

АДАПТАЦИЯ МЕТОДИКИ РАМЗЕС-КП ДЛЯ РЕШЕНИЯ ЗАДАЧ ГАЗОВОЙ ДИНАМИКИ И ТЕПЛОПРОВОДНОСТИ НА ГИБРИДНЫХ ПАРАЛЛЕЛЬНЫХ ЭВМ

*А. Н. Быков, Д. Г. Гордеев, Н. Н. Жильникова, В. Г. Куделькин,
Е. А. Сизов, А. А. Федоров*

Российский федеральный ядерный центр –
Всероссийский НИИ экспериментальной физики, г. Саров

Введение

Появление в последние годы гибридных ЭВМ требует учета особенностей их архитектуры в прикладных программах.

В докладе обобщен опыт перевода методики РАМЗЕС-КП [1], предназначенной для расчета многокомпонентных теплопроводных сред в эйлерово-лагранжевых координатах, на ЭВМ гибридной архитектуры. Показаны особенности методики с точки зрения распараллеливания на гибридных ЭВМ, описаны проблемы, встретившиеся при адаптации программ и способы их решения.

Для достижения приемлемых показателей ускорения (от 20 до 28 раз) понадобилась переработка практически всех модулей газовой динамики и теплопроводности для работы на нескольких ускорителях, реализована связь с библиотекой УРС-ОФ [2], предназначенной для расчета уравнений состояний и пробегов фотонов в веществе, разработаны методы решения СЛАУ трехдиагонального вида для работы на нескольких ускорителях.

В докладе приводятся результаты исследования точности адаптированной программы и результаты численных экспериментов по исследованию эффективности распараллеливания.

Реализация программ на АрУ

В самом начале работы было проведено профилирование РАМЗЕС-КП, которое показало, что в коде нет «горячих пятен», т. е. нельзя получить высокое ускорение за счет переписывания только некоторых программ. Это проиллюстрировано на рис. 1. Данный рисунок оценивает сверху возможное ускорение на АрУ, которое можно было бы получить, если бы удалось часть кода распараллелить на АрУ с эффективностью 100 %. Здесь по горизонтальной оси откладывается процент идеально распараллеленного на АрУ кода (100 % – это число строк во всех подпрограммах, которые участвовали в профилировании), а по вертикальной – процент времени работы после ускорения в процентах от времени работы первоначальной версии. Идеальное распараллеливание означает, что переработанный код ускоряет время работы подпрограммы в бесконечное число раз, т. е. сводит время работы переработанных программ к 0. Например, из графика видно, что для ускорения на 70 %, необходимо перестроить около 50 % кода. При полном отсутствии «горячих пятен» подобный график выглядел бы в виде прямой, соединяющей точки (100,0) и (0,100). А при ярко выраженном «горячем пятне» этот же график имел бы более сильный «прогиб», и находился бы около осей координат. Таким образом, пришлось переписывать весь код.

Распараллеливание программ было проведено в основном поточечным методом, т. е. каждую точку обчисляет отдельная нить. Также следует отметить, что ЦП используется лишь для обмена сообщениями между процессами.

