Phi_T^{k+1} + D_b \Phi_B^{k+1} + \\ & \left[a_E \Phi_E^k + a_W \Phi_W^k + a_N \Phi_N^k + a_S \Phi_S^k + a_T \Phi_T^k + a_B \Phi_B^k - a_P \Phi_P^k \right] + b \end{aligned} \quad (2)$$

где

$$\begin{aligned} a_E &= \max(-F_e, 0) + D_e; a_W = \max(F_w, 0) + D_w; a_N = \max(-F_n, 0) + D_n; \\ a_S &= \max(F_s, 0) + D_s; a_T = \max(-F_t, 0); a_B = \max(F_b, 0); \\ b &= \left((S^+)_P^k + (R^-)_P^k \right) \Delta x \Delta y \Delta z + Dm_P^*; a_P^0 = \frac{\Delta x \Delta y \Delta z}{\Delta t}. \end{aligned} \quad (3)$$

Здесь $\Phi_P^{k+1} \approx \Phi(t^k + \Delta t, x_i, y_j, z_m)$, k – номер слоя по времени, $\Delta x \Delta y \Delta z$ – размеры конечного объема с центром в точке P (рис.1); центры соседних конечных объемов обозначены по направлениям сторон света $E(x_i + \Delta x, y_j, z_m)$, $W(x_i - \Delta x, y_j, z_m)$, $N(x_i, y_j + \Delta y, z_m)$, $S(x_i, y_j - \Delta y, z_m)$. Для центров расположенных выше конечных объемов используется $T(x_i, y_j, z_m + \Delta z)$, а ниже – $B(x_i, y_j, z_m - \Delta z)$. Центры соответствующих граней конечного объема обозначены n, s, e, w, t, b . Расходы и эффективные коэффициенты диффузии на соответствующих гранях конечного объема записаны, как F и D . Под обозначением Dm_P^* понимается конечно-объемная аппроксимация смешанных производных по x, y, z от Φ на временном слое k и сумма наклонов, получаемых в MLU-представлении адвективных потоков.

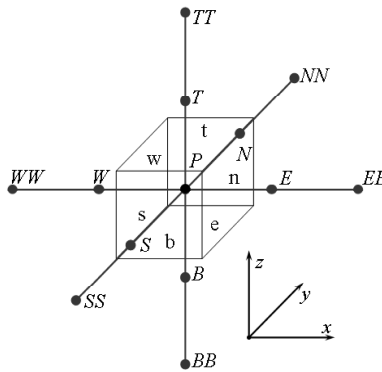


Рис. 1. Вычислительная ячейка с обозначением центров используемых по шаблону ячеек и граней выбранного конечного объема

Используемая явно- неявная разностная схема (2) позволяет неявным образом провести расчет в вертикальных столбцах, и тем самым избежать существенных ограничений на шаг по времени, вследствие малых вертикальных размеров вычислительных ячеек, высота которых из-за сгущения вычислительной сетки к поверхности в первом узле составляет всего 2 м. В тоже время горизонтальные размеры сетки (300-500м) позволяют использовать явную вычислительную схему. Такой подход обеспечивает высокую скорость вычислений вследствие применения экономичного метода прогонки для решения на каждом шаге по времени систем линейных уравнений с трехдиагональной матрицей (2).

Расчетная область представляет собой часть пограничного слоя атмосферы размером 50х50км на поверхности и 2 км в высоту, выбранную таким образом, что ее квадратное основание содержит участок подстилающей поверхности с крупным населенным пунктом в центре. В расчете учитывается эмиссия примеси от линейных, площадных и точечных источников, находящихся на поверхности или приподнятых над землей. Область покрывается вычислительной сеткой с размерами: $N_x \times N_y \times N_z = 100 \times 100 \times 30$ узлов. Период прогностического моделирования обычно составляет 48 часов. Использовался шаг по времени $\Delta t_c = 6$ сек для решения уравнений химической кинетики и $\Delta t_c = 60$ сек для интегрирования адвективно-диффузионных процессов. Поля метеорологических характеристик, необходимые для проведения расчета, подготавливались перед началом вычислений полей концентраций [4,12].

