УДК 519.6

© 2013 г. **В.А. Рукавишников**, д-р физ.-мат. наук,
С.Г. Николаев,
А.С. Сарыков
(Вычислительный центр ДВО РАН, Хабаровск)

ПРОГРАММА ДЛЯ ПАКЕТНОГО МОДЕЛИРОВАНИЯ СИНГУЛЯРНЫХ ЗАДАЧ НА ВЫСОКОПРОИЗВОДИТЕЛЬНОМ КЛАСТЕРЕ*

Решается задача автоматизации процесса проведения серий численных экспериментов для задач теории катастроф с сингулярностью в решении на высокопроизводительном вычислительном кластере. Предлагается использовать с этой целью встроенные средства операционной системы, обеспечивая максимальную переносимость.

Ключевые слова: задачи теории катастроф с сингулярностью, автоматизация расчетов, вычислительный кластер.

Введение

Математические модели различных естественных процессов описываются дифференциальными уравнениями и системами уравнений в частных производных в замкнутых областях. Особенности геометрии областей (наличие трещин, углов и изломов у границы области), поведение исходных данных (наличие сред с различными характеристиками) и «внутренние» свойства этих моделей приводят к сингулярности в решениях. Факторы, вызывающие сингулярность решения, служат, как правило, причинами катастроф, а точки, в которых функция решения имеет сингулярность, являются критическими. Это обстоятельство потребовало от нас создания специальных численных методов, учитывающих особенности задачи, – например, [1 – 7].

При создании алгоритмов и комплексов программ для численного анализа на многопроцессорном компьютере математических моделей, позволяющих определять условия происхождения катастроф в электродинамических, гидродинамических процессах и в строительных конструкциях приходится учитывать наличие большого числа параметров, из которых необходимо выбрать близкие к оптимальным значениям для обеспечения наилучшей точности нахождения решения. Это обусловливает потребность в проведении серии численных эксперимен-

* Работа выполнена при финансовой поддержке РФФИ (проекты 10-01-00060, 11-01-98502-р_восток_а) и Президиума ДВО РАН (гранты 12-I-П18-01, 12-III-А-01М-001).

тов, а следовательно, требует иного подхода по сравнению с однократным запуском задач.

Доступ к разделяемым ресурсам кластера накладывает дополнительные обязанности на оператора, осуществляющего постановку вычислительных программ в очереди для расчета. Для эффективного их использования при пакетном моделировании требуется оптимально организовать процесс отправки на кластер отдельных задач в серии. В частности, необходимо иметь возможность своевременно занимать ресурсы, освобожденные другими пользователями, чтобы сократить общее время проведения эксперимента. Работа по обеспечению этого функционала ложится на исследователя, так как системы диспетчеризации не предоставляют запущенному процессу возможности воспользоваться дополнительными узлами, если они не были выделены ему в момент запуска. Кроме того, при множественном запуске усложняется процесс контроля за выполняющимися процессами, затрудняется сбор результатов.

Целью создания программы пакетного моделирования сингулярных задач на высокопроизводительном кластере является повышение удобства и эффективности процесса тестирования новых методов за счет ограничения действия оператора, лишь созданием необходимых файлов с входными параметрами и однократным запуском управляющего модуля; своевременного автоматического «захвата» освободившихся ресурсов кластера; организации удобного контроля и сбора результатов.

Математическая модель и весовой метод конечных элементов

В качестве математической модели с сингулярностью приведем задачу теории упругости в области с границей, содержащей входящий тупой угол.

Через R^2 обозначим двумерное евклидово пространство, $x=(x_1,x_2)$ – произвольный элемент этого пространства, $\|x\|^2=x_1^2+x_2^2$. Пусть $\Omega\subset R^2$ – ограниченная невыпуклая многоугольная область с границей Г, содержащей тупой угол γ с вершиной в начале координат $(0,0)$, $\overline{\Omega}=\Omega\cup\Gamma$.