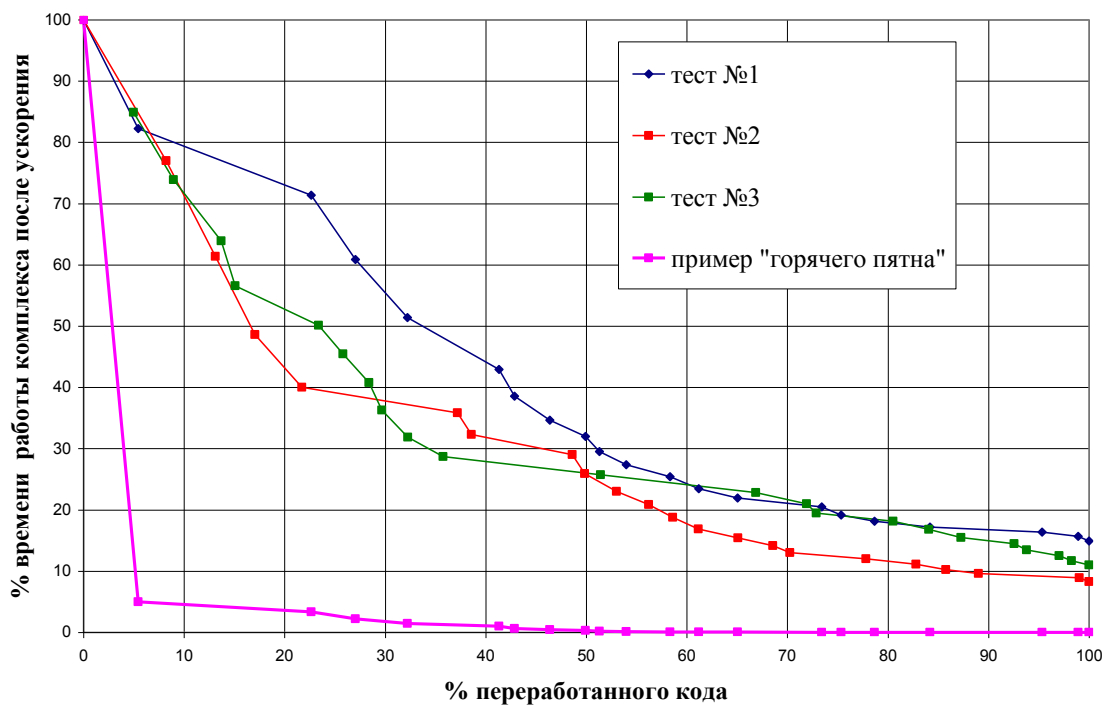


Рис. 1. Анализ трудозатрат РАЗЕС-КП на гибридных ЭВМ

В ходе адаптации программ для гибридных параллельных ЭВМ было реализовано следующее:

1. Логическое связывание ускорителя с ядром ЦП, чтобы каждый АрУ работал только с одним ядром ЦП.
2. Обмен данными между ЦП и АрУ для программ, содержащих MPI-обмены.
3. Для реализации метода прогонки, использующегося для нахождения решения систем линейных алгебраических уравнений с трехдиагональной матрицей, которая возникает из-за использования неявной схемы на нескольких АрУ, был реализован метод распараллеливания прогонки, предложенный Н. Н. Яненко [3]. Он подробно описан в [4].
4. Обработка списочных массивов, использующихся для хранения информации о смешанных ячейках (содержащих несколько веществ), была организована с помощью атомарных операций.
5. Решаемая в методике РАЗЕС-КП система уравнений газовой динамики, замыкается уравнениями, описывающими теплофизические свойства веществ: уравнениями состояния (УРС), пробегами фотонов в веществе. Доступ к программам расчета теплофизических свойств на АрУ реализован с помощью связи РАЗЕС-КП и специально разработанной версии автономной библиотеки УРС-ОФ [2].

Оптимизация реализованных программ

После первых полученных характеристик эффективности распараллеливания началась работа по оптимизации кода. Существенную часть счета на АрУ занимал обмен данными между ЦП и АрУ. Значительно сократить долю такого обмена нам позволили следующие действия:

1. Проведена оптимизация объемов копирования информации между ЦП и АрУ.
2. Использование pinned-памяти для временных массивов, которые создаются на ЦП при работе с АрУ, в два раза сокращает время их копирования. Такие массивы возникают при выполнении операций на ЦП над данными, которые были рассчитаны на АрУ (основные массивы в РАЗЕС-КП объявлены с атрибутом save, поэтому для них, к сожалению, нельзя использовать атрибут pinned).

3. Использование константной памяти для некоторых переменных, которые задаются при считывании файла расчета начальных данных задачи и не меняются в процессе расчета.

