ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 101 за 2014 г.

Андреев С.В., Бондарев А.Е.,
Михайлова Т.Н., Рыжова И.Г.

Реализация параллельного алгоритма решения задач оптимизационного анализа

Рекомендуемая форма библиографической ссылки: Реализация параллельного алгоритма решения задач оптимизационного анализа / С.В.Андреев [и др.] // Препринты ИПМ им. М.В.Келдыша. 2014. № 101. 11 с. URL: http://library.keldysh.ru/preprint.asp?id=2014-101

О р д е н а   Л е н и н а
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Р о с с и й с к о й   а к а д е м и и   н а у к

С.В.Андреев, А.Е.Бондарев, Т.Н.Михайлова, И.Г.Рыжова

Реализация параллельного алгоритма решения задач оптимизационного анализа

Москва — 2014

Андреев С.В., Бондарев А.Е., Михайлова Т.Н., Рыжова И.Г.

Реализация параллельного алгоритма решения задач оптимизационного анализа

В данной работе рассматривается разработка подхода к построению параллельного алгоритма решения задачи оптимизационного анализа. Процесс построения рассматривается с точки зрения таких критериев, как эффективность, простота реализации и инвариантность от разнообразных вариаций алгоритма. Описано построение интерфейса, позволяющего организацию многозадачного расчета в параллельном режиме с использованием технологии MPI. Приведены результаты тестовых расчетов.

Ключевые слова: оптимизационный анализ, параллельные вычисления, тестирование

Sergey Valerievich Andreev, Alexandr Evgenievich Bondarev, Tatiana Nikolaevna Mikhaylova, Irina Gennadievna Ryzhova

Design of parallel algorithm for optimizing analysis problems

The paper considers a development of approach to construction of parallel algorithm for optimizing analysis problems. The process of parallelizing is to be effective, simple and independent on variations of initial serial algorithm. The paper describes the construction of parallel interface providing multitask parallel calculations by means of MPI technology. The results of testing are described.

Key words: optimizing analysis, parallel calculations

Работа выполнена при поддержке Российского фонда фундаментальных исследований, проекты 13-01-00367-а и 14-01-00769-а.

Введение

Возможности параллельных вычислений и интенсивное развитие параллельных алгоритмов, методов и программных средств неуклонно ведут к изменению существующих приоритетов в задачах математического моделирования. До сих пор большинство современных постановок проблем численного моделирования относится к классу прямых задач, в которых решение вычисляется при полностью заданных граничных и начальных условиях. Согласно [1], для решения конкретных практических научно-технических проблем наиболее важно решать обратные задачи, включающие формальные параметры, которые надо оптимизировать по условию минимизации некоторого целевого функционала при известных дополнительных ограничениях на входные данные [2]. Решение обратной задачи методами оптимизации производится с помощью направленного перебора прямых задач, количество которых может измеряться сотнями и тысячами.

Следует заметить, что обратная задача – это всего лишь одна конкретная задача с фиксированными определяющими параметрами. Современные параллельные вычисления позволяют находить решение для класса задач, или, другими словами, решать задачу оптимизационного анализа [3]. Класс задач задается набором определяющих параметров задачи, включающих геометрические параметры, характерные параметры (характеризующие физическую модель), вычислительные параметры и др. В задачах вычислительной механики жидкости и газа характерными параметрами задачи (параметрами подобия) являются такие величины, как числа Маха, Эйлера, Фруда, Рейнольдса, Струхаля. При решении прямой задачи эти параметры, как правило, фиксированы. Также они фиксированы при решении обратной задачи. Решить задачу оптимизационного анализа – это значит решить обратную задачу во всех диапазонах возможного изменения определяющих параметров. Только тогда можно говорить о полученном решении для класса задач. Очевидно, что при этом вычислительные трудности и затраты вырастают на порядки. Однако именно решение подобных задач является одной из перспективных целей применения параллельных вычислений. В качестве примеров можно привести работы [3-5], посвященные различным аспектам решения задач оптимизационного анализа, в том числе, методам, алгоритмам и средствам интерпретации полученных результатов.

