

УДК 004.65

## АЛГОРИТМ С ПЕРЕКРЫТИЯМИ ДЛЯ РЕШЕНИЯ СИСТЕМЫ УРАВНЕНИЙ МЕЛКОЙ ВОДЫ НА ПАРАЛЛЕЛЬНЫХ КОМПЬЮТЕРАХ С РАСПРЕДЕЛЕННОЙ ПАМЯТЬЮ

В. В. Калмыков<sup>1</sup>, Р. А. Ибраев<sup>2</sup>

<sup>1</sup>vvk88@mail.ru, <sup>2</sup>ibrayev@mail.ru

<sup>1</sup>ФГБОУ ВПО «Московский государственный университет имени М. В. Ломоносова» (МГУ)

<sup>1,2</sup>ФГБОН «Институт океанологии им. П. П. Ширшова Российской академии наук» (ИО РАН)

<sup>2</sup>ФГБОН «Институт вычислительной математики Российской академии наук» (ИВМ РАН)

<sup>2</sup>ФГБУ «Гидрометцентр России»

<sup>2</sup>ФГБОУ ВПО «Московский физико-технический институт (государственный университет)» (МФТИ)

Поступила в редакцию 01.09.2013

**Аннотация.** Предложен и протестирован программный алгоритм ускорения счета системы уравнений мелкой воды на параллельных компьютерах с распределенной памятью. Алгоритм базируется на оптимизации объема вычислений, межпроцессорных обменов и количества пересылаемой информации, которая позволяет значительно улучшить масштабируемость вычислительного кода. Проведенные тесты показывают, что алгоритм с перекрытиями дает существенные преимущества при счете на достаточно большом (для данных размеров задачи) числе ядер, когда на ускорение сильное влияние начинают оказывать коммуникационные задержки. Алгоритм реализован в рамках работы одного из блоков параллельной вихреразрешающей модели Мирового океана. Статья рекомендована к публикации программным комитетом Международной суперкомпьютерной конференции "Научный сервис в сети Интернет: экзафлопное будущее".

**Ключевые слова:** уравнения мелкой воды; явные схемы; алгоритм с перекрытиями; массивно-параллельные приложения; MPI.

### ВВЕДЕНИЕ

Появление петафлопных и терафлопных компьютеров открыло беспрецедентные возможности для решения сложных, многокомпонентных задач с высоким разрешением на больших пространственных масштабах.

В частности, в исследованиях межгодовых и климатических изменений состояния Мирового океана стало возможным построение вихреразрешающих глобальных моделей с шагом 5–10 км. В опубликованных недавно работах по исследованию глобального океана [1] и систем оперативной океанологии [2] с применением

моделей с горизонтальными шагами 1/10 градуса и менее показана возможность реалистичного воспроизведения широкого спектра пространственно-временной изменчивости океана. В то же время для исследований климата в основном используются модели Мирового океана на порядок более грубым разрешением, равным 0,5–1° [3]. Поэтому актуальной является задача создания вычислительно эффективных моделей, способных вести на вихреразрешающих сетках исследования климата Мирового океана, предполагающие интегрирование системы на 100–1000 лет.

В данной работе мы рассмотрим применение алгоритма с перекрытиями для решения системы уравнений мелкой воды, которая является частью многих трехмерных моделей динамики Мирового океана, в том числе и модели, разрабатываемой в Институте вычислительной математики РАН и Институте океанологии им. П. П. Ширшова РАН [4, 5]. Система трехмерных

---

Работа поддержана грантами РФФИ 12-05-31317-мол\_а, 12-05-01155-а, 13-05-01141-а, стипендией Президента РФ для аспирантов № 136 (28/02/2013), проектом Программы фундаментальных исследований Президиума РАН № 23, грантом Министерства образования и науки по договорам №№ 8344 (17/08/2012) и 8328.

уравнений крупномасштабной динамики океана описывает спектр движений с большим диапазоном характерных скоростей: скорости течений и бароклинных гравитационных волн составляют 1–3 м/с; скорости баротропных гравитационных волн составляют 30–200 м/с. Применение явного метода аппроксимации нестационарных уравнений приводит к необходимости удовлетворить условию Куранта для всего диапазона описываемых системой уравнений волн, включая баротропные гравитационные волны. Наиболее значимые процессы (на настоящем уровне наших знаний) для климатических явлений в океане определяются течениями и бароклинными волнами. Способом обойти жесткое ограничение на шаги по времени является разделение решения системы уравнений динамики океана на бароклинные, сравнительно медленные движения, и баротропные, быстрые движения [6]. Система уравнений для баротропных движений получается путем интегрирования исходной трехмерной системы по вертикали, в результате мы получаем систему уравнений мелкой воды, которую решаем с шагом по времени, меньшим на один-два порядка по сравнению с шагом по времени для решения трехмерных уравнений для бароклинных движений.