4. Передача данных по значению в аргументах функций. Это параметры, которые используются только для чтения и управления внутри ядер, их можно явно не копировать. Каждое из написанных ядер было подвергнуто анализу на предмет наличия таких параметров. Как правило, среди таких параметров встречаются номер обчитываемого фрагмента и сеточные координаты участка, который рассчитывается на текущем процессе.

5. Наиболее эффективным, но при этом и самым трудоемким процессом, стала реализация асинхронного копирования при обмене данными между ЦП и АрУ там, где это было возможно. Для этого понадобился тщательный анализ работы с памятью на шаге в методике РАМЗЕС-КП. Целью анализа являлся поиск таких мест в коде, где возможно одновременно рассчитывать какое-либо ядро и копировать необходимую информацию с ЦП на АрУ или с АрУ на ЦП. Работа в данном направлении будет продолжена.

Также в процессе оптимизации активно искались пути снижения временных затрат, уходящих на решение трехдиагональных матриц. Помимо вышеперечисленных действий для уменьшения времени работы на АрУ прогонок по первому направлению реализовано транспонирование исходной трехдиагональной матрицы и реализация метода встречных прогонок, что позволило уменьшить время работы этого участка примерно в 2 раза.

Результаты тестирования

Рассмотрим задачу обжатия легкого вещества тяжелой оболочкой, постановка которой представлена на рис. 2. С внешней стороны навязываются температура и давление.

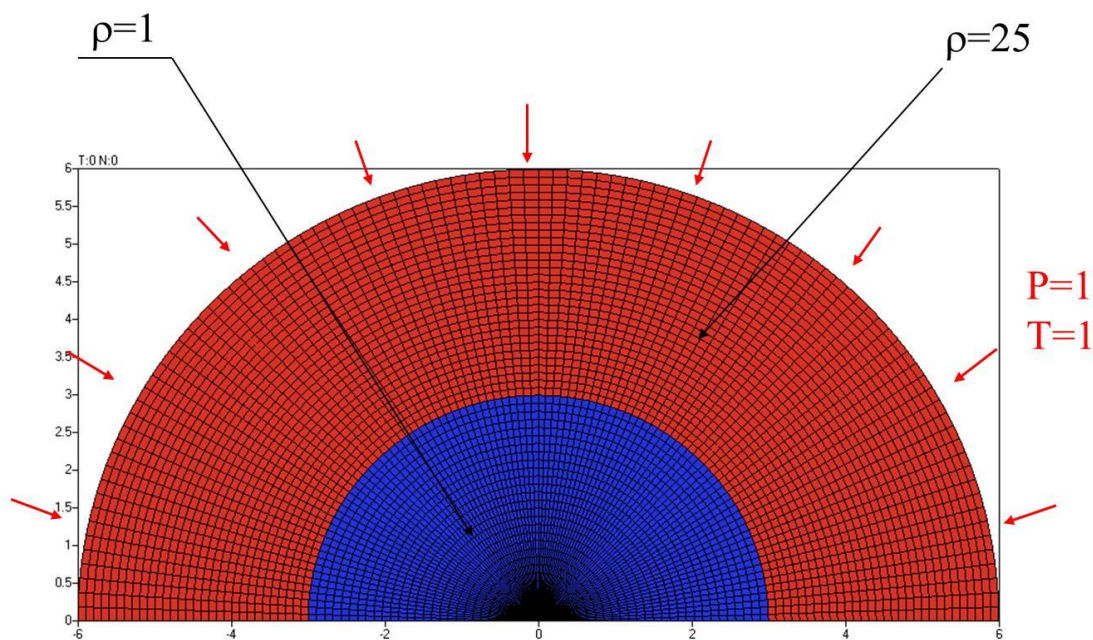


Рис. 2. Постановка тестовой задачи

Конфигурация узлов, на которых проводилось тестирование, была следующей:

- 1) ЦП содержит 12 ядер на узел;
- 2) АрУ 3 шт. на узел.