Задачи оптимизационного анализа удобны для организации параллельных расчетов, так как предполагают массовое решение однотипных задач с разными входными параметрами. При этом становятся актуальными вопросы построения простых и эффективных параллельных алгоритмов решения таких задач.

Данная работа ставит своей целью описание подхода к построению параллельного алгоритма оптимизационного анализа. Процесс распараллеливания рассматривается с точки зрения таких критериев, как эффективность, простота реализации и инвариантность от алгоритма, что в итоге должно обеспечивать получение надежного результата. На основании рассмотренных критериев строится интерфейс, позволяющий организацию многозадачного расчета в параллельном режиме с использованием технологии MPI. Приведены результаты тестовых расчетов.

Общая постановка задачи оптимизационного анализа и алгоритма решения

Приведем постановку задачи оптимизационного анализа в наиболее общем виде. Согласно [5], формальную общую постановку задачи оптимизационного анализа можно изложить следующим образом:

Предположим, что имеются математическая модель нестационарного процесса и надежный численный метод для решения этой модели. В этом случае мы можем решать прямую задачу численного моделирования нестационарного процесса. Допустим, что в моделируемом процессе происходит некое событие (явление, эффект). Численное решение $\boldsymbol{F}=\boldsymbol{F}(x,x_1,...,x_n)$ выбранной задачи формируется в процессе математического моделирования и определяется управляющим параметром x и конечным набором определяющих параметров задачи $(x_1,...,x_n)$, где каждый определяющий параметр x_i имеет свой диапазон изменения $x_i\in[x_i^*,x_i^{**}]$.

Обозначим $\bar{X}=(x,x_1,...,x_n)$ и введем *функционал события* $\Phi(F(\bar{X}))$, который на решении задачи принимает, подобно логической переменной, два значения: 1 – если событие, интересующее исследователя, наступило (независимо от рода события), и 0 – если событие не наступило.

$\Phi(F(\bar{X}))=0$ - событие не наступило (1)

$\Phi(F(\bar{X}))=1$ - событие наступило.

Пусть x' – значение управляющего параметра, при котором наступает изучаемое явление. Тогда общую постановку задачи можно сформулировать следующим образом:

- найти $\min\limits_{\Delta x} I(\Delta x)$ для всех значений определяющих параметров внутри диапазонов их изменения, т.е. $\forall x_i\in[x_i^*,x_i^{**}]$, где $I(\Delta x)$ – функционал следующего вида:

$$I(\Delta x)=1-\Phi(F(\bar{X})),\quad \Delta x=x-x'. \tag{2}$$

Таким образом, наша задача формально состоит в минимизации функционала $I(\Delta x)$ при помощи вариации управляющего параметра. А в реальности, варьируя Δx, мы должны с приемлемой точностью отыскать значение x', то есть то значение управляющего параметра, при котором событие наступает.

Решая одну обратную задачу, мы получаем одно значение x' для управляющего параметра при фиксированных определяющих параметрах. Это решение одной конкретной задачи. Однако нашей целью является получение решения для целого класса задач. Класс задач определяется диапазонами изменения определяющих (характерных) параметров задачи. Итоговая цель исследования состоит в том, чтобы построить зависимость $x'(x_1,...,x_n)$ для всех возможных значений определяющих параметров. Таким образом, если мы имеем в диапазоне разбиения каждого определяющего параметра M точек, то для того чтобы найти значения x' управляющего параметра во всей многомерной области определяющих параметров, необходимо решить M^n однотипных обратных задач. В результате решения этого набора задач находятся все точки в исследуемом пространстве определяющих параметров, где происходит событие.