Таким образом, в данной постановке для исследования распространения и рассеяния основных загрязнителей воздуха над крупным индустриальным центром требуется решить порядка десяти пространственных нестационарных адвективно-диффузионных уравнений переноса с учетом химических реакций. Основным направлением использования данной модели является краткосрочный прогноз качества атмосферного воздуха над урбанизированной территорией, поэтому важно обеспечить наименьшее время вычислений. Однако современные рабочие станции и сервера даже с многоядерной архитектурой не способны обеспечить в полной мере требуемого быстродействия при использовании обыкновенной последовательной версии расчетной программы. Но широко распространенные в наши дни системы с параллельной архитектурой, в особенности кластеры [14], делают возможным эффективно сократить время вычислений за счет распараллеливания алгоритма.

5. Распараллеливание численного алгоритма

При параллельной реализации алгоритма очень важно выбрать оптимальный способ распределения нагрузки между процессорами, который определяется особенностями самого алгоритма, архитектурой вычислительной системы, количеством доступных для расчета процессоров, а также соображениями физического и вычислительного характера.

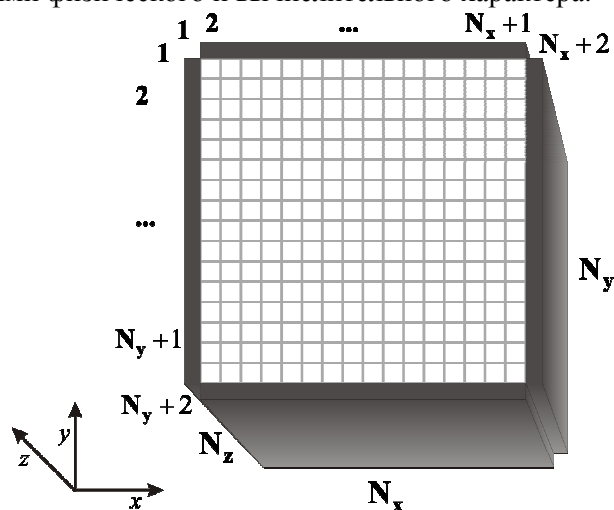


Рис. 2. Схема расчетной области

Рассмотрим особенности разработки параллельных программ, а также оценим ускорение и эффективность при одномерной и двумерной декомпозиции сеточной области. Схема расчетной области представлена на рис. 2.

Непосредственно область расчета $(2 \leq l \leq N_x + 1) \times (2 \leq j \leq N_y + 1)$ окаймлена по периметру фиктивными ячейками вследствие используемого расчетного шаблона и для обеспечения второго порядка аппроксимации граничных условий.

5.1. Одномерная декомпозиция

В случае одномерной декомпозиции расчетная область разделяется на блочные столбцы или строки (совокупности узлов расчетной сетки, попавших на один процессорный элемент) по одной из горизонтальных координат, выбор которой определяется особенностями хранения данных в оперативной памяти. В результате каждому вычислительному процессору предоставляется участок подстилающей поверхности и столб атмосферы над ним. На этапе инициализации процессы получают информацию о физических параметрах своей подобласти (метеорологические данные, параметры подстилающей поверхности, данные об источниках и т.д.) и подготавливаются к вычислениям.

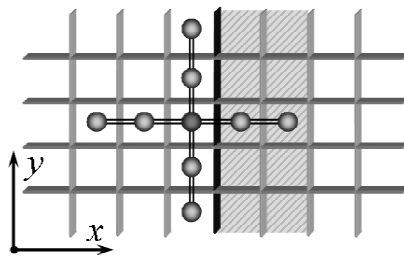


Рис. 3. Вычислительный шаблон разностной схемы вблизи границы раздела области на подобласти