Будем считать задачу с числом точек 256×800 (200000) и числом временных шагов 1400. Задача была посчитана как на ускорителях, так и на ядрах ЦП.

График средней плотности, полученный в результате счета тестовой задачи на ЭВМ с ускорителями и без них и показывающий хорошее совпадение результатов, представлен на рис. 2.

Для измерения эффективности распараллеливания будем считать задачу методом умножения, то есть при увеличении числа процессов в 2 раза, размер задачи также увеличивается в 2 раза, причем время счета всей задачи не должно увеличиваться. Задача считалась последовательно на 1, 2, 4, 8, 16, 32 ускорителях. При счете задачи использовался 1 АрУ и 1 ядро ЦП с узла.

Ускорение на гибридной ЭВМ по сравнению с многопроцессорной ЭВМ без ускорителей представлено в табл. 1, где $t_{\text{АрУ}}$ – время работы программы на гибридной ЭВМ, $t_{\text{ЦП}}$ – время работы программы на ЭВМ без ускорителей, $t_{\text{ЦП}}/t_{\text{АрУ}}$ – ускорение на гибридной ЭВМ относительно программы на ЭВМ без ускорителей. Для наглядности те же результаты представлены на рис. 3.

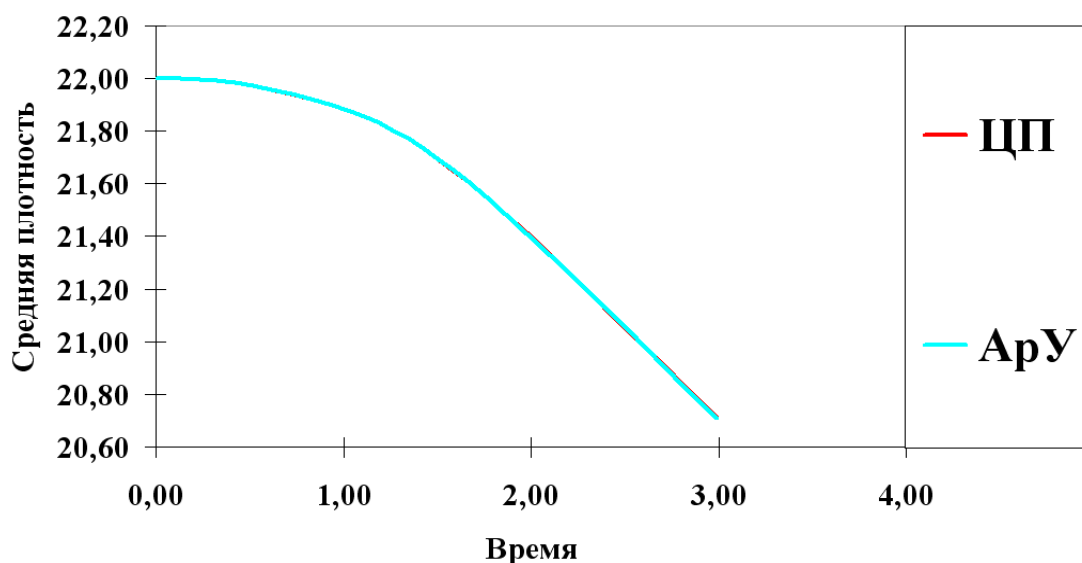


Рис. 3. График средней плотности тестовой задачи

Таблица 1

Сравнение результатов работы на ЭВМ с ускорителями и без них

| Число процессов | Размер задачи (число точек по направлениям M, K) | | $t_{\text{ЦП}}, \text{с}$ | $t_{\text{АрУ}}, \text{с}$ | $t_{\text{ЦП}}/t_{\text{АрУ}}$ |
|-----------------|---|------|---------------------------|----------------------------|--------------------------------|
| | M | K | | | |
| 1 | 800 | 256 | 4463,00 | 158,14 | 28,22 |
| 2 | 800 | 512 | 4541,13 | 192,53 | 23,59 |
| 4 | 800 | 1024 | 4597,59 | 206,46 | 22,27 |
| 8 | 800 | 2048 | 4619,47 | 211,59 | 21,83 |
| 16 | 800 | 4096 | 4652,80 | 215,00 | 21,64 |
| 32 | 800 | 8192 | 4806,30 | 222,02 | 21,65 |