Для того чтобы выразить общую постановку задачи оптимизационного анализа более строго с математической точки зрения, необходимо применить понятие многомерных пространств. Рассматривая $(x_1,...,x_n)$ как набор базисных векторов, можно представить пространство определяющих параметров $L(x_1,...,x_n)$, имеющее размерность n. Тогда в общем случае задачу оптимизационного анализа можно сформулировать как нахождение в пространстве L всех подобластей L^*, где наблюдается изучаемое событие, т.е. $\Phi(L^*)=1$.

Данная постановка задачи одновременно предполагает фильтрацию тех точек пространства определяющих параметров, где ожидаемое событие не наступает. Если для конкретной точки $(x_1,...,x_n)$ пространства определяющих параметров для любых значений управляющего параметра x событие не наступает, данная конкретная точка изымается из рассмотрения.

Исходя из того факта, что для решения общей задачи оптимизационного анализа, согласно [5], необходима организация массового численного решения однотипных обратных задач, различающихся лишь входными параметрами, строится алгоритм решения. На рис.1 представлена схема алгоритма решения задачи оптимизационного анализа.

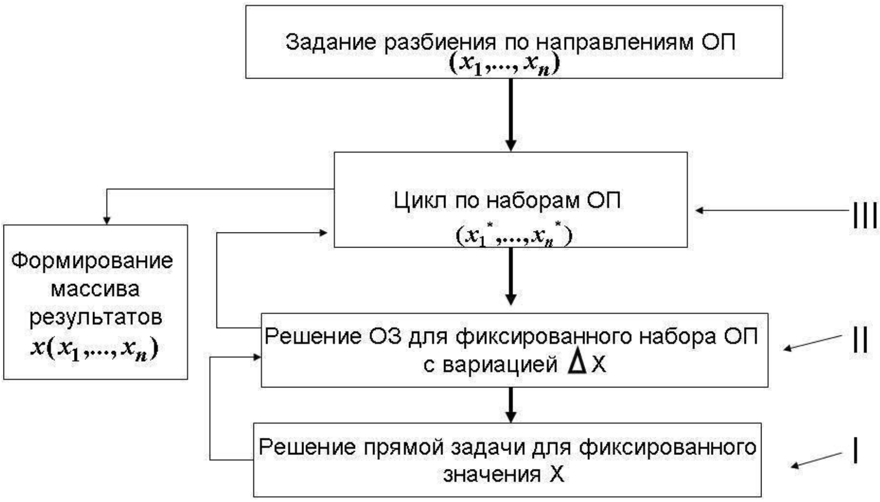

Рис.1. Схема алгоритма решения задачи оптимизационного анализа.

Данная схема построена на предположении, что решение обратной задачи для фиксированного набора определяющих параметров уже имеется. Теперь мы можем организовать массовое решение обратных задач для всех наборов определяющих параметров в диапазоне их изменения. Вначале задается сеточное разбиение по каждому определяющему параметру, рассматриваемому как соответствующее координатное направление в многомерном пространстве. В итоге получаем соответствующие наборы определяющих параметров, рассматриваемые как точки в многомерном пространстве. Далее организуется цикл по всем точкам, где в каждой точке решается обратная задача. По результатам работы данного цикла формируется итоговый многомерный массив результатов, содержащий искомую функциональную зависимость управляющего параметра от определяющих параметров класса задач, представленную в дискретном виде.

Построение параллельного алгоритма решения задачи оптимизационного анализа

Основная задача заключается в наиболее оптимальном распараллеливании алгоритма, представленного на рис.1. На сегодняшний день существует огромное количество методов и подходов к организации параллельных вычислений [6]. Однако задача выбора способа распараллеливания, обеспечивающего простое, быстрое и эффективное получение результатов, остается актуальной.