Обозначим через $\Omega'=\left\{x\in\overline{\Omega}:(x_1^2+x_2^2)^{1/2}\le\delta<1\right\}$ часть δ-окрестности точки $(0,0)$, лежащую в области $\overline{\Omega}$. Введем весовую функцию $\rho(x)$, совпадающую в Ω' с расстоянием до начала координат, т.е. $\rho(x)=(x_1^2+x_2^2)^{1/2}$ при $x\in\Omega'$ и равную δ для $x\in\overline{\Omega}\setminus\Omega'$.

Пусть $\mathbf{u}=(u_1,u_2)$ – вектор-функция перемещений. Предполагая, что область $\overline{\Omega}$ представляет собой однородное изотропное тело, а деформации малы, получим следующие выражения, связывающие перемещения, деформации и напряжения:

$$\varepsilon_{ij}(\mathbf{u})=\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right),\quad |\varepsilon|=\sum_{i,j=1}^{2}\varepsilon_{ij}^2=\varepsilon:\varepsilon,\quad \mathrm{tr}(\varepsilon)=\sum_{i=1}^{2}\varepsilon_{ii};$$

$$\sigma(\mathbf{u}) = \lambda \mathrm{tr}(\varepsilon(\mathbf{u})) + 2\mu\varepsilon(\mathbf{u}) = \begin{pmatrix} (\lambda+2\mu)\dfrac{\partial u_1}{\partial x_1} + \lambda\dfrac{\partial u_2}{\partial x_2} & \mu\left(\dfrac{\partial u_1}{\partial x_2} + \dfrac{\partial u_2}{\partial x_1}\right) \\ \mu\left(\dfrac{\partial u_2}{\partial x_1} + \dfrac{\partial u_1}{\partial x_2}\right) & \lambda\dfrac{\partial u_1}{\partial x_1} + (\lambda+2\mu)\dfrac{\partial u_2}{\partial x_2} \end{pmatrix}.$$

В области Ω рассмотрим краевую задачу для системы уравнений Ламе с постоянными коэффициентами λ и μ относительно поля перемещений $\mathbf{u}$:

$$-(2\mathbf{div}(\mu\varepsilon(\mathbf{u})) + \nabla(\lambda\,\mathrm{div}\,\mathbf{u})) = \mathbf{f}, \quad x \in \Omega, \tag{1}$$

$$u_i = q_i, \quad x \in \Gamma_i^D, \tag{2}$$

$$\sigma_i(\mathbf{u})\mathbf{n} = p_i, \quad x \in \Gamma_i^N, \quad i = 1,2. \tag{3}$$

В (2) и (3) на участках границы Γ_i^D и Γ_i^N для i-й компоненты вектор-функции $\mathbf{u}$ заданы значения перемещения и нагрузки соответственно $\Gamma = \Gamma_i^D \cup \Gamma_i^N$, $\Gamma_i^D \cap \Gamma_i^N = \varnothing$, $i = 1,2$; $\mathbf{n}$ – единичный вектор внешней нормали. Через $\mathbf{f}$ и $\mathbf{p} = (p_1, p_2)$ обозначены векторные поля массовых сил и напряжений соответственно. Кроме того, предположим, что правые части (1) – (3) удовлетворяют условиям:

$$\mathbf{f} \in \mathbf{L}_{2,\beta}(\Omega,\delta), \quad q_i \in W_{2,\beta}^{1/2}(\Gamma_i^D,\delta), \quad p_i \in L_{2,\beta-1/2}(\Gamma_i^N,\delta), \tag{4}$$

где $i = 1,2$, β, $\delta > 0$ (обозначения пространств см., напр., [8]).

Введем билинейную и линейную формы

$$a(\mathbf{u},\mathbf{v}) = \int_\Omega (2\mu\varepsilon(\mathbf{u}) : \varepsilon(\rho^{2\nu}\mathbf{v}) + \lambda \mathrm{div}\mathbf{u}\mathrm{div}(\rho^{2\nu}\mathbf{v}))dx,$$

$$l(\mathbf{v}) = \int_\Omega \rho^{2\nu}\mathbf{f}\mathbf{v}dx + \int_\Gamma \rho^{2\nu}\mathbf{p}\mathbf{v}ds.$$

