

Применение параллельных алгоритмов при численном моделировании кровотока в квазиодномерном приближении

А.Н. Авдеева <aan-bmstu@yandex.ru>

В.В. Пузикова <vvp@dms-at.ru>

Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана),
105005, Россия, г. Москва, ул. 2-я Бауманская, дом 5, стр. 1

Аннотация. Главной целью современного гемодинамического моделирования является предсказание поведения давления крови в артериях, а также изучение комплексного воздействия разнообразных факторов на характеристики сердечно-сосудистой системы. Наиболее популярными при этом являются квазиодномерные модели течения крови по сосудам, позволяющие моделировать кровоток во всей сосудистой системе. Поскольку полномасштабное моделирование сердечно-сосудистой системы требует больших вычислительных затрат, актуальной является задача распараллеливания вычислений. В данной работе проведено исследование эффективности распараллеливания вычислений при численном моделировании кровотока в квазиодномерном приближении. Для простоты рассмотрена задача о моделировании течения крови в отдельном кровеносном сосуде. При построении параллельного алгоритма был применен метод декомпозиции области. В каждой подобласти задача на каждом шаге по времени расщепляется на гиперболическую и параболическую подзадачи. Для решения гиперболической подзадачи используется интегро-интерполяционный метод, основанный на схеме MUSCL. Для интегрирования по времени применяются методы Рунге-Кутты и Адамса-Башфорта второго порядка. Для решения параболической подзадачи используется метод Кранка-Николсона. На стыках подобластей интерфейсные условия образуют нелинейные системы с тремя неизвестными. Эти системы решаются при помощи метода Ньютона. С помощью профилировщика AMD CodeAnalyst была определена структура временных затрат при решении тестовой задачи в последовательном режиме. При помощи закона Амдала получены оценки максимально возможного ускорения при распараллеливании наиболее дорогостоящих с вычислительной точки зрения операций. При реализации полученного алгоритма в разработанном авторами настоящей работы программном комплексе использовались технология OpenMP и библиотека MPI. Расчеты проводились на учебно-вычислительном кластере кафедры ФН-2 «Прикладная математика» МГТУ им. Н.Э. Баумана. Результаты вычислительных экспериментов показали, что выигрыш по времени, достигаемый за счет использования библиотеки MPI, не превышает нескольких процентов по сравнению с применением технологии

OpenMP. В связи с этим, принимая во внимания простоту распараллеливания алгоритмов посредством OpenMP, можно остановить выбор на данной технологии, однако использование MPI позволяет сделать программный комплекс универсальным – работающим как на системах с общей памятью, так и на системах с распределенной памятью.

Ключевые слова: технология OpenMP; MPI; квазиодномерная модель; кровоток; метод декомпозиции области; параллельный алгоритм; кластер; метод MUSCL; расщепление Годунова.

DOI: 10.15514/ISPRAS-2018-30(2)-15

Для цитирования: Авдеева А.Н., Пузикова В.В. Применение параллельных алгоритмов при численном моделировании кровотока в квазиодномерном приближении. Труды ИСП РАН, том 30, вып. 2, 2018 г., стр. 301-316. DOI: 10.15514/ISPRAS-2018-30(2)-15

1. Введение

В последние годы главной причиной смертности в мире являются сердечно-сосудистые заболевания. В настоящее время значительно увеличивается количество стентирований кровеносных сосудов и ангиопластик, причем в проекте государственной программы по развитию здравоохранения планируется дальнейшее усиление мер по оказанию специализированной медицинской помощи больным с острой сосудистой патологией. Нередко для восстановления кровообращения в пораженных сосудах помимо медикаментозного лечения проводятся дорогостоящие реконструктивные операции, и зачастую невозможно объективно оценить, какой тип оперативного вмешательства будет оптимальным для конкретного пациента, а также, насколько близок будет кровоток в сосуде к нормальному после операции. Таким образом, очень важной задачей представляется не только усовершенствование методов ранней диагностики сердечно-сосудистых заболеваний, но и разработка методов, позволяющих смоделировать для конкретного пациента последствия проведения той или иной реконструктивной кардиохирургической операции.

Эффективным методом решения данной проблемы является использование вычислительного эксперимента: в виртуальной системе кровообращения можно реализовать различные сценарии операции для конкретного пациента, спрогнозировать исход той или иной тактики лечения и выбрать оптимальную [1, 2]. Использование относительно недорогих средств математического моделирования может обеспечить значительный прогресс в разработке и оптимизации формы искусственных клапанов сердца, шунтов, имплантируемых насосов крови, способов управления ими, и, самое главное, позволит исследовать их влияние на гемодинамику и спрогнозировать успешность применения данного метода терапии для конкретного пациента в дооперационный период. Прогностическую ценность и клинический потенциал персонализированного моделирования сердечно-сосудистой системы для ряда

клинически значимых параметров доказал международный исследовательский проект «euHeart» [3], который возглавляла компания Royal Philips Electronics.

Главной целью современного гемодинамического моделирования является предсказание поведения давления крови в артериях, а также изучение комплексного воздействия разнообразных факторов на характеристики сердечно-сосудистой системы. Наиболее популярными при этом являются квазиодномерные модели течения крови по сосудам [4–8], позволяющие моделировать кровоток во всей сосудистой системе. Параметры отдельных сосудов (длины сосудов, скорости распространения пульсовых волн и т.д.) берут из результатов медицинских исследований, например, цветового дуплексного сканирования артерий.