Для осуществления такого выбора необходимо руководствоваться набором критериев, призванных обеспечивать получение надежного результата. К числу необходимых критериев, в первую очередь, следует отнести эффективность. Это подразумевает, что полученный в результате параллельный алгоритм должен делать в точности то же самое, что и последовательный, на порядки быстрее. Другим не менее важным критерием является простота построения параллельного алгоритма. Третьим важнейшим критерием является инвариантность, т.е. независимость метода распараллеливания от конкретного распараллеливаемого алгоритма.

Рассмотрим общую схему последовательного варианта алгоритма решения задачи оптимизационного анализа, выделив наиболее интересные узлы для распараллеливания данного алгоритма римскими цифрами (рис.1). На предварительном этапе задается сеточное разбиение пространства определяющих параметров (ОП), формируя всевозможные фиксированные наборы ОП $(x_1^*,...,x_n^*)$. Далее в цикле по всем заданным наборам $(x_1^*,...,x_n^*)$ для каждого набора проводится решение обратной задачи (ОЗ). Обратная задача для каждого набора решается путем вариации управляющего параметра x, вплоть до нахождения с заданной точностью значения x', т.е. наступления искомого события. В процессе вариации управляющего параметра x на каждом шаге решается прямая задача моделирования при заданном значении x. В результате работы алгоритма формируется многомерный массив результатов, представляющий собой дискретную зависимость $x'(x_1,...,x_n)$. Далее к массиву результатов могут применяться методы обработки многомерных данных.

Данный алгоритм в целом предполагает решение очень большого количества обратных задач численного моделирования M^n (при задании M точек в диапазоне разбиения каждого определяющего параметра), каждая из которых предполагает, в свою очередь, решение большого количества прямых задач. В данной ситуации необходимо применить параллельные вычисления.

Рассмотрим наиболее интересные узлы для распараллеливания данного алгоритма, выделенные на рис. 1 римскими цифрами. Будем оценивать пригодность данных узлов к распараллеливанию, прежде всего, с точки зрения инвариантности или независимости от конкретного алгоритма, реализуемого в данном узле.

В первую очередь, рассмотрим алгоритм решения прямой задачи математического моделирования (узел I) при заданном значении x. Данный узел полностью зависит от конкретного алгоритма решения прямой задачи, а этот алгоритм может быть малопригоден для распараллеливания. Второй узел – это организация решения обратной задачи для фиксированного набора ОП (узел II). Он сводится к поиску значения x' с заданной точностью. Здесь решается оптимизационная задача и, аналогично предыдущему узлу, имеется зависимость от выбора конкретного алгоритма решения оптимизационной задачи.

Зато третий узел обладает естественной полной инвариантностью, и здесь вне зависимости от алгоритмов возможна организация параллельных вычислений однотипных обратных задач (ОЗ) с разными входными данными, представляющими собой фиксированные наборы определяющих параметров (ОП) по принципу «один вариант ОП – один процессор».

При реализации на многопроцессорной вычислительной системе общая схема параллельного варианта решения задачи оптимизационного анализа сводится к заданию разбиения по всем определяющим параметрам, формированию таким образом входных данных для однотипных обратных задач, заданию числа вычислительных узлов и раздаче заданий каждому узлу со своими входными параметрами. По завершении работы всех процессоров проводится сбор данных и формирование массива результатов для последующей обработки.

В силу того, что процессы решения однотипных обратных задач происходят фактически без обменов информацией между процессорами, распараллеливание здесь сводится к организации интерфейса, управляющего распределением вариантов по процессорам и сбором данных в единый массив результатов. Данный вариант является наиболее легким в программной реализации и позволяет ускорить расчет во столько раз, сколько процессоров может быть выделено одновременно.

Данный вариант распараллеливания алгоритма решения задачи оптимизационного анализа является полностью инвариантным относительно внутренних алгоритмов, так как алгоритмы решения прямой и обратной задач не затрагиваются при распараллеливании. Предложенный вариант может быть наиболее легко и просто реализован с помощью технологии MPI (Message-Passing Interface) [7].