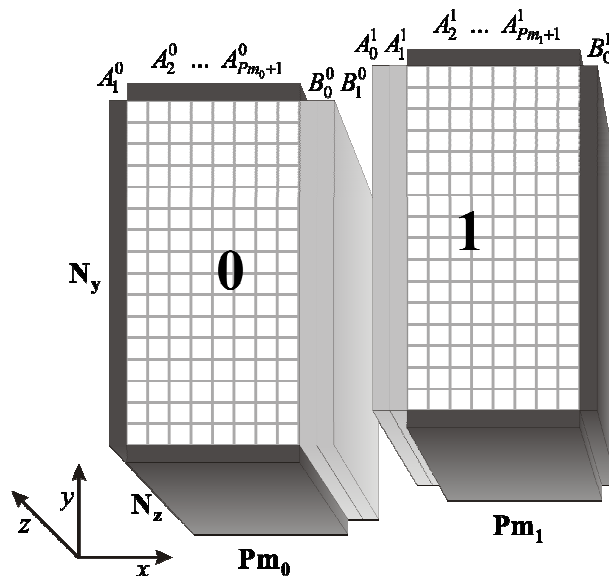


Рис. 4. Схема одномерной декомпозиции параллельной реализации явно-неявного метода на примере двух процессоров

При расчете приграничных значений в силу используемого разностного шаблона ("расширенный" крест – рис. 3.) процессы нуждаются в данных, которые находятся в локальной памяти соседних процессов, поэтому возникает необходимость обеспечения обменов данными из приграничных ячеек между соседними процессами в течение всего времени расчета. Рассмотрим пример декомпозиция сеточной области в случае использования двух процессоров (нулевого и

первого), представленный на рис. 4. Вдоль горизонтального направления Ox , по которому производится декомпозиция, выполняется перенумерация узлов сетки следующим образом. Нулевой вычислительный процесс ведет расчеты в подобласти шириной $Pm_0 = N_x/2$ с нумерацией узлов по оси Ox со второго до узла с номером $N_x/2 + 1$ (под делением подразумевается целочисленная операция).

Вычислительная область нулевого процесса ограничивается справа двумя фиктивными сечениями, в которые заносятся данные, рассчитываемые процессом с номером на единицу большим (первым процессом). Таким образом, область, с которой работает нулевой процессор, состоит из $Pm_0 + 3$ сечений с нумерацией от $A_0^0 = 1$ до $B_1^0 = Pm_0 + 3 = N_x/2 + 3$. Аналогичным образом, расчетная область первого процессора, состоит из $Pm_1 + 3$ сечений с нумерацией от $A_0^1 = N_x/2$ до $B_1^1 = Pm_1 + 2 = N_x + 2$ (рис. 4).

В общем случае, при декомпозиции на p процессоров ширина расчетной области для любого j -ого процесса, имеет вид:

$$Pm_j = \frac{N_x}{p}; \quad (4)$$

Здесь также подразумевается целочисленное деление. В случае, если N_x не делится нацело на число используемых процессоров p , то оставшиеся от деления полосы в количестве $N_x - N_x/p$ дораспределяются по процессам, т.е. часть процессоров имеет ширину расчетной области $Pm_j = N_x/p$, а оставшиеся – $Pm_j = N_x/p + 1$. При этом, естественно, выполняется условие:

$$N_x = \sum_{j=0}^{p-1} Pm_j. \quad (5)$$

Первоначально процедура обмена сеточными значениями искомой функции между соседними (по направлению декомпозиции расчетной области) процессами была реализована при помощи двух пар последовательных блокирующихся операций `MPI_SENDRECV`. Во время первой серии операций каждый процессор передает свои правые приграничные сечения, в данном случае $A_{Pm_0}^0, A_{Pm_0+1}^0$ процессу с номером на единицу больше и принимает левые граничные сечения A_0^0, A_1^0 от процесса с номером на единицу меньше (рис. 4). Вторая часть передачи данных выполняется в обратном направлении: процессоры передают левые приграничные сечение A_2^1, A_3^1 своим предыдущим по номерам процессам и принимают правые граничные сечения B_0^0, B_1^0 от последующих по номерам процессоров [6]. Обмены осуществляются на каждом временном шаге после расчета новых значений концентраций в подобласти. При этом индекс пересылаемого массива концентраций, соответствующий координате, по которой производится разбиение области, следует делать последним, поскольку это обеспечивает хранение пересылаемых элементов в области памяти последовательно и помогает получить более простой и быстрый доступ к ним [6].