Определение 1. Функцию $\mathbf{u}_\nu(x)$ из множества $\mathbf{W}_{2,\nu}^1(\Omega,\delta)$ будем называть R_ν-обобщенным решением задачи (1) – (4), если почти всюду на Γ_i^D $(i = 1,2)$ она удовлетворяет краевому условию (2) и для любых $\mathbf{v}(x)$ из $\mathbf{W}_{2,\nu}^{1,D}(\Omega,\delta)$ и $\nu \ge \beta$ выполнено интегральное тождество

$$a(\mathbf{u}_\nu, \mathbf{v}) = l(\mathbf{v}). \tag{5}$$

Для нахождения R_ν-обобщенного решения построим схему метода конечных элементов. Для этого произведем квазиравномерную триангуляцию T^h области $\overline{\Omega}$ и введем специальные базисные функции.

Разбиваем $\overline{\Omega}$ на конечное число треугольников K, называемых конечными элементами, с вершинами P_k $(k = \overline{0, n-1})$ – узлами триангуляции. Обозначим через $\Omega^h = \bigcup_{K \in T^h} K$ объединение всех элементов; h – максимальная из длин их сторон.

Потребуем, чтобы: 1) два различных треугольника либо не пересекались, либо имели общими только вершину или смежную сторону; 2) площади элементов были величинами одного порядка; 3) вершины, принадлежащие границе триангулированной области $\partial\Omega^h$, принадлежали также и Γ; 4) минимальный из углов элементов был строго положительный и не зависел от триангуляции.

Во множестве $\mathrm{P}=\{P_k\}_{k=0}^{n-1}$ всех узлов триангуляции выделим подмножество P_i^D $(i=1,2)$ граничных узлов, в которых для i-й компоненты задано краевое условие первого рода.

Каждому узлу $P_k \in \mathrm{P}$ ставим в соответствие функцию ψ^k вида

$$\psi^k(x)=\rho^{\nu^*}(x)\varphi^k(x), \quad k=\overline{0,n-1},$$

где $\varphi^k(x)$ линейна на каждом конечном элементе и $\varphi^k(P_j)=\delta_{kj}$; $k,j=\overline{0,n-1}$; δ_{kj} – символ Кронекера; ν^* – вещественное число.

Определим множество V^h как линейную оболочку, натянутую на систему базисных функций $\{\psi^k\}_{k=0}^{n-1}$. Выделим в V^h подмножества V_i^h $(i=1,2)$ функций, обращающихся в ноль в узлах, принадлежащих Γ_i^D, т.е.:

$$V_i^h=\left\{v_i \in V^h : v_i(P_k)\Big|_{P_k\in P_i^D}=0\right\}, \ i=1,2.$$

Соответствующие векторные множества, используемые в дальнейшем, обозначим через $\mathbf{V}^h=[V^h]^2$ и $\mathbf{V}_D^h=V_1^h\times V_2^h$.

Конечно-элементная аппроксимация компонент вектора перемещений, связанная с построенной триангуляцией, имеет вид

$$u_{\nu,1}^h=\sum_{k=0}^{n-1} d_{2k}\psi^k(x), \quad u_{\nu,2}^h=\sum_{k=0}^{n-1} d_{2k+1}\psi^k(x), \quad d_j=\rho^{-\nu}(P_{[j/2]})c_j, \quad j=\overline{0,2n-1}.$$

Определение 2. Приближенным R_ν-обобщенным решением краевой задачи (1)–(4) по весовому методу конечных элементов будем называть такую функцию $\mathbf{u}_\nu^h(x)\in\mathbf{V}^h$, что в узлах множества P_i^D ее i-е компоненты удовлетворяют краевому условию (2) и для произвольных $\mathbf{v}_D^h(x)\in\mathbf{V}_D^h$ и $\nu\ge\beta$ выполнено интегральное тождество

$$a(\mathbf{u}_\nu^h,\mathbf{v}_D^h)=l(\mathbf{v}_D^h), \tag{6}$$

где $\mathbf{u}_\nu^h=(u_{\nu,1}^h,u_{\nu,2}^h)$.