Уменьшение шага по времени соответствует увеличению числа итераций цикла решения системы уравнений мелкой воды, что, за счет необходимости межпроцессорных обменов на каждой итерации, ведет к тому, что простейшая явная схема решения уравнения мелкой воды становится узким местом программы с точки зрения параллельной реализации.

Данная проблема отмечена, например, в работе [7]. Авторы использовали одномерное разбиение области и 3 типа обмениваемых данных между ядрами-соседями, что соответствует 6 сообщениям за один шаг цикла. Они получили очень низкое ускорение и связали результаты с большим числом межпроцессорных обменов (21600 сообщений за 3600 шагов по времени).

В следующем эксперименте авторы попытались решить проблему коммуникаций за счет перекрытия вычислений и обменов. Они рассчитывали приграничные ячейки, асинхронно отправляли их, и затем досчитывали внутреннюю область. В конце шага проверяли, что все обмены завершились, и выполняли следующую итерацию цикла. Данный метод также не принес существенно лучших результатов ускорения, к тому же усложнил код.

Другой метод был предложен в работе [8]. Авторы предлагают использовать параллельный метод коллокаций и тестируют его на примере решения системы уравнений мелкой воды. Однако их метод, хотя и дает суперлинейное ускорение для небольшого (меньшего 8) числа процессоров, авторы отмечают, что при дальнейшем увеличении процессорной решетки затраты на коммуникации будут быстро расти. Уже на 16 ядрах ускорение падает в 4 раза по сравнению с результатами на 8 ядрах.

Важно отметить, что процедура решения системы уравнений мелкой воды является частью трехмерной модели Мирового океана. Используемые в процедуре массивы значений различных функций подчиняются двумерной декомпозиции области, используемой во всей модели для всех глобальных массивов. Поэтому использование неявного метода решения или сложного метода декомпозиции области (пример такого метода в работе [9]) приведет к нетривиальным обменам данными между вычислительными ядрами, которые затронут еще и массивы внешних для процедуры решения системы уравнений мелкой воды модельных величин. Это приведет к дополнительным обменам и существенному усложнению идеи естественно-го параллелизма.

В данной работе мы рассматриваем применение алгоритма с перекрытиями для решения системы уравнений мелкой воды на параллельных компьютерах с распределенной памятью. Алгоритм базируется на оптимизации вычислений, пересылок и объема пересылаемой информации.

## ПОСТАНОВКА ЗАДАЧИ

Система уравнений мелкой воды выводится из трехмерной системы уравнений крупномасштабной динамики океана путем интегрирования уравнений по вертикали [10]:

$$U_t - fV = -gH \frac{\partial \eta}{\partial x} + R^U,$$

$$V_t + fU = -gH \frac{\partial \eta}{\partial y} + R^V,$$

$$\frac{\partial \eta}{\partial t} + \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = R^n.$$

Здесь  $U$ ,  $V$  – проинтегрированные по вертикали (баротропные) потоки,  $\eta$  – уровень океана,  $f$  – сила Кориолиса,  $g$  – ускорение свободного падения,  $H$  – глубина океана,  $R^U$ ,  $R^V$ ,  $R^n$  – члены источника.

### РАЗНОСТНАЯ СХЕМА

Для решения системы применим явную схему аппроксимации по времени:

$$\frac{\eta^{p+1} - \eta^p}{\tau} + (\delta_x \bar{U}^y)^p + (\delta_y \bar{V}^x)^p = R^\eta,$$

$$\frac{U^{p+1} - U^p}{\tau} - fV^p = -gH(\delta_x \bar{\eta}^y)^{p+1} + R^U,$$

$$\frac{V^{p+1} - V^p}{\tau} + fU^p = -gH(\delta_y \bar{\eta}^x)^{p+1} + R^V,$$

где  $p$  – целочисленный момент времени для интегрирования баротропной системы с временным шагом  $\tau$ ;  $\delta$  – оператор разностного дифференцирования. Разрешим уравнения относительно  $\eta^{p+1}$ ,  $U^{p+1}$ ,  $V^{p+1}$ :

$$\eta^{p+1} = -\tau(\delta_x \bar{U}^y + \delta_y \bar{V}^x)^p + \tau R^\eta + \eta^p,$$

$$U^{p+1} = -\tau g H(\delta_x \bar{\eta}^y)^{p+1} + \tau R^U + U^p + \tau f V^p,$$

$$V^{p+1} = -\tau g H(\delta_y \bar{\eta}^x)^{p+1} + \tau R^V + V^p - \tau f U^p.$$