Для характеристики полноты использования ресурсов использовалась эффективность распараллеливания в двух видах:

– вычисляемая по формуле $V_p = \frac{t_1}{t_p} \cdot 100$ %. Здесь t_1 обозначает время счета задачи на одном

ускорителе, а t_p – время счета задачи на p ускорителях;

– полученная с помощью набора программных средств для исследования эффективности выполнения параллельных приложений STK (Statistics Tool Kit) [5].

В табл. 2 и 3 приведены эффективности распараллеливания, полученные в расчетах с ускорителями (Эффективность на АрУ) и без них (Эффективность на ЦП).

Таблица 2

Эффективность работы на ЭВМ с ускорителями

| Число ускорителей | Размер задачи (число точек по направлениям M, K) | | Эффективность на АрУ, % | Эффективность на АрУ по STK, % |
|-------------------|---|------|-------------------------|--------------------------------|
| | M | K | | |
| 1 | 800 | 256 | 100,00 | 99,83 |
| 2 | 800 | 512 | 82,14 | 94,32 |
| 4 | 800 | 1024 | 76,59 | 89,99 |
| 8 | 800 | 2048 | 74,74 | 88,55 |
| 16 | 800 | 4096 | 73,55 | 87,94 |
| 32 | 800 | 8192 | 71,23 | 84,84 |

Таблица 3

Эффективность работы на ЭВМ без ускорителей

| Число ядер | Размер задачи (число точек по направлениям M, K) | | Эффективность на ЦП, % | Эффективность на ЦП по STK, % |
|------------|---|------|------------------------|-------------------------------|
| | M | K | | |
| 1 | 800 | 256 | 100,00 | 99,59 |
| 2 | 800 | 512 | 98,28 | 99,32 |
| 4 | 1600 | 512 | 97,07 | 98,74 |
| 8 | 1600 | 1024 | 96,61 | 98,20 |
| 16 | 3200 | 1024 | 95,92 | 97,64 |
| 32 | 3200 | 2048 | 92,86 | 95,72 |

В табл. 4 представлено время расчета тестовой задачи при использовании 3 АрУ и 3 ядер ЦП в узле в сравнении с результатами, полученными, используя 1 АрУ и 1 ядро ЦП в узле. Здесь $t_{1АрУ}$ – время работы программы при использовании 1 АрУ и 1 ядра ЦП в узле, $t_{3АрУ}$ – время работы программы при использовании 3 АрУ и 3 ядер ЦП в узле, $t_{1АрУ}/t_{3АрУ}$ – отношение времени работы с 1 АрУ в узле к времени работы программы с 3 АрУ в узле. В графическом виде это представлено на рис. 4.