Интегральное тождество (6) приводит к системе линейных алгебраических уравнений (СЛАУ), элементы основной матрицы которой формируются из локальных матриц жесткости элемента, а правая часть – из локальных векторов нагрузки стандартным образом (см., напр., [9]).

В общем виде матрицу жесткости и вектор нагрузки элемента можно записать следующим образом:

$$\begin{pmatrix} a_{ij} & a_{ij+1} \\ a_{i+1j} & a_{i+1j+1}\end{pmatrix}\begin{pmatrix} u_1^k \\ u_2^k\end{pmatrix}=\begin{pmatrix} b_i \\ b_{i+1}\end{pmatrix}, \quad \begin{cases} i=2k, \\ j=2l,\end{cases} \quad k,l=0,1,2.$$

$$a_{ij}=\int_K\left((\lambda+2\mu)\frac{\partial\psi^l}{\partial x_1}\frac{\partial(\rho^{2\nu}\psi_1^k)}{\partial x_1}+\mu\frac{\partial\psi^l}{\partial x_2}\frac{\partial(\rho^{2\nu}\psi_1^k)}{\partial x_2}\right)dx,$$

$$a_{ij+1} = \int\limits_K \left(\lambda \frac{\partial \psi^l}{\partial x_2} \frac{\partial(\rho^{2\nu}\psi_1^k)}{\partial x_1} + \mu \frac{\partial \psi^l}{\partial x_1} \frac{\partial(\rho^{2\nu}\psi_1^k)}{\partial x_2} \right) dx,$$

$$b_i = \int\limits_K \rho^{2\nu} f_1 \psi_1 \, dx + \int\limits_{\gamma_1^N} \rho^{2\nu} p_1 \psi_1^k \, ds,$$

$$a_{i+1j} = \int\limits_K \left(\lambda \frac{\partial \psi^l}{\partial x_1} \frac{\partial(\rho^{2\nu}\psi_2^k)}{\partial x_2} + \mu \frac{\partial \psi^l}{\partial x_2} \frac{\partial(\rho^{2\nu}\psi_2^k)}{\partial x_1} \right) dx,$$

$$a_{i+1j+1} = \int\limits_K \left((\lambda + 2\mu) \frac{\partial \psi^l}{\partial x_2} \frac{\partial(\rho^{2\nu}\psi_2^k)}{\partial x_2} + \mu \frac{\partial \psi^l}{\partial x_1} \frac{\partial(\rho^{2\nu}\psi_2^k)}{\partial x_1} \right) dx,$$

$$b_{i+1} = \int\limits_K \rho^{2\nu} f_2 \psi_2^k \, dx + \int\limits_{\gamma_2^N} \rho^{2\nu} p_2 \psi_2^k \, ds.$$

Здесь γ_i^N, $(i = 1,2)$ – объединение сторон элемента, совпадающих с участками границы области, на которых для i-й компоненты заданы краевые условия второго рода; k и l – локальные номера узлов треугольника; ψ^l – сужение на элемент компонент базисных функций, определенных для l-го узла; ψ_1^k, ψ_2^k – сужение на элемент первой и второй компонент пробных функций, определенных в k-м узле.

Описание алгоритма численного анализа представленной модели и результаты расчета ряда модельных задач, проведенных с использованием высокопроизводительного кластера ВЦ ДВО РАН, содержатся в [8]. Для различных величин входящего тупого угла γ и разных комбинаций краевых условий на его сторонах был проведен поиск параметров δ, ν, ν^* и исследована скорость сходимости описанного выше весового метода конечных элементов при измельчении сетки. Процесс поиска включал в себя проведение вычислений для большого количества наборов значений δ, ν, ν^* и h.

В данной работе предлагается описание алгоритма пакетного запуска серий расчетов для задачи, описанной выше, и других задач, методика решения которых предполагает подбор различных управляющих параметров, обеспечивающих лучшие характеристики метода. Описана программа, которая самостоятельно анализирует загруженность кластера и своевременно ставит в очереди задачи с еще необработанными наборами параметров. При этом организована гибкая схема выбора ресурсов, позволяющая использовать вычислительный модуль, оптимизированный под архитектуру конкретного узла. Кроме того, обеспечивается сбор результатов в форме, удобной для последующего анализа и выбора оптимальных параметров.