Решение ищется с шагом  $\tau \ll \Delta t$  на временном интервале  $2\Delta t: [t^l; t^l + 2\Delta t]$ , где  $t^l$  – момент времени на временной сетке решения 3-мерной модели океана. Количество шагов интегрирования равно  $2 \cdot pl$ , где  $pl = \Delta t / \tau$ .

Начальные условия:

$$U^{p=0} = U^l, V^{p=0} = V^l, \eta^{p=0} = \zeta^l.$$

### ПРЯМОЙ АЛГОРИТМ

Прямой алгоритм вычисления решения на параллельных компьютерах с распределенной памятью состоит в последовательном расчете по формулам и обмене данными с соседними вычислительными ядрами.

Одномерный по пространству вариант пространственно-временной сетки изображен на рис. 1. Шаги по времени отсчитываются по вертикальной оси, по горизонтали отложены узлы одномерной по пространству сетки. Пусть расчетная область в соответствии с декомпозицией области состоит из  $K$  подобластей. Рассмотрим  $k$ -ю подобласть,  $k = \overline{1, K}$ , соответствующую  $k$ -му вычислительному ядру. Предположим, что размеры подобластей одинаковы и равны  $n$ . Для примера возьмем  $n = 5$ . В соответствии с прямым алгоритмом решение в черных узлах рассчитывается на соседних  $(k-1, k+1)$  ядрах и пересылаются на  $k$ -е ядро с помощью межпро-

цессорных обменов, решение в серых узлах рассчитывается на  $k$ -м ядре. На рисунке изображены первые 6 шагов алгоритма. Здесь используется естественная схема счета – 1 шаг по времени / 1 межпроцессорный обмен:

- **Шаг 1.** Момент времени  $p$ . Получение с помощью межпроцессорного обмена значений массивов в приграничных ячейках.

- **Шаг 2.** Интегрирование системы уравнений на один шаг по времени, явное вычисление неизвестных  $U^{p+1}, V^{p+1}, \eta^{p+1}$  по формулам для  $n$  точек подобласти.

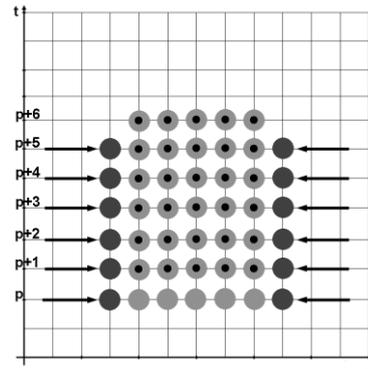


Рис. 1. Схема работы прямого алгоритма на примере одномерной сетки

Тестовые расчеты по прямому алгоритму проводились для массивов следующих размеров:  $2880 \times 1440$  и  $4320 \times 2160$  (или  $4,1 \times 10^6$  и  $9,3 \times 10^6$  точек), что соответствует сеткам в широтно-долготной системе координат с разрешением  $1/8$  и  $1/12$  градуса для Мирового океана. Расчет велся на 57600 и 86400 шагов соответственно. Разбиение на подобласти двумерное равномерное.

Расчеты проводились на суперкомпьютере МВС-100К, входящем в состав Межведомственного суперкомпьютерного центра РАН ([www.jssc.ru](http://www.jssc.ru)).

Вычислительные мощности суперкомпьютера состоят из 1460 вычислительных модулей (11680 процессорных ядер). Базовый вычислительный модуль МВС-100К представляет собой сервер HP Proliant, содержащий два четырехядерных микропроцессора Intel Xeon, работающих на частоте 3 ГГц на 8 Гб оперативной памяти. Вычислительные модули связаны между собой высокоскоростной коммуникационной сетью Infiniband DDR, транспортной и управляющей сетями Gigabit Ethernet.