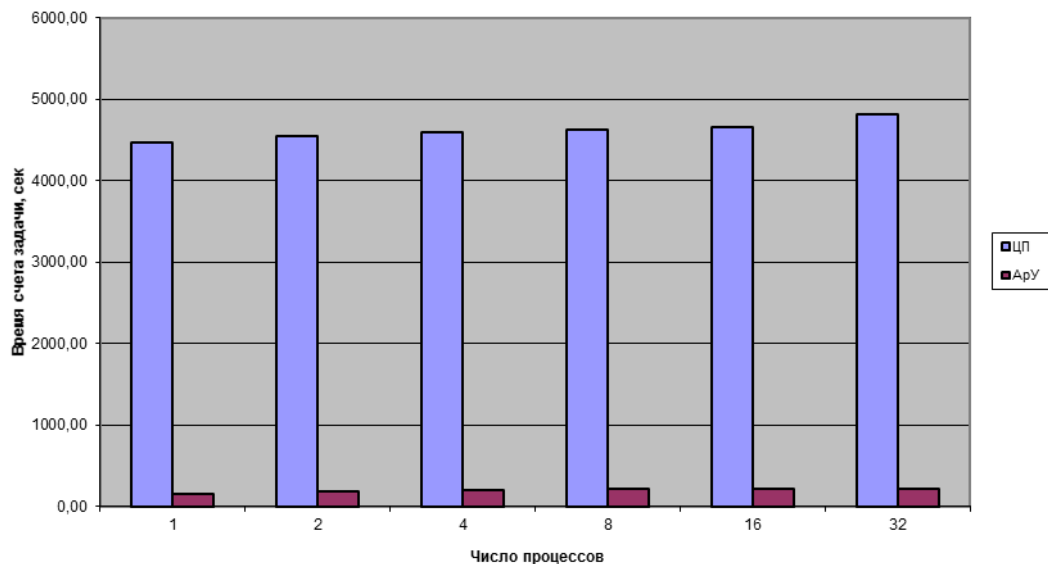


Рис. 4. Сравнение результатов работы на ЭВМ с ускорителями и без них

Таблица 4

Сравнение результатов на ЭВМ с ускорителями, используя 3 АрУ и 3 ядра ЦП в узле

| Число ускорителей | Размер задачи (число точек по направлениям M, K) | | $t_{1АрУ}, c$ | $t_{3АрУ}, c$ | $t_{1АрУ}/t_{3АрУ}$ |
|-------------------|---|------|---------------|---------------|---------------------|
| | M | K | | | |
| 1 | 800 | 256 | 158,14 | 158,14 | 1,000 |
| 2 | 800 | 512 | 192,53 | 198,04 | 0,972 |
| 4 | 1600 | 512 | 206,46 | 225,22 | 0,917 |
| 8 | 1600 | 1024 | 211,59 | 231,82 | 0,913 |
| 16 | 3200 | 1024 | 215,00 | 237,32 | 0,906 |
| 32 | 3200 | 2048 | 222,02 | 248,40 | 0,894 |