Поскольку полномасштабное моделирование сердечно-сосудистой системы требует больших вычислительных затрат, актуальной является задача распараллеливания вычислений. Вопросам разработки параллельных алгоритмов решения разнообразных задач вычислительной механики и анализу их эффективности посвящено большое число исследований [9-15]. При этом рассматриваются вопросы проведения расчетов как на вычислительных системах кластерного типа при помощи библиотеки параллельных вычислений MPI [16, 17], так и на системах с общей памятью при помощи таких технологий параллельного программирования, как Intel® Cilk™ Plus [18], Intel® TBB [19], OpenMP [20].

Целью данной работы является изучение возможности эффективного распараллеливания вычислений при численном моделировании кровотока в квазиодномерном приближении. Для простоты рассматривается задача о моделировании течения крови в отдельном кровеносном сосуде. При построении параллельного алгоритма применяется метод декомпозиции области [21]. При реализации полученного алгоритма в разработанном авторами настоящей работы программном комплексе используются технология OpenMP и библиотека MPI.

2. Квазиодномерная модель кровотока в отдельном сосуде

Для простоты рассмотрим один участок эластичного кровеносного сосуда (рис. 1) в цилиндрической системе координат (r, θ, x) . Внутренний радиус сосуда R зависит от координаты x и времени t . Кровь полагается однородной и несжимаемой (плотность $\rho = const$) ньютоновской жидкостью (вязкость $\nu = const$). Течение характеризуется скоростью $\vec{v} = (v_r, v_\theta, v_x)$ и давлением P . Отметим, что P – это внутреннее давление, т.е.

$$P = P_{ext} + P_{art}.$$

Здесь P_{ext} – внешнее давление, т.е. атмосферное давление, а P_{art} – трансмуральное давление, которое в медицине принято называть артериальным давлением [1, 2]. Предполагаем, что P_{ext} постоянно вдоль осевой переменной

x . Считается, что силой тяжести можно пренебречь [7, 8]. Сосуд и течение крови полагаются осесимметричными, поэтому $v_\theta = 0$ и ни одна из переменных не зависит от θ . Полагаем, что течение описывается уравнением неразрывности и уравнением Навье-Стокса [7, 8]. Эти уравнения дополняются граничным условием прилипания

$$v_x(R) = 0.$$

Кроме того, предполагаем, что стенка сосуда движется только вдоль r , т.е.

$$v_r(R, x, t) = \frac{\partial R}{\partial t}.$$

Из этого уравнения следует, что давление постоянно вдоль радиуса.

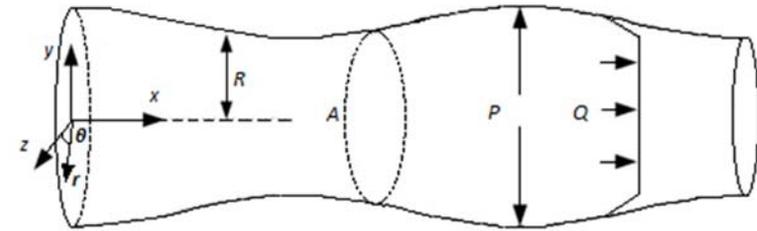


Рис. 1. Расчетная область
Fig. 1. Computational domain

Перепишем уравнение неразрывности в интегральном виде:

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0. \quad (1)$$

Здесь Q – расход потока, A – площадь поперечного сечения:

$$Q = \int_0^R 2\pi v_x r dr, \quad A = \pi R^2.$$

Для описания вязкоупругого поведения стенки сосуда используем модель Кельвина – Фойгта. Тогда внутреннее давление можно вычислять следующим образом [7]:

$$P = P_{ext} + \beta(\sqrt{A} - \sqrt{A_0}) + \nu_s \frac{\partial A}{\partial t}. \quad (2)$$

Здесь β – коэффициентом жесткости и ν_s – коэффициент вязкости.

В работе [7] показано, что при малых возмущениях можно полагать, что

$$\frac{A}{\rho} \frac{\partial}{\partial x} \left(\nu_s \frac{\partial Q}{\partial x} \right) \approx C_\nu \frac{\partial^2 Q}{\partial x^2}.$$

Здесь $C_v = \frac{Av_s}{\rho}$. Тогда уравнение Навье–Стокса в интегральном виде с учетом

(2) можно переписать следующим образом:

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q^2}{A} + \frac{\beta}{3\rho} A^{3/2} \right) - C_v \frac{\partial^2 Q}{\partial x^2} = -C_f \frac{Q}{A} + \frac{A}{\rho} \left(\frac{\partial(\beta\sqrt{A_0})}{\partial x} - \frac{2}{3}\sqrt{A} \frac{\partial\beta}{\partial x} \right). \quad (3)$$

Здесь $C_f = 22\pi\nu$ [7]. Уравнения (1), (3) можно записать следующим образом:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = S. \quad (4)$$