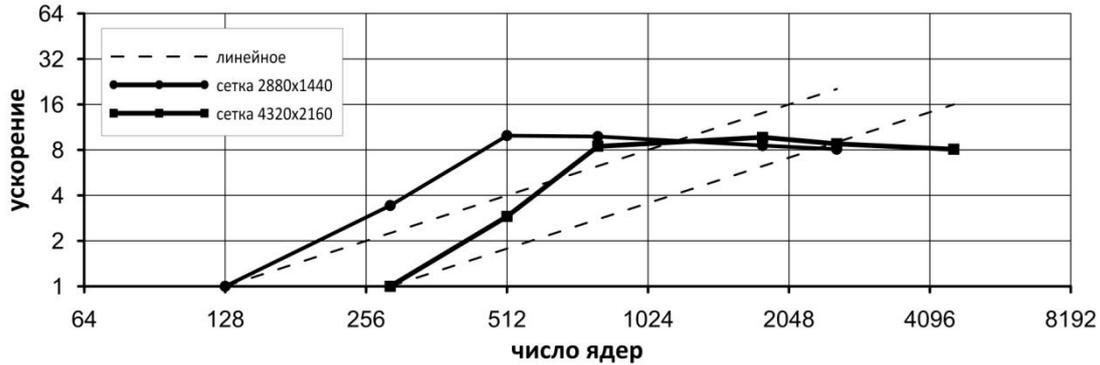


Рис. 2. Масштабируемость прямого алгоритма решения системы уравнений мелкой воды для двух размеров сеток

График ускорения работы прямого алгоритма относительно числа вычислительных ядер для двух сеток приведен на рис. 2. Число вычислительных ядер для первой сетки равнялось 128, 288, 512, 800, 1800, 2592; для второй – 288, 512, 800, 1800, 2592, 4608.

Отметим следующие характеристики масштабирования прямого алгоритма. Во-первых, значительный рост ускорения на 288, 512 и 512, 800 вычислительных ядрах для соответствующих размеров сеток. Дело в том, что ускорение отсчитывается от минимального числа вычислительных ядер (128 и 288), когда массивы данных еще не полностью используют возможности кэш-памяти. По мере роста числа ядер размеры подобластей уменьшаются, что ведет к более эффективному использованию кэш-памяти и появлению суперлинейного ускорения. Во-вторых, падение ускорения при увеличении числа вычислительных ядер, начиная с 512 и 1800 соответственно. Поскольку размеры подобластей, а значит и объемы вычислений, продолжают уменьшаться, увеличение времени решения задачи происходит из-за увеличения коммуникационных задержек.

Решение системы уравнений мелкой воды является частью решения трехмерной модели динамики океана. При этом линейная масштабированность модели океана за исключением системы уравнений мелкой воды сохраняется при количестве вычислительных ядер значительно больших, чем для системы уравнений мелкой воды. Поэтому естественным образом встала задача оптимизации вычислений и пересылок при решении системы уравнений мелкой воды так, чтобы сохранить масштабированность всего кода модели динамики океана. Очевидный подход состоит в том, чтобы уменьшить количество

пересылок, не изменяя порядка вычислений и сохраняя двумерную декомпозицию расчетной области.

### АЛГОРИТМ С ПЕРЕКРЫТИЯМИ

Идея алгоритма с перекрытиями хорошо иллюстрируется геометрически. Если рассмотреть прямую схему решения на  $k$ -м вычислительном ядре, то она будет иметь вид столба, составленного из усеченных пирамид, высотой в 1 шаг по времени. Задача состоит в том, чтобы найти решение в  $n$  серых точках на некоторый момент времени. Теперь увеличим высоту каждой усеченной пирамиды, сохранив ширину верхней грани в  $n$  узлов и соответственно увеличив нижнее основание.

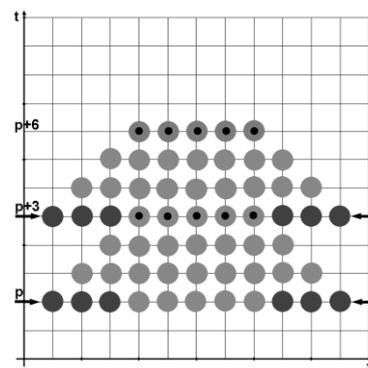


Рис. 3. Схема работы алгоритма с перекрытиями на примере одномерной сетки

Введем числовой параметр  $q$ , обозначающий количество ячеек, на которое будет увеличена одномерная область в каждом направлении.