Такой подход легко реализуется и гарантированно обеспечивает синхронизацию между одновременно выполняемыми процессами при расчете [2]. В тоже время он не является оптимальным, поскольку каждый процесс вынужден бездействовать до момента завершения обменов, т.е. пока все процессорные элементы не получат отправленные им сообщения.

Устранить указанный выше недостаток можно используя неблокирующиеся или асинхронные операции обмена `MPI_ISEND` и `MPI_Irecv`. В этом случае операции получения и отправки сообщений идут во время расчета и не требуют ожидания их завершения.

5.2. Двумерная декомпозиция

Двумерная декомпозиция предполагает лучшую масштабируемость задачи, решаемой конечно-разностными или конечноэлементными методами на многопроцессорных вычислитель-

ных системах, поскольку такой подход позволяет задействовать большее число процессоров. Так, при некотором ограничении на ширину обрабатываемой сеточной области при одномерном разбиении можно использовать p процессоров, а при двумерном p^2 . Теоретически при увеличении числа процессоров и заданной размерности задачи время на пересылку будет сокращаться. Тогда, начиная с некоторой размерности и некоторого числа процессоров, для данной задачи более эффективным является двумерное разбиение области [6].

Двумерное разбиение можно рассматривать как два одномерных разбиения, выполненных независимо вдоль координатных направлений Ox и Oy на p_x и p_y ($p_x \times p_y = p$) процессоров соответственно рис. 5.

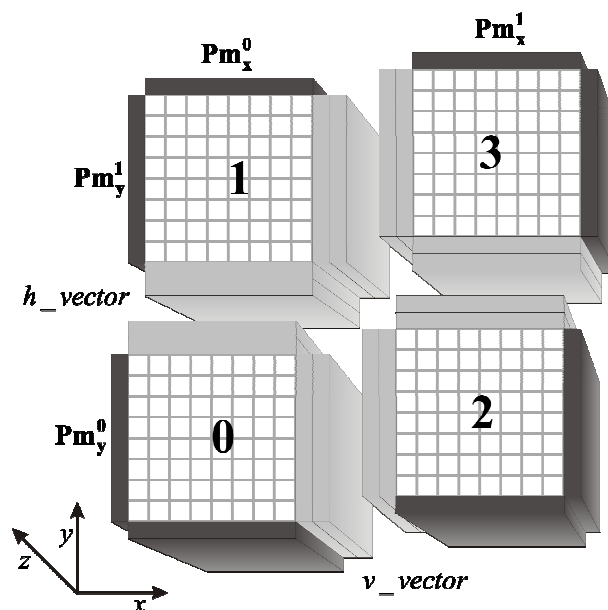


Рис. 5. Схема двумерной декомпозиции параллельной реализации явно- неявного метода на примере четырех процессоров ($p = 4$, $p_x = 2$, $p_y = 2$)

Реализация двумерной декомпозиции с использованием стандарта MPI [2] существенно отличается от одномерной. Во-первых, необходимо организовать двумерную топологию одновременно выполняемых процессов. В данной работе выбрана декартова топология. После инициализации MPI и определения числа задействованных процессоров вызывается процедура MPI_Dims_Create, подбирающая оптимальные параметры $p_x \times p_y = p$ для двумерной декомпозиции, используя которые процедура MPI_Cart_Create создает декартову топологию с перенумерацией процессов. Далее MPI_Comm_Rank и MPI_Cart_Shift позволяют определить новый номер процесса и новые номера соседних процессоров уже в декартовой топологии. После этого вычисляются размеры подобластей для каждого процесса. Такой подход универсален и позволяет организовать вычисления на любом четном числе процессоров, но для сбалансированного распределения нагрузки рекомендуется выбирать p , удовлетворяющее условию:

$$p_x = p_y = \sqrt{p}.$$

Во-вторых, в двумерной декомпозиции более сложной является реализация межпроцессорных обменов. Если в одномерном случае всегда можно выбрать направление декомпозиции таким образом, что элементы трехмерного массива концентрации вдоль границы будут располагаться в памяти последовательно друг за другом что, несомненно, будет удобным для формирования пересылаемого сообщения, то при двумерной декомпозиции добиться этого для всех направлений, по которым проводится декомпозиция, не представляется возможным. Поэтому для организации обменов при помощи процедуры MPI_Type_Hvector создаются новые типы