Здесь U – консервативная переменная, F – соответствующий поток, S – источник:

$$U = \begin{pmatrix} A \\ Q \end{pmatrix}, \quad F = F_c + F_v = \begin{pmatrix} Q^2 \\ \frac{\beta}{3\rho} A^{3/2} \end{pmatrix} + \begin{pmatrix} 0 \\ -C_v \frac{\partial Q}{\partial x} \end{pmatrix},$$

$$S = \begin{pmatrix} 0 \\ -C_f \frac{Q}{A} + \frac{A}{\rho} \left(\frac{\partial(\beta\sqrt{A_0})}{\partial x} - \frac{2}{3}\sqrt{A} \frac{\partial\beta}{\partial x} \right) \end{pmatrix}.$$

Заметим, что поток состоит из двух частей, конвективной F_c и диффузионной F_v . В общем случае, при работе с конвективными и диффузионными потоками применяются разные численные методы. Поэтому диффузионные слагаемые часто отделяют и переносят в правую часть. Таким образом, мы можем переписать задачу (4) следующим образом:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = S + D, \quad F = F_c, \quad D = \begin{pmatrix} 0 \\ C_v \frac{\partial^2 Q}{\partial x^2} \end{pmatrix}. \quad (5)$$

Запишем в общем виде, как будут выглядеть начальные и граничные условия для каждого сосуда. В качестве начального условия полагаем

$$U(x,0) = U_{in}(x).$$

На левом конце рассматриваемого сосуда зададим граничное условие первого рода, а на правом конце – условие свободного выхода потока:

$$U(0,t) = U_0(t), \quad \frac{\partial U(L,t)}{\partial \bar{n}} = 0.$$

3. Численный метод

Как и в работах [7, 8] используем метод дробных шагов: исходную задачу (5) разделим на гиперболическую и параболическую соответственно:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = S, \quad (6)$$

$$\frac{\partial U}{\partial t} = D. \quad (7)$$

Тогда, решение исходной задачи (5) может быть заменено последовательным решением этих двух подзадач. В данной работе используется метод расщепления Годунова: на каждом шаге по времени будем решать гиперболическую подзадачу, чтобы получить прогноз U^* , который используется в качестве начального значения в параболической подзадаче

$$U^j \xrightarrow{\Delta t, H} U^* \xrightarrow{\Delta t, P} U^{j+1}.$$

Здесь H и P означает решение гиперболической и параболической подзадач на j -м шаге по времени. Второй шаг можно рассматривать как корректор.

Для решения гиперболической подзадачи (6) используем метод контрольных объемов: каждую подобласть разделим на ячейки I_i . Для каждой такой ячейки должен выполняться закон сохранения

$$\int_{I_i} \frac{\partial U}{\partial t} dx + \int_{I_i} \frac{\partial F}{\partial x} dx = \int_{I_i} S dx.$$

При построении дискретных аналогов производных по пространству используем MUSCL-ограничитель [22]. Для решения дифференциальное уравнение первого порядка, получающегося после дискретизации по пространству в разработанном программном комплексе реализованы два метода интегрирования по времени второго порядка: метод Рунге–Кутты и двухшаговый метод Адамса–Башфорта.

Заметим, что для A решать параболическую подзадачу (7) не требуется, так как $A^{j+1} = A^*$. Поэтому (7) принимает вид:

$$\frac{\partial Q}{\partial t} = C_v \frac{\partial^2 Q}{\partial x^2}.$$

Для решения данной задачи используем метод Кранка–Николсона:

$$\frac{Q_i^{j+1} - Q_i^j}{\Delta t} = C_v \left(\frac{Q_{i+1}^{j+1} - 2Q_i^{j+1} + Q_{i-1}^{j+1}}{\Delta x^2} + \frac{Q_{i+1}^j - 2Q_i^j + Q_{i-1}^j}{\Delta x^2} \right). \quad (8)$$

Здесь $Q^j = Q^*$, т.е. Q^j является решением гиперболической подзадачи. Разностная задача (8) решается методом прогонки.

4. Декомпозиция области

Для распараллеливания вычислений при решении задач вычислительной механики часто используют методы декомпозиции области [10]. Идея методов декомпозиции заключается в том, что расчетная область разбивается на пересекающиеся или непересекающиеся подобласти и исходная задача представляется в виде совокупности вспомогательных краевых задач в этих подобластях. При этом на границах подобластей, совпадающих с границами исходной расчетной области, ставятся граничные условия из исходной задачи, а на остальных границах подобластей, называемых внутренними, ставятся условия сопряжения (интерфейсные условия). Решение вспомогательных задач может осуществляться параллельно.

На рис. 2 показан пример декомпозиции расчетной области на две подобласти: в месте стыковки подобластей имеем четыре неизвестные: A_p , Q_p на выходе из родительской артерии и A_d , Q_d на входе в дочернюю. Здесь и далее нижний индекс p , d определяет принадлежность к родительской и дочерней артериям соответственно.

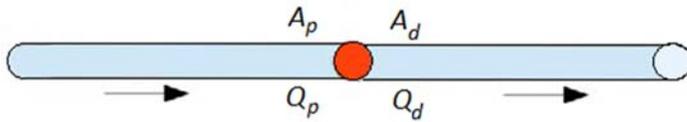


Рис. 2. Декомпозиция области
Fig. 2. Domain decomposition

В точке соединения на каждом $n+1$ -м шаге по времени должны выполняться уравнение расхода потока жидкости и закон сохранения импульса:

$$Q_p^{n+1} = Q_d^{n+1}, \quad (9)$$

$$\frac{1}{2} \rho_p \left(\frac{Q_p^{n+1}}{A_p^{n+1}} \right)^2 + P_p^{n+1} = \frac{1}{2} \rho_d \left(\frac{Q_d^{n+1}}{A_d^{n+1}} \right)^2 + P_d^{n+1}. \quad (10)$$

Значения давления P_p^{n+1} и P_d^{n+1} зависят от площади поперечного сечения и вычисляются при помощи (2).