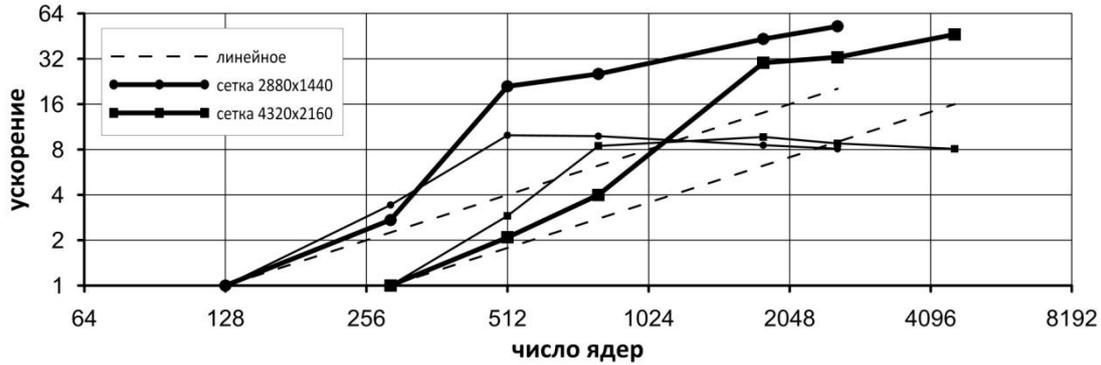


Рис. 4. Масштабируемость алгоритма с перекрытиями решения системы уравнений мелкой воды для двух размеров сеток

Заменим прямой алгоритм решения, а именно: *1 шаг по времени / 1 межпроцессорный обмен* на схему *q шагов по времени / 1 межпроцессорный обмен*. В этом случае решаются те же уравнения, но на расширенной на  $q$  ячеек области, и с каждым шагом эта область сужается на одну ячейку по обоим направлениям.

На рис. 3 приведен пример работы нового алгоритма. Решение в черных узлах по-прежнему рассчитывается на соседних  $(k-1, k+1)$  ядрах и пересылаются на  $k$ -е ядро с помощью (теперь расширенных) межпроцессорных обменов, решение в серых узлах рассчитывается на  $k$ -м ядре. На рисунке сделано 2 шага алгоритма, то есть опять рассчитаны 6 шагов временного цикла. Алгоритм сейчас выглядит так:

- **Шаг 1.** Момент времени  $p$ . Получение с помощью межпроцессорного обмена значений массивов в  $q$  (для примера  $q = 3$ ) приграничных ячейках.
- **Шаг 2.** Интегрирование системы уравнений на один шаг по времени, явное вычисление неизвестных  $U^{p+1}, V^{p+1}, \eta^{p+1}$  по формулам для  $n + 2 \cdot q - 2 \cdot 1$  точек подобласти.
- **Шаг 1+2.** Интегрирование системы уравнений на один шаг по времени, явное вычисление неизвестных  $U^{p+2}, V^{p+2}, \eta^{p+2}$  по формулам для  $n + 2 \cdot q - 2 \cdot 2$  точек подобласти.
- ...
- **Шаг 1+q.** Интегрирование системы уравнений на один шаг по времени, явное вычисление неизвестных  $U^{p+q}, V^{p+q}, \eta^{p+q}$  по формулам для  $n + 2 \cdot q - 2 \cdot q$  точек подобласти.

В результате через  $q$  шагов область, в которой рассчитано решение, сузится до размера

подобласти для  $k$ -го вычислительного ядра. Далее обрабатывает обменная процедура, вычислительное ядро опять получает расширенную область и цикл повторяется. То есть, увеличивая расчетную область на  $q$  точек в каждом направлении, мы уменьшаем число обменов в  $q$  раз. Несмотря на то, что при межпроцессорном обмене объем пересылаемой информации увеличивается в  $q$  раз, пересылка 1 раз  $2q$  чисел значительно экономичнее, чем  $q$  раз по 2 числа. Очевидно, прямой алгоритм можно рассматривать как алгоритм с перекрытиями с параметром  $q$ , равным 1.

Алгоритм прямо переносится на двумерный случай. Если размер подобласти, соответствующей  $k$ -му вычислительному ядру, равен  $n_1 \times n_2$ , то межпроцессорные обмены должны обеспечить заполнение массива размером  $(n_1 + 2 \cdot q) \times (n_2 + 2 \cdot q)$  на временном шаге  $p$ . Процедура межпроцессорных обменов в двумерном случае будет несколько сложнее. Программа решения системы уравнений на параллельном компьютере с распределенной памятью по быстрому алгоритму будет выглядеть так:

```

do p=1,pl/q
  call send_receive (n+2q data)
  do pq=q,1,-1
    calculate u,v,eta
  end do
end do

```

Алгоритм с перекрытиями тестировался с теми же параметрами задачи. Параметр расширения подобласти  $q$  брался равным 10. Ускорение времени решения системы уравнений мелкой воды алгоритмом с перекрытиями относительно времени решения системы прямым алго-

ритмом на 128 и 288 вычислительных ядрах представлено на рис. 4. Для сравнения более тонкими линиями обозначены результаты ускорения начальной версии, приведенные на рис. 2.