Особенности реализации

Разработка проводилась с ориентацией на высокопроизводительный кластер ВЦ ДВО РАН, представляющий гетерогенный комплекс с одним управляющим и семнадцатью вычислительными узлами [http://hpc.febras.net]. Узлы различаются прежде всего моделью процессора (Six Core AMD Opteron 8431, Quad

Core Intel® Xeon® E5450 EM64T, Dual Core Intel® Xeon® 5060 EM64T) и объемом оперативной памяти. Кластер ВЦ работает под управлением операционной системы Linux CentOS 5.3 с системой диспетчеризации заданий TORQUE [www.adaptivecomputing.com].

Для проведения расчетов пользователям доступны три очереди, соответствующие узлам с различной архитектурой. Запущенные задания ставятся в очередь и ожидают освобождения ресурсов. Будучи поставленным в одну из очередей, задание не может быть перемещено в другую, поэтому при множественных расчетах требуется заблаговременно занять место во всех подходящих очередях, в которых могут освободиться ресурсы. При этом:

1) если существует ограничение на количество заданий, одновременно ожидающих запуска, то оператору потребуется постоянно следить за освобождающимися узлами, чтобы вовремя добавлять задания в очередь;

2) если такого ограничения нет, то множество заданий, поставленных одним пользователем кластера, значительно увеличит время ожидания остальными, что является нарушением правил пользования общедоступными ресурсами и может повлечь блокировку учетной записи.

Разработанная программа призвана снизить трудозатраты оператора, обходя имеющиеся ограничения и не нарушая правил работы на кластере. Она состоит из двух сценариев: управляющего сценария «запускатель», вызываемого из командной строки, и q-sub сценария, который ставится в очереди высокопроизводительного кластера «запускателем» и осуществляет запуск вычислительных программ. Входные и выходные данные для программы записываются в виде текстовых файлов.

Программное обеспечение предназначено для работы под операционной системой Linux. Для разработки был выбран язык сценариев bash [www.bash.org]. Это связано с тем, что подавляющее большинство вычислительных кластеров работает под управлением различных версий ОС Linux, в которых зачастую оболочка bash установлена по умолчанию. В совокупности с богатыми средствами обработки текстовых файлов, которые есть в Linux, bash является мощным инструментом, позволяющим решить все поставленные задачи. Сценарии, написанные на bash, легко переносить на другие вычислительные комплексы и дорабатывать, так как нет необходимости в компиляции написанного кода. Для того, чтобы сценарии могли работать на других кластерах с системой диспетчеризации Torque, необходимо лишь изменить информацию об именах очередей и количестве процессорного места в каждой из них, полностью переписывать сценарии не потребуется. Кроме того, для языка сценариев bash существует множество руководств и учебников, в том числе на русском языке.

Работа с кластером ВЦ осуществляется через ssh клиент, поэтому программа не имеет графического интерфейса. В его создании нет необходимости, так как работа оператора сводится к подготовке входных данных и запуску управляющего сценария. При этом файлы могут быть подготовлены в любой удобной среде и загружены на кластер перед стартом серии расчетов.

Для работы программы необходимо вызвать из командной строки сценарий «запускатель» и указать файл входных данных, содержащий все необходимые параметры для счета. Этот файл может содержать четыре и более строк, в зависимости от количества описанных в нем очередей кластера:

1) количество очередей;

2) имя очереди, число выделяемых под одну задачу процессоров данной очереди, ключевые параметры, имя исполняемого файла – «вычислителя» (построчно для каждой очереди);

3) файл параметров «вычислителя»;

4) файл с наборами параметров для счета.

На рис. 1 представлена схема работы управляющего сценария «запускатель».