В работе [7] показано, что в физиологических условиях число Уомерсли W_0 достаточно мало, из-за чего источникные члены в рассматриваемой системе уравнений оказываются незначительными и в системе преобладает гиперболичность. Поэтому два недостающих условия на стыке подобластей могут получены при помощи характеристик или инвариантов Римана[22]

$$W_{1,2} = \frac{Q}{A} \pm 4c. \quad (11)$$

Здесь c – скорость Моэнса – Кортевега, которая является скоростью волны давления, т.е. скоростью пульсовой волны:

$$c = \sqrt{\frac{\beta}{2\rho} A^2}.$$

Два собственных значения λ_1 и λ_2 имеют противоположные знаки:

$$\lambda_{1,2} = \frac{Q}{A} \pm c.$$

Соответственно, на каждом конце вычислительной области существует ровно одна входящая характеристика и одна выходящая. Поскольку $W_{1,2}$ постоянны вдоль линий $D_t X_{1,2} = \lambda_{1,2}$ в пространственно-временной плоскости, можно получить значения $W_1^{n+1}(L_i)$ и $W_2^{n+1}(0)$, применив интерполяцию данных с n -го шага по времени:

$$W_1^{n+1}(L_i) = W_1^n(L - \lambda_1^n(L_i)\Delta t), \quad W_2^{n+1}(0) = W_2^n(-\lambda_2^n(0)\Delta t), \quad t > 0. \quad (12)$$

Таким образом, в родительской артерии значение $(W_1)_p^{n+1}$ можно вычислить по формуле (12) и потребовать, чтобы оно было равно значению $W_1(U_p^{n+1})$, которое задается соотношением (11). В итоге имеем уравнение

$$(W_1)_p^{n+1} = W_1(U_p^{n+1}). \quad (13)$$

Аналогично строится уравнение для W_2 для дочерней артерии:

$$(W_2)_d^{n+1} = W_2(U_d^{n+1}). \quad (14)$$

Таким образом, имеем систему из четырех уравнений (9), (10), (13) и (14) с четырьмя неизвестными. Ее решение даст значения неизвестной U на стыке подобластей. С учетом условия (9) уравнения (10), (13), (14) примут вид:

$$\begin{cases} \frac{1}{2} \rho \left(\frac{Q_p^{n+1}}{A_p^{n+1}} \right)^2 + P_p^{n+1} - \frac{1}{2} \rho \left(\frac{Q_p^{n+1}}{A_d^{n+1}} \right)^2 + P_d^{n+1} = 0, \\ (W_1)_p^{n+1} - W_1(A_p^{n+1}, Q_p^{n+1}) = 0, \\ (W_2)_d^{n+1} - W_2(A_d^{n+1}, Q_p^{n+1}) = 0. \end{cases}$$

Таким образом, имеем нелинейную алгебраическую систему вида

$$F(X) = 0, \quad X = (A_p^{n+1}, Q_p^{n+1}, A_d^{n+1})^T.$$

Для ее решения применим итерационный метод Ньютона. Матрица Якоби считается аналитически. В качестве начального приближения будем использовать значения неизвестных с предыдущего шага по времени

$$X^{(0)} = (A_p^n, Q_p^n, A_d^n)^T.$$

Расчетная формула для расчета $k+1$ -го итерационного приближения метода Ньютона имеет следующий вид

$$X^{(k+1)} = X^{(k)} - (F'(X^{(k)}))^{-1} F(X^{(k)}), \quad k = 0, 1, \dots$$

Отметим, что во всех вычислительных экспериментах было достаточно нескольких итераций для достижения точности 10^{-6} .

5. Оценка эффективности распараллеливания вычислений

Определим, какие вычисления имеет смысл распараллелить. Для этого необходимо выделить участки программы, на выполнение которых расходуется наибольшее количество времени. Полная структура временных затрат при решении тестовой задачи была определена с помощью профилировщика AMD CodeAnalyst [23]: 95,8 % времени работы программы занимает расчет значений A и Q внутри каждой подобласти (операция 1), а 1,2 % – вычисление неизвестных из интерфейсных условий на стыках подобластей (операция 2). Все прочие операции занимают 3 % времени выполнения расчета (рис. 3).