По графику можно отследить несколько тенденций. Во-первых, прямой алгоритм дает лучшие результаты на небольшом числе процессоров. Это логично, так как алгоритм с перекрытиями использует массивы, расширенные на 10 ячеек в каждом направлении. Для небольшого числа процессоров, когда коммуникационные задержки еще невелики, это ведет к увеличению расчетной области и соответствующему росту времени вычислений. По этой же причине ускорение отсчитывается от времени работы начальной версии, чтобы искусственно не завышать показатели.

Поэтому для максимального ускорения на всех размерах процессорных решеток можно до некоторого числа процессоров использовать значение  $q$ , равное 1, а далее увеличивать  $q$ . Хотя нас больше интересуют результаты в правой части графика, так как модель Мирового океана с хорошим разрешением требует значительных вычислительных мощностей.

Во-вторых, ясно, что при минимальном числе процессоров, от которых отсчитывается ускорение, данные не полностью используют возможности кэш-памяти. Вследствие этого дальнейшее уменьшение областей при отсутствии коммуникационных проблем ведет к значительному росту ускорения.

## ВЫВОДЫ

В работе предложен и протестирован программный алгоритм ускорения времени счета системы уравнений мелкой воды на параллельных компьютерах с распределенной памятью. Результаты ускорения представляются удовлетворительными. Метод был введен в разрабатываемую нами модель Мирового океана, где также показал хорошие результаты.

Тесты показывают, что предложенный подход позволяет существенно ускорить решение задачи по сравнению с прямым алгоритмом. По всей вероятности, близкая к линейной масштабируемость кода может быть продлена как минимум до 10–20 тыс. вычислительных ядер. Мы планируем проверить это в ближайшем будущем для более мелких пространственных сеток.

Применение алгоритма с перекрытиями обосновано лишь в функциях, по своей структуре перенасыщенных обменными процедурами. То есть замена всех явных схем в программе

описанным образом приведет к неоправданному усложнению кода и вряд ли повысит эффективность, так как в крупных моделях между функциями межпроцессорной коммуникации содержится достаточное количество локальных вычислений и обмены в этом смысле довольно редки. А вот для функций вида шаг–обмен, где под шагом подразумевается вычисление нескольких потоков и операторов, метод может дать существенные результаты. Особенно при использовании большого числа процессоров, что становится необходимостью при разработке современных глобальных моделей высокого пространственного разрешения.

Алгоритм с перекрытиями представляет интерес и в дальнейшем развитии модели Мирового океана, связанной с созданием глобальной модели Земной системы [11]. В одном из заключительных пунктов конференции по созданию глобальных систем Coupling Technologies for Earth System Modeling: Today and Tomorrow [12] были отмечены некоторые условия, диктуемые наступающей эксафлопной эрой, а именно: ограничение памяти одного процессорного элемента, неоднородность вычислительных ресурсов (разделение их на классические CPU и графические GPU), повышение стоимости межпроцессорных обменов вследствие значительного роста общего числа ядер при сохранении текущей скорости интерконнекта. В связи с этим было предложено ограничить внутримодельные обмены как раз за счет перекрывающихся избыточных вычислений, что в условиях большого числа ядер и медленной сети позволяет избежать падения ускорения.

## СПИСОК ЛИТЕРАТУРЫ

1. Maltrud M. E., McClean J. L. An eddy resolving global  $1/10^\circ$  ocean simulation // Ocean Modelling. 2005. Vol. 8. P. 31–54.
2. Dombrowsky E. L., Bertino G. B., Brassington E. P., Davidson Ch. F., Hurlburt H. E., Kamachi M., Lee T., Martin M. J., Mei S., Tonani M. GODAE Systems in operation // Oceanography. 2009. Vol. 22, No. 3. P. 80–95.
3. Володин Е. М., Дианский Н. А. Моделирование изменений климата в XX–XXII столетиях с помощью совместной модели общей циркуляции атмосферы и океана // Изв. РАН, ФАО. 2006. Т. 42, № 3. С. 291–306.
4. Ибраев Р. А., Калмыков В. В., Ушаков К. В. Мировой океан, изменения климата и суперкомпьютерное моделирование // Суперкомпьютерные технологии в науке, образовании и промышленности: сб. науч. ст. / Под ред. акад. В. А. Садовниченко, акад. Г. И. Савина, чл.-кор. РАН Вл. В. Воеводина. М.: Изд-во Моск. ун-та, 2010. С. 49–54.
5. Ибраев Р. А., Калмыков В. В., Ушаков К. В., Хабеев Р. Н. Вихререзающая  $1/10^\circ$  модель Мирового океана