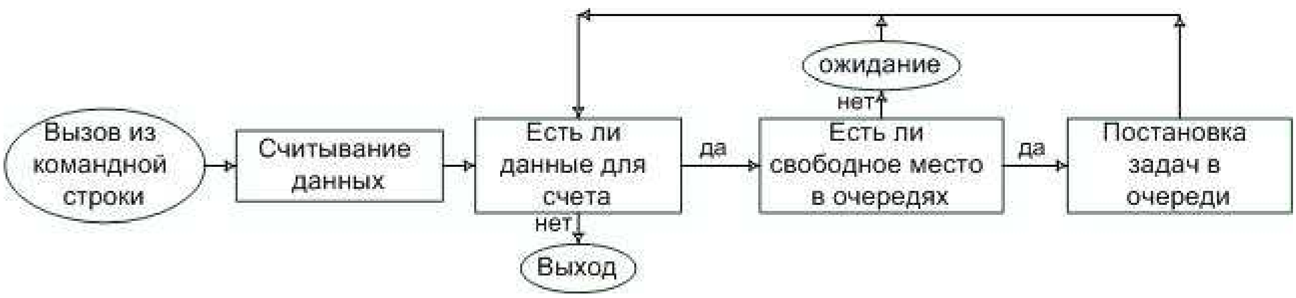

Рис. 1. Схема работы сценария «запускатель».

После вызова из командной строки с именем входного файла в качестве параметра сценарий «запускатель» получает следующие необходимые данные:

имя файла с наборами параметров для счета; на данном этапе важно, чтобы пользователем во входном файле было корректно указано имя файла с наборами параметров для счета, иначе «запускатель» выдаст ошибку, что файл не найден, и завершит свою работу;

количество очередей;

имя файла параметров «вычислителя»; если имя файла не будет указано или будет указано некорректно, то сценарий «запускатель» завершит свою работу.

Проверяя, есть ли данные для счета, сценарий «запускатель» просматривает файл наборов параметров в поиске строк, не находящихся в обработке. Если таких строк нет, то сценарий завершает работу. Иначе формируется список очередей, в которых можно произвести оставшиеся расчеты. Данный этап необходим в связи с тем, что программа ориентирована на работу с «вычислителями», применяющими метод конечных элементов. Это означает, что среди наборов параметров, меняющихся в серии, присутствует и порядок измельчения конечно-элементной сетки. От того, насколько сильно дискретизирована область, напрямую зависит количество памяти, необходимой для расчетов. Выше было указано, что узлы кластера ВЦ, соответствующие разным очередям, имеют различный объем оперативной памяти. В связи с этим некоторые из задач не могут быть отправлены на вычисления в определенные очереди, потому что это приведет к переполнению оперативной памяти узла и задействованию файла подкачки. При этом произойдет значительное падение скорости работы вычислительной программы. Чтобы избежать этого, управляющий сценарий сравнивает значения ключевого параметра для данной очереди из входного файла со значением ключевого параметра в строке набора параметров решения и выбирает тот, который можно обработать в данной очереди, учитывая степень измельчения сетки.

Далее проверяется, есть ли свободное место на узлах кластера, которые могут быть использованы для обсчета еще не обработанных задач. Для этого используется команда: pbsnodes <имя узла очереди>, которая выводит для указанного узла информацию о количестве зарезервированных ядер. Затем, если свободное процессорное место нашлось в одной из очередей, в нее отправляется q-sub сценарий (рис. 2), которому из «запускателя» передаются специфичные для данной очереди параметры и имя исполняемого файла. При этом исполняемый файл «вычислителя» может быть задан отдельно для каждой очереди, что позволяет использовать программы, оптимизированные под архитектуру конкретных узлов.

Рис. 2. Схема работы q-sub сценария.

Q-sub сценарий осуществляет резервирование свободных наборов для счета, запуск их на вычисление и запись результатов счета в выходной файл, общий для всех экземпляров сценария, запущенных на кластере.

После того как сценарий поставлен в очередь, он ищет в файле параметров для счета наборы, не взятые в обработку, и продолжает свою работу до тех пор, пока не закончатся свободные наборы параметров, предназначенные для этой очереди. Таким образом, если задачи, которые возможно запустить в данной очереди кластера, просчитаны, q-sub сценарий завершит свою работу, освободив тем самым процессорное место для других задач.