Рис. 3. Структура временных затрат при решении тестовой задачи
Fig. 3. The time-cost structure

Оценим максимальное ускорение, которое можно получить при распараллеливании только операции 1 или операций 1 и 2, при помощи закона Амдала [24], который гласит, что для системы из S вычислительных ядер максимальное ускорение программы с долей P параллельного кода и $(1-P)$ последовательного кода равно

$$\alpha = \frac{1}{\frac{P}{S} + (1-P)}.$$

Полученные оценки приведены в табл. 1. Они соответствуют случаю идеального распараллеливания, при котором параллельный код на

вычислительной системе с S ядрами выполняется в S раз быстрее. Реальное ускорение будет ниже, поскольку при переходе от последовательной программы к параллельной добавятся накладные расходы на поддержку многопоточных вычислений. Далее для распараллеливания вычислений будем использовать технологию параллельного программирования OpenMP и библиотеку MPI. Технология OpenMP предполагает, что пользователь при помощи ключевых слов лишь обозначает задачи, выполняемые параллельно, а организация управления потоками и работы с ними определяются самой технологией. Таким образом, вышеперечисленные накладные расходы на поддержку многопоточности, а, значит, и реальное ускорение, зависят от используемой технологии параллельного программирования.

Табл. 1. Максимально возможное ускорение при распараллеливании операций 1 и 2

Table 1. The maximum possible speed up at operations 1 and 2 in parallel mode

Распараллеливаемые операции	P	Максимальное ускорение, раз			
		2 ядра	4 ядра	8 ядер	12 ядер
1	0,958	1,92	3,55	6,18	8,21
1, 2	0,970	1,94	3,67	6,61	9,02

Вычислительные эксперименты проводились на учебно-экспериментальном вычислительном кластере кафедры ФН-2 "Прикладная математика" МГТУ им. Н.Э. Баумана [25]. Расчетная область разбивалась на 100 подобластей. Каждая подобласть состояла из 200 контрольных объемов. Моделировалось 1000 шагов по времени. Распараллеливались операции 1 и 2.

На рис. 4 представлена зависимость ускорения от числа вычислительных ядер при распараллеливании при помощи технологии OpenMP.



Рис. 4. Реальное ускорение при использовании технологии OpenMP и оценки ускорения
Fig. 4. The real acceleration (OpenMP technology) and acceleration estimates

Видно, что реальное ускорение превышает линейное. Это можно объяснить тем, что при решении задачи на одном процессоре данные постоянно подгружаются в кэш-память, а при распараллеливании задачи происходит деление данных по процессорам, и кэш каждого модуля смог вместить свою порцию данных.

На рис. 5 представлена зависимость ускорения от числа вычислительных ядер при распараллеливании, использующем библиотеку MPI.

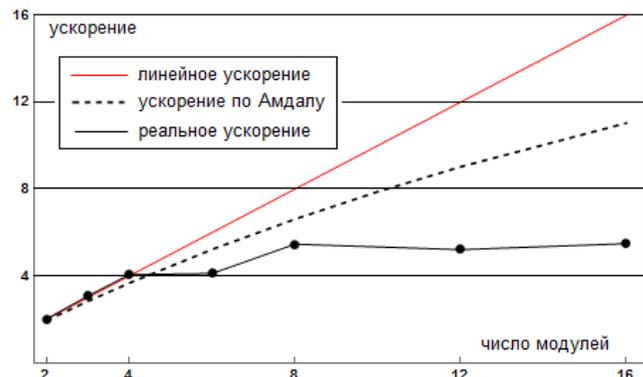


Рис. 5. Реальное ускорение при использовании библиотеки MPI и оценки ускорения
Fig. 5. The real acceleration (MPI library) and acceleration estimates

На рассмотренной тестовой задаче при числе модулей, превышающем четыре, наблюдается существенное расхождение реального ускорения с оценкой по закону Амдала. Это объясняется главным образом влиянием межмодульных обменов. При меньшем числе вычислительных ядер выигрыш по времени, достигаемый за счет использования библиотеки MPI, не превышает нескольких процентов по сравнению с применением технологии OpenMP.

6. Заключение

Исследована эффективность распараллеливания вычислений при численном моделировании кровотока в квазиодномерном приближении. Рассмотрена задача о моделировании течения крови в отдельном кровеносном сосуде. Построение параллельного алгоритма произведено при помощи метода декомпозиции области. В каждой подобласти задача на каждом шаге по времени расщепляется на гиперболическую и параболическую подзадачи при помощи расщепления Годунова. Для решения гиперболической подзадачи используется интегро-интерполяционный метод, основанный на схеме MUSCL. Для интегрирования по времени применяются методы Рунге-Кутты и Адамса-Башфорта второго порядка. Для решения параболической подзадачи используется метод Кранка-Николсона. На стыках подобластей интерфейсные условия образуют нелинейные системы с тремя неизвестными. Эти системы

решаются при помощи метода Ньютона. С помощью профилировщика AMD CodeAnalyst была определена структура временных затрат при решении тестовой задачи в последовательном режиме. При помощи закона Амдала получены оценки максимально возможного ускорения при распараллеливании наиболее дорогостоящих с вычислительной точки зрения операций. При реализации полученного алгоритма в разработанном авторами настоящей работы программном комплексе использовались технология OpenMP и библиотека MPI. Расчеты проводились на учебно-вычислительном кластере кафедры ФН-2 "Прикладная математика" МГТУ им. Н.Э. Баумана.

Результаты вычислительных экспериментов показали, что выигрыш по времени, достигаемый за счет использования библиотеки MPI, не превышает нескольких процентов по сравнению с применением технологии OpenMP. В связи с этим, принимая во внимания простоту распараллеливания алгоритмов посредством OpenMP, можно остановить выбор на данной технологии, однако использование MPI позволяет сделать программный комплекс универсальным – работающим как на системах с общей памятью, так и на системах с распределенной памятью.