данных – трехмерные массивы, представляющие собой сечения подобласти, выполненные плоскостями $X = const$ и $Y = const$ (h_vector и v_vector (рис.5)), с размерами $(N_z, 1, N_y/p_y)$ и $(N_z, N_x/p_x, 1)$. Пересылка данных между процессорами осуществляется такими блоками при помощи блокирующих операций обмена MPI_SENDRECV.

В-третьих, при двумерной декомпозиции непросто осуществить сбор решения на одном процессорном элементе. Для сбора решения на одном процессорном элементе, например, перед выдачей в файл процессы формируют одномерные массивы данных и пересылают процессорному элементу с номером "0", который собирает полученные данные и вновь их переписывает в соответствующем порядке в трехмерный массив.

6. Ускорение параллельной программы

В работе проведено экспериментальное исследование ускорения параллельной программы по отношению к последовательной и ее эффективность. С этой целью была проведена серия расчетов на кластере ТГУ СКИФ Cyberia при следующих условиях: период моделирования 1 час, вычислительная сетка $100 \times 100 \times 30$ узлов, кинетическая схема химических и фотохимических реакций – GRS [9]. Для обеспечения сбалансированной нагрузки и достижения тем самым наилучшей производительности расчеты проводились с использованием такого числа процессоров, чтобы количество узлов сетки расчетной области, обрабатываемых каждым процессорным элементом, было одинаковым. Для одномерной декомпозиции использовались 1, 2, 4, 10, 20, 25 и 50 процессоров вычислительного кластера, для двумерной – 1, 2, 4, 16, 25, 64. Графики достигнутого ускорения представлены на рис. 6.

Численное решение используемых в модели уравнений переноса и химических превращений представляет собой достаточно трудоемкую в вычислительном плане задачу, поэтому доля временных затрат на пересылки в общем времени работы программы незначительна. В результате при использовании малого числа процессоров (2 или 4) наблюдается практически линейное равное для обеих конфигураций ускорение. Действительно, при использовании 2-х процессоров двумерная декомпозиция сводится к одномерной и ни чем от неё не отличается.

При задействовании десяти и более процессоров ситуация становится более определенной. При использовании блокирующихся операций при увеличении числа межпроцессорных обменов увеличивается время простоя процессоров в ожидании завершения передачи сообщения. Эффективность одномерной конфигурации снижается, что особенно отчетливо видно на фоне более высокой производительности алгоритма с двумерной декомпозицией.

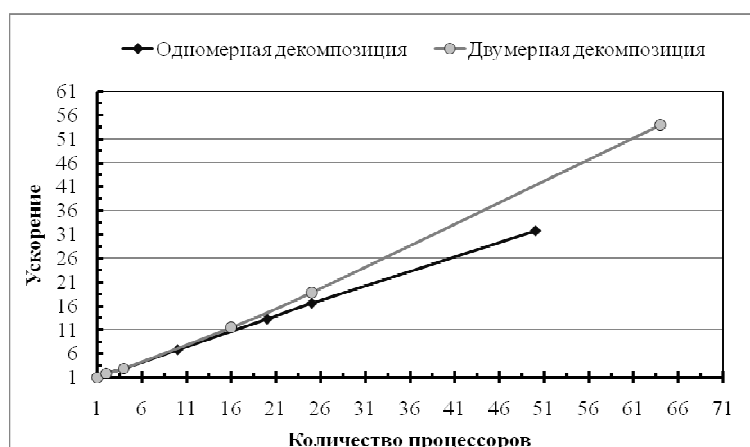


Рис. 6. Ускорение параллельной программы для различных вариантов декомпозиции

При двумерной декомпозиции, увеличение числа процессов для решения задачи приводит к уменьшению размера пересылаемого сообщения при неизменном числе обменов на каждый процессор (общее количество обменов растет вместе с числом задействованных процессоров),

поэтому производительность продолжает расти. При использовании 64 процессоров ускорение составляет 54, а эффективность соответственно 0,84.