В том случае, если найден подходящий набор, в соответствующей строке исходного файла ставится метка «о», означающая что строка находится в обработке. Далее зарезервированный набор параметров отправляется для расчетов в «вычислитель», имя которого передано «запускателем». Полученные результаты вычислений записываются построчно в общий для всех q-sub сценариев выходной файл. Если работа «вычислителя» с данным набором завершилась неудачей, то сценарий поставит необходимую метку в файле параметров, чтобы при обработке результатов счета исследователь легко мог определить наборы, для которых произошел сбой.

После окончания работы «запускателя» и всех q-sub сценариев остаются преобразованный исходный файл наборов параметров для счета (перед каждым набором будет стоять метка – найдено решение или нет) и файл с результатами счета, единый для всех q-sub сценариев.

Заключение

Разработано программное обеспечение, которое позволяет автоматически запускать задания, выбирая свободные ресурсы кластера в соответствии с правилами, определенными во входном файле, а параметры для задач – из единого списка; контролировать обработанные и необработанные задания; собирать результаты расчетов; использовать одновременно разные вычислительные модули, оптимизированные под архитектуру узлов конкретной очереди.

Программа для пакетного моделирования сингулярных задач обладает следующими преимуществами:

1) минимизирует участие пользователя в процессе постановки задач в очереди кластера;

2) может быть легко модифицирована для работы на других вычислительных кластерах с системой диспетчеризации TORQUE.

Описанный комплекс был введен в эксплуатацию в Лаборатории математического моделирования в физике и технике ВЦ ДВО РАН и успешно применяется для организации серий вычислительных экспериментов. С помощью системы пакетного запуска задач осуществлен поиск оптимальных параметров приближенного R_ν-обобщенного решения краевой задачи (1) – (4) по весовому методу конечных элементов.

Данное программное обеспечение предполагается использовать как часть системы, обеспечивающей проведение и анализ численных расчетов. При этом запуск новых серий с параметрами, сгенерированными автоматически, будет происходить уже без участия пользователя.

ЛИТЕРАТУРА

1. *Рукавишников В.А.* О весовых оценках скорости сходимости разностных схем // Докл. АН СССР. – 1986. – Т. 288, № 5. – С.1058-1062.
2. *Rukavishnikov V. A., Kuznetsova E.V.* A finite element method scheme for boundary value problems with noncoordinated degeneration of input data // Numer. Anal. Appl. – 2009. – Vol. 2, № 3. – P. 250-259.
3. *Rukavishnikov V.A.* The methods of numerical analysis for boundary value problems with strong singularity // Russ. J. of Numer. Anal. and Math. Model. – 2009. – Vol. 24, № 6. – P.565–590.
4. *Rukavishnikov V.A., Rukavishnikova H.I.* The finite element method for boundary value problem with strong singularity // J. Comp. Appl. Math. – 2010. – Vol. 234. – P. 2870-2882.
5. *Rukavishnikov V.A., Tkachenko O.P.* Numerical analysis of the mathematical model of hydroelastic oscillations in a curved pipeline // Mathematical Models and Computer Simulations. – 2011. – Vol. 3, No. 4. – P.508-516.
6. *Rukavishnikov V.A.* On differential properties R-generalized solution of the Dirichlet problem with coordinated degeneration of the input data // ISRN Mathematical Analysis. – 2011. – Vol. 2011.
7. *Rukavishnikov V.A., Mosolapov A.O.* Numerical method for solving time-harmonic Maxwell equations with strong singularity // Journal of Computational Physics. – 2012. – Vol. 231, № 6. – P.2438-2448.
8. *Рукавишников В.А., Николаев С.Г.* Метод конечных элементов для задачи теории упругости с сингулярностью // Препринт №182, ВЦ ДВО РАН. – 2012.
9. *Saad Y.* Iterative methods for sparse linear systems. – New Jersey: PWS, 1996.

Статья представлена к публикации членом редколлегии С.И. Смагиным.

E-mail:
Рукавишников Виктор Анатольевич – vark0102@mail.ru;
Николаев Сергей Георгиевич – snik-post@ya.ru;
Сарыков Александр Сергеевич – zak_990@mail.ru.