на // Экологическая безопасность прибрежной и шельфовой зон и комплексное использование ресурсов шельфа: сб. науч. тр. Севастополь: Изд-во МГИ НАНУ, 2011. Т. 25, № 2. С. 30–44.

6. **Ибраев Р. А.** Математическое моделирование термогидродинамических процессов в Каспийском море. М.: ГЕОС, 2008.

7. **Pau J. C., Sanders B. F.** Performance of parallel implementations of an explicit finite-volume shallow-water model // *ASCE J. Computing in Civil Engineering*. 2006. Vol. 20, No. 2. P. 99–110.

8. **Tsai Y. M., Kuo H. C., Chang Y. C.** A new parallel domain-decomposed Chebyshev collocation method for atmospheric and oceanic modeling // *Terrestrial, Atmospheric and Oceanic Sciences*. 2012. Vol. 23. P. 439–450.

9. **Chefter J. G., Chu C. K., Keyes D. E.** Domain decomposition for the shallow water equations // *Contemporary Mathematics: Proc. 7th Int. Conf. on Domain Decomposition Methods in Sci. and Eng.* 1994. P. 485–490.

10. **Killworth P. D., Stainforth D., Webb D. J., Paterson S.** The development of a free surface Bryan–Cox–Semtner model // *J. Phys. Oceanogr.* 1991. Vol. 21. P. 1333–1348.

11. **Толстых М. А., Ибраев Р. А., Володин Е. М., Ушаков К. В., Калмыков В. В., Шляева А. В., Мизяк В. Г., Хабеев Р. Н.** Модели глобальной атмосферы и Мирового океана: алгоритмы и суперкомпьютерные технологии. М.: МГУ, 2013. 144 с.

12. **Valcke S., Balaji V., Craig A., DeLuca C., Dunlap, R., Ford R. W., Jacob R., Larson J., O’Kuinghtons R., Riley G. D., Vertenstein M.** Coupling technologies for Earth System modelling // *Geosci. Model Dev.* 2012. Vol. 5. P. 1589–1596.

#### ОБ АВТОРАХ

**КАЛМЫКОВ Владимир Владимирович**, асп. каф. выч. технологий и моделирования фак-та выч. математики и кибернетики. Дипл. математик, системный программист (ВМК МГУ, 2010). Готовит дис. о прогр. компл. совместн. моделирования системы «океан–атмосфера–лед–почва».

**ИБРАЕВ Рашит Ахметзиевич**, гл. науч. сотр. Ин-та выч. математики РАН. Дипл. инж.-физик (МФТИ, 1982). Д-р физ.-мат. наук (ИВМ РАН, 2002). Чл.-корр. РАН (2008). Иссл. в обл. физики океана.

#### METADATA

**Title:** The overlapping algorithm for solving shallow water equations on massively-parallel architectures with distributed memory.

**Authors:** V. V. Kalmykov and R. A. Ibrayev.

#### Affiliation:

Moscow State University of Lomonosov (MSU), Russia;  
P. P. Shirshov Institute of Oceanology Russian Academy of Sciences (IO RAS), Russia;  
Institute of Numerical Mathematics Russian Academy of Sciences (INM RAS), Russia;  
Hydrometcentre of Russia;  
Moscow Institute of Physics and Technology (MIPT), Russia.

**Email:** vvk88@mail.ru, ibrayev@mail.ru

**Language:** Russian.

**Source:** Vestnik UGATU (scientific journal of Ufa State Aviation Technical University), vol. 17, no. 5 (58), pp. 252-259, 2013. ISSN 2225-2789 (Online), ISSN 1992-6502 (Print).

**Abstract:** The paper proposes and tests algorithm for acceleration shallow water equations on parallel computers with distributed memory. The algorithm is based on the optimization of computations, interprocessor communications and the amount of transferred information, which can significantly improve the scalability of computer code. The tests show that the overlapping algorithm provides serious advantages for large communicator sizes of the model when the acceleration strongly depends of communications delays. Algorithm is realized as one of the blocks of parallel eddy-resolving World Ocean Model. The paper was recommended for publication by the program committee of the International Supercomputer Conference "Scientific service on the Internet: exaflops future."