Таким образом, для обеспечения высокой производительности параллельной программы с двумерной декомпозицией на кластере ТГУ СКИФ Cyberia принципиальное значение имеет размер пересылаемого сообщения (количество пересылаемых элементов), а не общее количество обменов.

Вычислительная система ТГУ СКИФ Cyberia собрана из двухпроцессорных узлов и сконфигурирована так, что при запуске задачи в стандартном режиме процессы распределяются по принципу минимизации количества работающих узлов, т.е. задействуются оба процессора на узле. Такой подход способствует экономии ресурсов, но не оптимален с точки зрения быстродействия, поскольку в этом случае оба процессора на одном узле используют одну сетевую карту и один канал связи, а также делят между собой оперативную память. Это, прежде всего, затрудняет коммуникационные обмены между процессами, что в конечном итоге сказывается на производительности вычислений.

При помощи специальных директив ОС Linux возможно использование в параллельных вычислениях только одного процессора на узле. Проведенная серия вычислений с использованием только одного и двух процессоров на узле показывает, что эффективность рассматриваемых способов декомпозиции сеточной области в этом случае возрастает (см. пунктирные кривые на рис. 7). Особенно заметен прогресс для одномерной декомпозиции, эффективность которой увеличивается на 30-45%. Причина кроется в следующем. При синхронных обменах оба процессора на узле вынуждены одновременно использовать один канал связи, следовательно, скорость обменов уменьшается примерно в 2 раза. В случае запуска программы только на одном процессоре на узле этой проблемы нет, так как нет конкурента, претендующего на сетевые ресурсы. Двумерная конфигурация также показывает более высокие результаты, эффективность достигает 0.95. Но наибольший прирост производительности (25-35%) наблюдается при использовании 4-25 процессоров, когда размеры пересылаемых блоков значительны и двухпроцессорная конфигурация вычислительного узла заметно замедляет передачу данных другим вычислительным узлам.

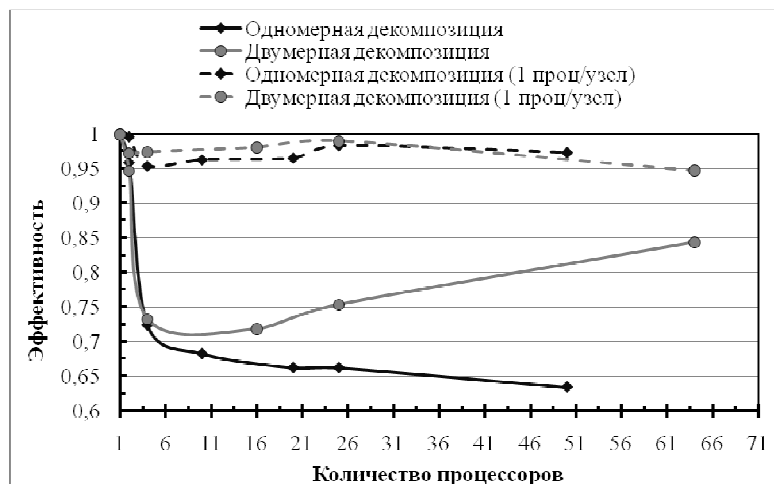


Рис. 7. Эффективность параллельной программы для различных вариантов декомпозиции

Таким образом, использование двухпроцессорных вычислительных узлов с одной сетевой картой на узле сужает канал передачи сообщений и уменьшает скорость обменов при использовании блокирующихся операций, особенно, при одномерной декомпозиции. Возможно, для конструирования высокоэффективной параллельной программы следует использовать асинхронные обмены, которые более рационально используют коммуникационную среду. При обеспечении высокой скорости передачи сообщений эффективность разных подходов при организации межпроцессорных обменов примерно одинакова. Но и в этом случае предпочтительнее использовать более масштабируемую двумерную декомпозицию, позволяющую получить

решение за минимальное время. Так один 1 час моделирования на 64 процессорах выполняется за 21 секунду, а на одном - за 19 минут.