Благодарности

Работа выполнена при частичной финансовой поддержке гранта Президента Российской Федерации для молодых российских ученых-кандидатов наук (проект МК-743.2018.8).

Список литературы

- 1]. Quarteroni A., Formaggia L. Mathematical Modelling and Numerical Simulation of the Cardiovascular System. Handbook on numerical analysis, Ed. By P. G. Ciarlet, J. L. Lions. Amsterdam: Elsevier, 2004, 101 p. DOI: 10.1016/S1570-8659(03)12001-7
- 2]. Quarteroni A., Formaggia L., Veneziani A. Cardiovascular Mathematics: Modeling and Simulation of the Circulatory System. Milano: Springer, 2011, 526 p.
- 3]. Goals – euHeart. URL: https://www.euheart.eu/index_id_27.html (дата обращения: 09.05.2018).
- 4]. Formaggia L., Lamponi D., Quarteroni A. One dimensional models for blood flow in arteries. Journal of Engineering Mathematics, vol. 47, 2003, pp. 251-276. DOI: 10.1023/B:ENGI.0000007980.01347.29.
- 5]. Azer K., Peskin C. S. A one-dimensional model of blood flow in arteries with friction and convection based on the Womersley velocity profile. Cardiovasc. Eng., vol. 7, 2007, pp. 51–73. DOI: 10.1007/s10558-007-9031-y
- 6]. Sherwin S. J., Franke V., Peiro J., Parker K. One-dimensional modeling of vascular network in space-time variables. Journal of Engineering Mathematics. vol. 47, 2003, pp. 217–250. DOI: 10.1023/B:ENGI.0000007979.32871.e2
- 7]. Wang X., Delestre O., Fullana J.-M., Saito M., Ikenaga Y., Matsukawa M., Lagree P.-Y. Comparing different numerical methods for solving arterial 1D flows in networks. Comput. Methods Appl. Mech. Eng., vol. 15, 2012, pp. 61–62. DOI: 10.1080/10255842.2012.713677

- [8]. Wang X., Fullana J.-M., Lagrèe P.-Y. Verification and comparison of four numerical schemes for a 1D viscoelastic blood flow model. *Computer Methods in Biomechanics and Biomedical Engineering*, 2014, pp.1-22. DOI: 10.1080/10255842.2014.948428
- [9]. Аветисян А.И., Бабкова В.В., Гайсарян С.С., Губарь А.Ю. Разработка параллельного программного обеспечения для решения трехмерной задачи о рождении торнадо по теории Николаевского. *Матем. моделирование*, том 20, № 8, 2008 г., стр. 28–40.
- [10]. Марчевский И.К., Токарева С.А. Сравнение эффективности параллельных алгоритмов решения задач газовой динамики на разных вычислительных комплексах. *Вестник МГТУ им. Н.Э. Баумана. Серия "Естественные науки"*, № 1 2009 г., стр. 90–97.
- [11]. Марчевский И.К., Щеглов Г.А. Применение параллельных алгоритмов при решении задач гидродинамики методом вихревых элементов. *Вычислительные методы и программирование*, том 11, стр. 105–110.
- [12]. Морева В.С. Способы ускорения вычислений при решении плоских задач аэродинамики методов вихревых элементов. *Вестник МГТУ им. Н.Э. Баумана. Серия "Естественные науки"*, № 5, 2011 г., стр. 83–95.
- [13]. Лукин В.В., Марчевский И.К., Морева В.С., Попов А.Ю., Шаповалов К.Л., Щеглов Г.А. Учебно-экспериментальный вычислительный кластер. Ч. 2. Примеры решения задач. *Вестник МГТУ им. Н.Э. Баумана. Серия "Естественные науки"*. 2012. № 4. С. 82–102.
- [14]. Марчевский И.К., Пузикова В.В. Исследование эффективности распараллеливания вычислений при моделировании течений вязкой несжимаемой среды методом LS-STAG на системах с общей памятью. *Вычислительные методы и программирование*, том 16, 2015 г., стр. 595–606.
- [15]. Пузикова В.В. Реализация параллельных вычислений в программном комплексе «LS-STAG_turb» для моделирования течений вязкой несжимаемой среды на системах с общей памятью. *Труды ИСП РАН*, том 28, вып. 1, 2016 г., стр. 221-242. DOI: 10.15514/ISPRAS-2016-28(1)-13
- [16]. MPICH Overview | MPICH. URL: <https://www.mpich.org/about/overview/> (дата обращения: 09.05.2018).
- [17]. Avetisyan A.I., Gaisaryan S.S., Ivannikov V.P., Padaryan V.A. Productivity prediction of MPI programs based on models. *Autom. Remote Control.*, vol. 68, 2007, pp. 750-759. DOI: 10.1134/S0005117907050037
- [18]. Intel (R) Cilk (TM) Plus | Intel® Software. URL: <https://software.intel.com/ru-ru/node/522579> (дата обращения 09.05.2018).
- [19]. Reinders J. *Intel Threading Building Blocks: Outfitting C++ for Multi-Core Processor Parallelism*. Sebastopol: O'Reilly, 2007, 336 p.
- [20]. OpenMP FAQ – OpenMP. URL: <https://www.openmp.org/about/openmp-faq/> (дата обращения: 09.05.2018).
- [21]. Quarteroni A., Valli A. *Domain decomposition methods for partial differential equations*. Oxford: Clarendon Press, 1999, 360 p.
- [22]. Годунов С.К., Забродин А.В., Иванов М.Я., Крайко А.Н., Прокопов Г.П. Численное решение многомерных задач газовой динамики. М.: Изд-во "Наука", 1976. 400 с.
- [23]. Drongowski P., Lei Yu, Swehosky F., Suthikulpanit S., Richter R., Incorporating Instruction-Based Sampling into AMD CodeAnalyst. 2010 IEEE International Symposium on Performance Analysis of Systems & Software (ISPASS 2010), White Plains, NY, 2010, pp. 119-120. DOI: 10.1109/ISPASS.2010.5452049.