**Key words:** shallow water equations; explicit schemes; overlapping algorithm; massively-parallel applications; MPI.

#### References (English transliteration):

1. M. E. Maltrud and J. L. McClean, "An eddy resolving global 1/10° ocean simulation," *Ocean Modelling*, vol. 8, pp. 31-54, 2005.
2. E. L. Dombrowsky, G. B. Bertino, E. P. Brassington, C. F. Davidson, H. E. Hurlburt, M. Kamachi, T. Lee, M. J. Martin, S. Mei, and M. Tonani, "GODAE Systems in operation," *Oceanography*, vol. 22, no. 3, pp. 80-95, 2009.
3. E. M. Volodin and N. A. Diansky, "Climate change modelling in the XX-XXII centuries using coupled model of general circulation of the atmosphere and ocean," (in Russian), *Math. Russian Academy of Sciences*, vol. 42, no. 3, pp. 291-306, 2006.
4. R. A. Ibrayev, V. V. Kalmykov, and K. V. Ushakov, "World Ocean, climate change and supercomputer modeling," in *Supercomputer technologies in science, education and industry*, (in Russian). Moscow: Publishing House of Moscow University, pp. 49-54, 2010.
5. R. A. Ibrayev, V. V. Kalmykov, K. V. Ushakov, and R. N. Khabeev, "Eddy-resolving 1/10 ° World Ocean model," (in Russian), *Ecological safety of coastal and shelf areas and comprehensive utilization of resources*, vol. 25, no. 2, pp. 30-44, 2011.
6. R. A. Ibrayev, *Mathematical Modeling of Thermo hydrodynamic Processes in the Caspian Sea*. Moscow: GEOS, 2008.
7. J. C. Pau and B. F. Sanders, "Performance of parallel implementations of an explicit finite-volume shallow-water model," *ASCE J. Computing in Civil Engineering*, vol. 20, no. 2, pp. 99-110, 2006.
8. Y. M. Tsai, H.C. Kuo, and Y. C. Chang, "A new parallel domain-decomposed Chebyshev collocation method for atmospheric and oceanic modeling," *Terrestrial, Atmospheric and Oceanic Sciences*, vol. 23, pp. 439-450, 2012.
9. J. G. Chefter, C. K. Chu, and D. E. Keyes, "Domain decomposition for the shallow water equations," in *Contemporary Mathematics: Proc. 7th Int. Conf. Domain Decomposition Methods in Sci. and Eng.* pp. 485-490, 1994.
10. P. D. Killworth, D. Stainforth, D. J. Webb, and S. Paterson, "The development of a free surface Bryan–Cox–Semtner model," *J. Phys. Oceanogr.*, vol. 21, pp. 1333-1348, 1991.

11. M. A. Tolstykh, R. A. Ibrayev, E. M. Volodin, K. V. Ushakov, V. V. Kalmykov, A. V. Shlyayeva, V. G. Mizyak, and R. N. Khabeev, *Global Atmosphere and World Ocean Models: Algorithms and Supercomputing Technologies*, manual (Series "Supercomputer Education"). Moscow: Moscow University Publishing House, 2013.
12. S. Valcke, V. Balaji, A. Craig, C. DeLuca, R. Dunlap, R. W. Ford, R. Jacob, J. Larson, R. O'Kuinghttons, G. D. Riley, and M. Vertenstein, "Coupling technologies for Earth System Modelling," *Geosci. Model Dev.*, vol. 5, pp. 1589-1596, 2012.

**About authors:**

**KALMYKOV, Vladimir Vladimirovich**, Postgrad. Student, Dept. Computational Mathematics and Cybernetics of Lomonosov Moscow State University. Master thesis: "Development of a parallel hydrodynamic World Ocean model". Prep. ph.d thesis: "Framework for ocean-ice-atmosphere-land coupled model on massively-parallel architectures".

**IBRAYEV, Rashit Akhmetzievich**, Principal Scientific Researcher, Institute of Numerical Mathematics Russian Academy of Sciences (INM RAS). Dr. Sc. in Physics and Mathematics, thesis "Mathematical Modeling of Thermodynamics of the Caspian Sea" (INM RAS, 2002). Corresponding Member of the Russian Academy of Sciences, 2008.