Таким образом, в данной работе проведено исследование эффективности одно- или двумерной декомпозиции расчетной области для распараллеливания алгоритма решения системы адвективно-диффузионно-кинетических уравнений в математической модели переноса примеси. Двумерная декомпозиция производится по двум горизонтальным координатам, одномерная - по одной. Межпроцессорная передача данных основана на синхронных операциях обмена. Выявлено, что производительность параллельной программы с двумерной декомпозицией выше чем с одномерной. А ее эффективность начинает расти при использовании 16-ти и более процессоров. При одномерной декомпозиции эффективность с увеличением доступных для вычислений процессоров наоборот падает.

Применение двухпроцессорных вычислительных узлов с одним каналом передачи сообщений уменьшает скорость обменов при использовании блокирующих операций при рассмотренных способах декомпозиции. Применение специальных директив для задействования при вычислениях только одного процессора на узле позволяет увеличить производительность параллельной программы. В данном случае прирост эффективности программы с одномерной декомпозицией составляет 30-45%, с двумерной - 20-30%.

Данная модель и разработанный алгоритм решения будут использованы в создаваемой в Томском государственном университете информационно-вычислительной системе для контроля и прогноза качества воздуха над городом Томском. Высокая эффективность численного алгоритма открывает широкие возможности для модификации модели: укрупнения расчетной области и увеличения разрешения сетки, включения более сложных механизмов химических реакций и других усовершенствований.

Литература

1. Dabdub D., Seinfeld J.H. Parallel Computation in Atmospheric Chemical Modeling // *Parallel Computing*. –1996. –V.22. –P.111-130.
2. Gropp W., Lusk E., Skjellum A. Using MPI-2: Portable Parallel Programming with the Message-Passing Interface –MIT Press, 1999. –382 p.
3. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. –СПб.: БХВ-Петербург, 2002. – 608 с.
4. Есаулов А.О. Использование параллельных вычислений при моделировании мезомасштабных атмосферных процессов // *Труды Второй Сибирской школы-семинара по параллельным вычислениям*. –Томск: Изд-во Том. ун-та, 2004. –С.100-107.
5. Беликов Д.А., Старченко А.В. Исследование образования вторичных загрязнителей (озона) в атмосфере г. Томска // *Оптика атмосферы и океана*. –2005. –Т.18, №05-06. –С.435-443.
6. Толстых М.А. Особенности использования MPI для распараллеливания задач математической физики на вычислительных системах с невысокой скоростью обменов между процессорами // *Сибирская школа-семинар по параллельным вычислениям*. –Томск: Изд-во Том. ун-та, 2006. –С.99-125.
7. Беликов Д.А., Старченко А.В. Численная модель турбулентного переноса примеси в пограничном слое атмосферы // *Оптика атмосферы и океана*. –2007, –Т.20, №8. –С.667-673.
8. Morison R.P., Leslie L.M., Speer M.S. Atmospheric modeling of air pollution as a tool for environmental prediction management // *Meteorology and atmospheric physics*. –2002. –V.80. – P.141-151.
9. Hurley P. J. The Air Pollution Model (TAPM) Version 2 // *CSIRO Atmospheric Research Technical Paper*. –2002. –N.55. –P.37.

10. Короленок Е. В., Нагорнов О. В. Моделирование приземных концентраций озона городского региона // Математическое моделирование. –2002. –Т.14, №4. –С.80-94.
11. Perego S. Metphomod – a Numerical Mesoscale Model for Simulation of Regional Photosmog in Complex Terrain: Model Description and Application during Pollumet 1993 (Switzerland) // Meteorology and Atmospheric Physic. –1999. –V.70. –P.43-69.
12. Старченко А.В., Беликов Д.А. Численная модель для оперативного контроля уровня загрязнения городского воздуха // Оптика атмосферы и океана. –2003. –Т.16, №7. –С.657-665.
13. Van Leer B. Towards the ultimate conservative difference scheme. Part IV: A new approach to numerical convection // J.Comput. Phys. –1977. –V.23. –P.276-299.
14. Электронный ресурс. Режим доступа (<http://parallel.ru/>).