- [24]. Гергель В.П. *Высокопроизводительные вычисления для многопроцессорных многоядерных систем*. М.: Изд-во Моск. ун-та, 2010, 544 с.
- [25]. Лукин В.В., Марчевский И.К. Учебно-экспериментальный вычислительный кластер. Ч.1. Инструментарий и возможности. *Вестник МГТУ им. Н.Э. Баумана. Серия «Естественные науки»*, № 4, 2011 г., стр. 28–43.

Application of parallel algorithms for numerical simulation of quasi-one dimensional blood flow

A.N. Avdeeva <aan--bmstu@yandex.ru>

V.V. Puzikova <vvp@dms-at.ru>

Federal state budgetary institution of higher professional education «Bauman Moscow State Technical University (National research university of technology)», 5/1, 2-nd Baumanskaya st., Moscow, 105005, Russia

Abstract. The main goal of modern hemodynamic simulation is the prediction of blood pressure in the arteries, as well as the study of the various factors complex effect on the cardiovascular system characteristics. Quasi-one dimensional models of blood flow through blood vessels are the most popular. They allow to model the blood flow in the entire vascular system. Since full-scale simulation of the cardiovascular system requires large computational costs, the problem of parallelizing computation is actual. In this paper, the efficiency study was carried out for parallel algorithms in the numerical simulation of blood flow in the quasi-one-dimensional approximation. For simplicity, we consider the problem of the blood flow simulation in a separate blood vessel. When constructing a parallel algorithm, the domain decomposition method was applied. In each subdomain, the problem at each time step splits into a hyperbolic and parabolic subproblems. To solve the hyperbolic subtask, an integro-interpolation method based on the MUSCL scheme is used. To integrate over time, the second order Runge-Kutta and Adams-Bashfort methods are applied. The Crank-Nicholson method is used to solve the parabolic subproblem. At the subdomains conjunctions, the interface conditions are non-linear systems with three unknowns. These systems are solved using the Newton method. The time-cost structure was obtained for solving the test problem in a serial mode using the AMD CodeAnalyst profiler. Computations were carried out at the cluster of the BMSTU "Applied Mathematics" FN-2 department. The computational results showed that the time gain achieved by using the MPI library does not exceed a few percent in comparison with the OpenMP technology usage. Taking into account the simplicity of parallelizing algorithms through OpenMP, we can choose this technology, but MPI usage allows to make the software package universal (it can work both on shared memory systems and on distributed memory systems).

Keywords: OpenMP technology; MPI; quasi-one dimensional model; blood flow; domain decomposition method; parallel algorithm; cluster; MUSCL scheme; Godunov splitting.

DOI: 10.15514/ISPRAS-2018-30(2)-15

For citation: Avdeeva A.N., Puzikova V.V. Application of parallel algorithms for numerical simulation of quasi-one dimensional blood flow. *Trudy ISP RAN/Proc. ISP RAS*, vol. 30, issue 2, 2018, pp. 301-316 (in Russian). DOI: 10.15514/ISPRAS-2018-30(2)-15