Практические результаты

Данный подход был применен к конкретной практической задаче. Рассматриваемая здесь задача оптимизационного анализа основана на массовом решении обратных задач при задании различных наборов определяющих параметров. В свою очередь, решение каждой обратной задачи основано на многократном решении прямой задачи математического моделирования о течении вязких теплопроводных сред в каналах. В качестве обратной задачи рассматривалась задача нахождения оптимального значения весового коэффициента применяемой гибридной разностной схемы. В качестве варьируемых определяющих (характерных) параметров выбирались характерные числа Маха и Прандтля. В качестве математической модели прямой задачи использовалась полная система нестационарных уравнений Навье-Стокса для движения сплошной среды с учетом эффектов вязкости, теплопроводности и сжимаемости.

При проведении расчетов конкретной практической задачи применялась гибридная неявная конечно-разностная схема, представленная в [4]. Данная схема на каждом направлении обладает вторым порядком аппроксимации по времени и пространству с добавочным членом, выполняющим функции искусственной вязкости.

Необходимо было найти минимальный весовой коэффициент разностной схемы, не допускающий возникновения нежелательных осцилляций.

Реализация обратной связи при решении обратной задачи проводилась с помощью анализатора решения, основанного на анализе и контроле количества локальных экстремумов в полученном численном решении.

В качестве теста в режиме последовательных вычислений проводились расчеты задачи оптимизационного анализа в варианте с двумя выбранными определяющими параметрами: характерным числом Маха прямой задачи $\boldsymbol{M_\infty}$ и характерным числом Прандтля $\mathbf{Pr}$. В качестве диапазонов изменения были выбраны $\mathbf{2.0 < M_\infty < 3.0}$ и $\mathbf{0.72 < Pr < 1.0}$. В качестве сеточного разбиения для числа Маха были взяты 5 точек, а для числа Прандтля – 4 точки. Таким образом, тестовый вариант потребовал проведения решения 20 обратных задач и значительных временных затрат. Для реализации аналогичных расчетов в параллельном режиме в рамках настоящей работы были использованы вычислительные возможности многопроцессорного гибридного вычислительного кластера К-100, разработанного в Институте прикладной математики им. М.В. Келдыша РАН.

Проведенные в параллельном режиме расчеты тестов позволили получить искомый результат в виде четких ограничений для весового коэффициента разностной схемы для широкого диапазона чисел $\boldsymbol{M_\infty}$ и $\mathbf{Pr}$ в таком классе задач, как вязкие теплопроводные течения в канале. В результате проведения расчетов был получен итоговый численный массив, представляющий зависимость управляющего параметра от определяющих параметров рассматриваемой задачи $\boldsymbol{S_k(M_\infty, \mathrm{Pr})}$ в дискретной форме. Данный массив представлен на рис.2 в виде трехмерной поверхности.