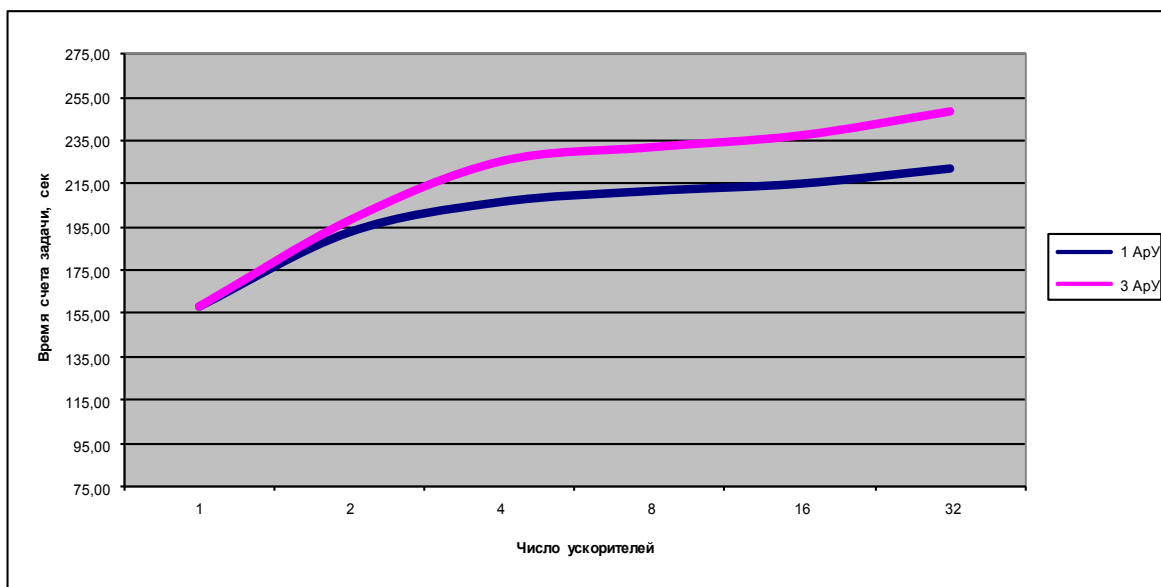


Рис. 5. Сравнение результатов на ЭВМ с ускорителями, используя 3 АрУ и 3 ядра ЦП в узле

Из представленных данных видно, что использование ускорителей позволяет уменьшить время работы программы до 28 раз и при этом время счета задачи, используя 3 АрУ в узле, отличается от времени счета, используя 1 АрУ в узле, не более чем на 11 %.

Заключение

Здесь приводится описание проделанной работы по адаптации методики РАМЗЕС-КП для счета задач газовой динамики и теплопроводности на гибридных параллельных ЭВМ с арифметическими ускорителями и оптимизации кода с целью повышения эффективности распараллеливания. Получено хорошее согласие результатов расчета тестовой задачи на гибридной параллельной ЭВМ с результатами, полученными на многопроцессорной ЭВМ (без АрУ). Эффективность распараллеливания составила не ниже 71 %. Также мы получили, что использование АрУ позволяет уменьшить время счета задачи до 28 раз.

Литература

1. Быков А. Н., Веселов В. А., Воронин Б. Л., Ерофеев А. М. Методика РАМЗЕС-КП для расчета пространственных движений многокомпонентных теплопроводных сред в эйлери-лагранжевых координатах // Труды РФЯЦ-ВНИИЭФ. 2008. Вып. 13. С. 50–57.
2. Быков А. Н., Гордеев Д. Г., Куделькин В. Г. и др. Организация взаимодействия прикладных программ и библиотеки УРС-ОФ расчета теплофизических свойств веществ на ЭВМ с арифметическими ускорителями / Под ред. Р. М. Шагалиева // Труды XIV Международ. конф. «Супервычисления и математическое моделирование». 1–5 октября 2012. Саров: РФЯЦ-ВНИИЭФ, 2013. С. 120–128.
3. Яненко Н. Н., Коновалов А. Н., Бугров А. Н., Шустов Г. В. Об организации параллельных вычислений и «распараллеливании» прогонки // Числ. методы мех. спл. среды. 1978. Т. 9, № 7. С. 139–146.
4. Быков А. Н., Ерофеев А. М., Сизов А. Н., Федоров А. А. Метод распараллеливания прогонки на гибридных ЭВМ // Вычислительные методы и программирование. 2013. Т. 14. С. 43–47.
5. Новаев Д. А., Бартнев Ю. Г., Липов Д. И. и др. Программные средства STK для исследования эффективности выполнения параллельных приложений // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 2011. Вып.4. С. 72–81.

3D-МОДЕЛИРОВАНИЕ ПОВЕДЕНИЯ ЗАЩИТНЫХ ЖЕЛЕЗОБЕТОННЫХ КОНСТРУКЦИЙ АЭС ПРИ ПАДЕНИИ САМОЛЕТА

В. И. Голяков, А. В. Наумкин

ОАО «Атомэнергопроект», г. Москва

Существующие отечественные и зарубежные нормы по проектированию АЭС предусматривают обязательный учет воздействия, вызванного падением самолета. Устойчивость функционирования при ударе самолета должна обеспечиваться для реакторного отделения, хранилищ свежего и отработанного топлива и ряда других объектов на территории станции. Необходимый уровень защиты достигается использованием монолитных железобетонных ограждающих конструкций.