References

- [1]. Quarteroni A., Formaggia L. *Mathematical Modelling and Numerical Simulation of the Cardiovascular System*. Handbook on numerical analysis, Ed. By P. G. Ciarlet, J. L. Lions. Amsterdam: Elsevier, 2004, 101 p. DOI: 10.1016/S1570-8659(03)12001-7
- [2]. Quarteroni A., Formaggia L., Veneziani A. *Cardiovascular Mathematics: Modeling and Simulation of the Circulatory System*. Milano: Springer, 2011, 526 p.
- [3]. Goals – euHeart. URL: https://www.euheart.eu/index_id_27.html (дата обращения: 09.05.2018).
- [4]. Formaggia L., Lamponi D., Quarteroni A. One dimensional models for blood flow in arteries. *Journal of Engineering Mathematics*, vol. 47, 2003, pp. 251-276. DOI: 10.1023/B:ENGI.0000007980.01347.29.
- [5]. Azer K., Peskin C. S. A one-dimensional model of blood flow in arteries with friction and convection based on the Womersley velocity profile. *Cardiovasc. Eng.*, vol. 7, 2007, pp. 51–73. DOI: 10.1007/s10558-007-9031-y
- [6]. Sherwin S. J., Franke V., Peiro J., Parker K. One-dimensional modeling of vascular network in space-time variables. *Journal of Engineering Mathematics*. vol. 47, 2003, pp. 217–250. DOI: 10.1023/B:ENGI.0000007979.32871.e2
- [7]. Wang X., Delestre O., Fullana J.-M., Saito M., Ikenaga Y., Matsukawa M., Lagree P.-Y. Comparing different numerical methods for solving arterial 1D flows in networks. *Comput. Methods Appl. Mech. Eng.*, vol. 15, 2012, pp. 61–62. DOI: 10.1080/10255842.2012.713677
- [8]. Wang X., Fullana J.-M., Lagree P.-Y. Verification and comparison of four numerical schemes for a 1D viscoelastic blood flow model. *Computer Methods in Biomechanics and Biomedical Engineering*, 2014, pp.1-22. DOI: 10.1080/10255842.2014.948428
- [9]. Avetisyan A.I., Babkova V.V., Gaisaryan S.S., Gubar' A.Yu. Development of parallel software for 3D tornado arising modeling by Nikolaevskiy theory. *Matematicheskoe modelirovanie [Mathematical Models and Computer Simulations]*, vol. 20, № 8, 2008, pp. 28–40 (in Russian)
- [10]. Marchevskii I.K., Tokareva S.A. Comparison of the parallel algorithms effectiveness for solving gas dynamics problems on different computational complexes. *Vestnik MGTU im. N.E. Baumana. Ser. Estestvennyye nauki [Herald of the Bauman Moscow State Technical University. Series Natural Sciences]*, № 1, 2009, pp. 90–97 (in Russian)
- [11]. Marchevskii I.K., Shcheglov G.A. Application of parallel algorithms for solving hydrodynamic problems by the vortex element method. *Vychislitel'nye metody i programirovanie [Computational methods and programming]*, vol. 11, 2010, pp. 105–110 (in Russian)
- [12]. Moreva V.S. Ways for calculation speed-up in solving 2D aerodynamics problems by vortex element method. *Vestnik MGTU im. N.E. Baumana. Ser. Estestvennyye nauki [Herald of the Bauman Moscow State Technical University. Series Natural Sciences]*, № S, 2011, pp. 83–95 (in Russian)
- [13]. Lukin V.V., Marchevskii I.K., Moreva V.S., Popov A.Yu, Shapovalov K.L., Shcheglov G.A. Educational-experimental computing cluster. Part 2. Examples of problem solving. *Vestnik MGTU im. N.E. Baumana. Ser. Estestvennyye nauki [Herald of the Bauman Moscow State Technical University. Series Natural Sciences]*, № 4, 2012, pp. 82–102 (in Russian)
- [14]. Marchevskiy I.K., Puzikova V.V. Efficiency investigation of computation parallelization for viscous incompressible flow simulation on systems with shared memory. *Vychislitel'nye metody i programirovanie [Computational methods and programming]*, vol. 16, 2015, pp. 595–606 (in Russian)
- [15]. Puzikova V.V. Realization of parallel computations in the software package «LS-STAG_turb» for viscous incompressible flow simulation on systems with shared memory. *Trudy ISP RAN / Proc. ISP RAS*, vol. 28, issue 1. 2016, pp. 221-242. DOI: 10.15514/ISPRAS-2016-28(1)-13 (in Russian)
- [16]. MPICH Overview | MPICH. URL: <https://www.mpich.org/about/overview/> (accessed: 09.05.2018).
- [17]. Avetisyan A.I., Gaisaryan S.S., Ivannikov V.P., Padaryan V.A. Productivity prediction of MPI programs based on models. *Autom. Remote Control.*, vol. 68, 2007, pp. 750-759. DOI: 10.1134/S0005117907050037
- [18]. Intel (R) Cilk (TM) Plus | Intel® Software. URL: <https://software.intel.com/ru-ru/node/522579> (accessed: 09.05.2018).
- [19]. Reinders J. *Intel Threading Building Blocks: Outfitting C++ for Multi-Core Processor Parallelism*. Sebastopol: O'Reilly, 2007, 336 p.
- [20]. OpenMP FAQ – OpenMP. URL: <https://www.openmp.org/about/openmp-faq/> (дата обращения: 09.05.2018).
- [21]. Quarteroni A., Valli A. *Domain decomposition methods for partial differential equations*. Oxford: Clarendon Press, 1999, 360 p.
- [22]. Godunov S.K., Zabrodin A.V., Ivanov M.Ya., Kraiko A.N., Prokopov G.P. *Numerical solution of gas dynamics multidimensional problems*. Moscow: Science Publ., 1976. 400 p. (in Russian)
- [23]. Drongowski P., Lei Yu, Swehosky F., Suthikulpanit S., Richter R., Incorporating Instruction-Based Sampling into AMD CodeAnalyst. 2010 IEEE International Symposium on Performance Analysis of Systems & Software (ISPASS 2010), White Plains, NY, 2010, pp. 119-120. DOI: 10.1109/ISPASS.2010.5452049.
- [24]. Gergel' V.P. High-performance computing for multi-core systems. *Moscow: Moscow University Publ.*, 2010, 544 p. (in Russian)
- [25]. Lukin V.V., Marchevskii I.K. Educational-experimental computing cluster. Part 1. Tools and capabilities. *Vestnik MGTU im. N.E. Baumana. Ser. Estestvennyye nauki [Herald of the Bauman Moscow State Technical University. Series Natural Sciences]*, № 4, 2011, pp. 28–43 (in Russian)