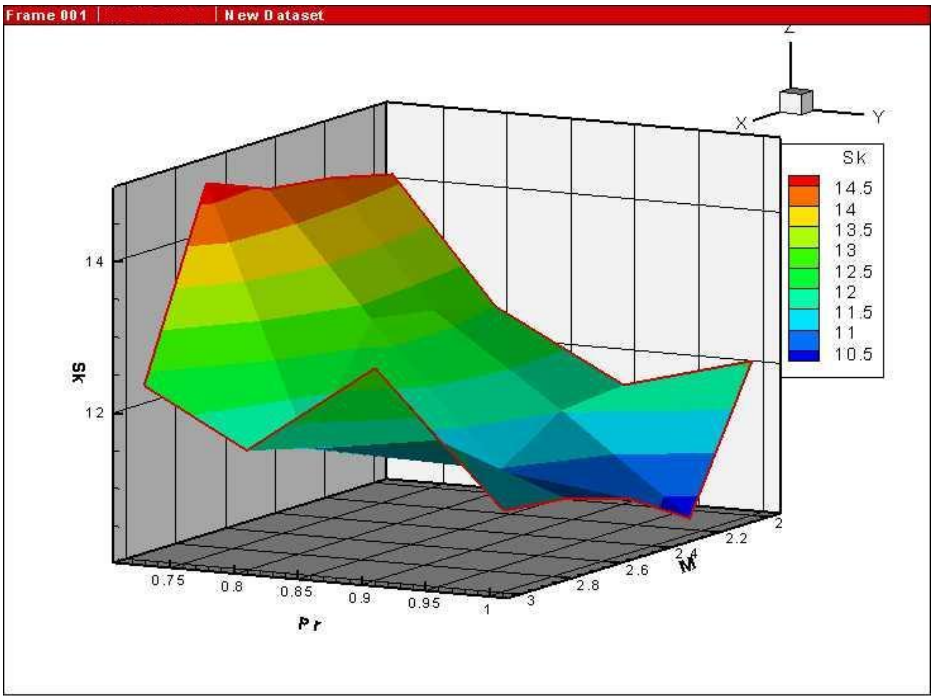

Рис.2. Массив результатов в виде трехмерной поверхности.

Применение параллельного подхода к построению алгоритма решения задачи оптимизационного анализа с помощью технологии MPI позволило получить существенный выигрыш с точки зрения временных затрат на проведение вычислений. Так, последовательный расчет на персональном компьютере занял более 19 часов, расчет на кластере К-100 в последовательном режиме занял 47 минут 17 секунд, а параллельный расчет на том же кластере занял 2 минуты 22 секунды.

По результатам проведенных вычислений можно утверждать, что в результате работы построен прототип параллельного алгоритма, удовлетворяющий перечисленным выше критериям результативности. Данный алгоритм эффективен, так как позволяет получать искомое численное решение. Описанный подход является простейшим вариантом распараллеливания, доступным для массового пользователя. Предложенный подход является полностью инвариантным относительно как алгоритма и численного метода решения прямой задачи, так и алгоритма оптимизации решения обратной задачи. Удовлетворяя всем критериям обеспечения надежности результата, данный простой прототип может служить основой для организации массовых расчетов задач подобного типа.

Заключение

В данной работе описана выработка подхода к оптимальному распараллеливанию алгоритма решения задачи оптимизационного анализа. Процесс распараллеливания рассмотрен с точки зрения таких критериев, как эффективность, простота реализации и инвариантность от разнообразных вариаций алгоритма. На основании рассмотренных критериев построен интерфейс, позволяющий организацию многозадачного расчета в параллельном режиме с использованием технологии MPI. Приведены результаты тестовых расчетов.

Список литературы

[1] Ильин В.П. Стратегии и тактики экстремального параллелизма // Наука в Сибири. 2012. № 6. URL: http://www.sbras.ru/HBC/article.phtml?nid=622&id=11

[2] Алифанов О.М. Обратные задачи в теплопередаче. М.: Машиностроение. 1988.

[3] Бондарев А.Е. Оптимизационный анализ нестационарных пространственно-временных структур с применением методов визуализации // Научная визуализация. 2011. Т.3. №2. С.1-11.

[4] Разработка программного комплекса BURGERS2 для оптимизации и визуализации вычислительных свойств гибридных разностных схем / Бондарев А.Е. [и др.] // Научная визуализация. 2013. Т.5. №1. С. 26-37.

[5] Bondarev A.E, Galaktionov V.A. Parametric Optimizing Analysis of Unsteady Structures and Visualization of Multidimensional Data // International Journal of Modeling, Simulation and Scientific Computing. 2013. V.04 N supp01. 13 p. DOI 10.1142/S1793962313410043.

[6] Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург. 2002. 608 с.

[7] Антонов А.С. Параллельное программирование с использованием технологии MPI. М.: Изд.МГУ. 2004. 72 с.

Оглавление