

ПАРАЛЛЕЛЬНАЯ ВЕРСИЯ КОМПЛЕКСА ПРОГРАММ РАСЧЕТА ХАРАКТЕРИСТИК ЯЧЕЕК РЕАКТОРОВ CONCORD

*С. С. Касаткин, А. Н. Гребенников, С. В. Мжачих, А. В. Алексеев, Н. А. Крутько, Е. А. Гусев,
А. В. Бнятов, В. В. Евдокимов, А. Н. Москвин, М. П. Пенелеев*

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

Введение

С 2008 года в ИТМФ существенно возрос объем работ по решению задач в интересах гражданских отраслей. Одно из развивающихся направлений – создание комплексов программ для высокоточного моделирования на суперЭВМ основных теплогидравлических и нейтронно-физических процессов, протекающих в ядерных энергетических установках различного типа. К этому моменту в ИТМФ уже создан целый ряд отдельных пакетов программ, предназначенных для моделирования этих процессов. Один из них – комплекс программ SERENA (ранее часть этого комплекса носила название КОПАТ [0]). Его основное назначение – проведение на суперЭВМ связанных нейтронно-физических расчетов реакторных установок (РУ) в реальной трехмерной геометрии.

Одним из способов получения нейтронных данных, описывающих состояние ячеек активной зоны РУ, является расчет однородных малогрупповых нейтронных констант для ячейки или какой-либо ее части. Данная процедура выполняется в комплексе CONCORD.

Комплекс программ расчета нейтронно-физических характеристик ячеек реактора CONCORD

Комплекс программ CONCORD является ключевым звеном в технологии константного обеспечения 3D-динамических нейтронно-физических расчетов реакторных установок, принятой в ИТМФ РФЯЦ-ВНИИЭФ. Он предназначен для расчетов нейтронно-физических характеристик ячеек активной зоны реакторов с учетом выгорания топлива и теплового расширения тепловыделяющих элементов (ТВЭЛ). С использованием данного комплекса проводятся расчеты кампаний выгорания топливных элементов, или ТВС (тепловыделяющих сборок), а также осуществляется подготовка наборов однородных (произвольное число однородных областей в пределах ячейки) малогрупповых констант, описывающих состояния ячеек активной зоны и ее отражателей в различные моменты времени, соответствующие различным значениям глубины выгорания, для некоторого набора теплогидравлических состояний. Комплекс включает в себя:

1) блок расчета многогрупповых макроконстант для отдельных областей ячеек с учетом теплового движения ядер, химических связей атомов в веществе, резонансной экранировки сечений;

2) блок расчета нейтронно-физических характеристик ячеек в многогрупповом кинетическом приближении с учетом анизотропии рассеяния для одномерной сферической и цилиндрической геометрий;

3) блок расчета нейтронно-физических характеристик в двумерной геометрии на произвольных регулярных и нерегулярных сетках в многогрупповом анизотропном приближении с использованием Sn-метода;

4) блок расчета кинетики выгорания ядер делящихся материалов с учетом наработки выделенных осколков деления, их выгорания в нейтронных потоках и процессов радиоактивных распадов;

5) блок свертки многогрупповых нейтронных констант в малогрупповые макроконстанты с учетом нейтронных спектров в различных областях ячейки и блок гомогенизации констант в пределах определенных однородных зон ячейки (или всей ячейки).

Структура комплекса программ CONCORD для расчета характеристик ячеек тепловых РУ с учетом выгорания топлива представлена на рис. 1. Блоки комплекса, соответствующие одной сущности с точки зрения данных и назначения, выделены одинаковыми цветами.

Расчет кампании ячеек реактора и расчет малогрупповых констант для различных состояний ячейки проводится по схеме (рис. 1 Блок решения КУ). Определяется набор параметров интерполяции для данной ячейки. Определяется номинальное состояние ячейки и выбранных параметров, определяются диапазоны изменения каждого из параметров и опорные точки в этом диапазоне. Задается время окончания расчета кампании и шаги по времени. Задается удельная мощность ячейки (МВт/м^3), которая нужна для нормировки нейтронных потоков.

Сначала проводится расчет временного шага для номинального состояния ячейки. На данном шаге решается уравнение переноса нейтронов, рассчитываются критические параметры и многогрупповые нейтронные спектры, которые нормируются на заданную удельную мощность. Эти потоки передаются в блок кинетики ядер, где происходит расчет выгорания топлива ячейки. Проводится расчет глубины выгорания топлива данного шага.

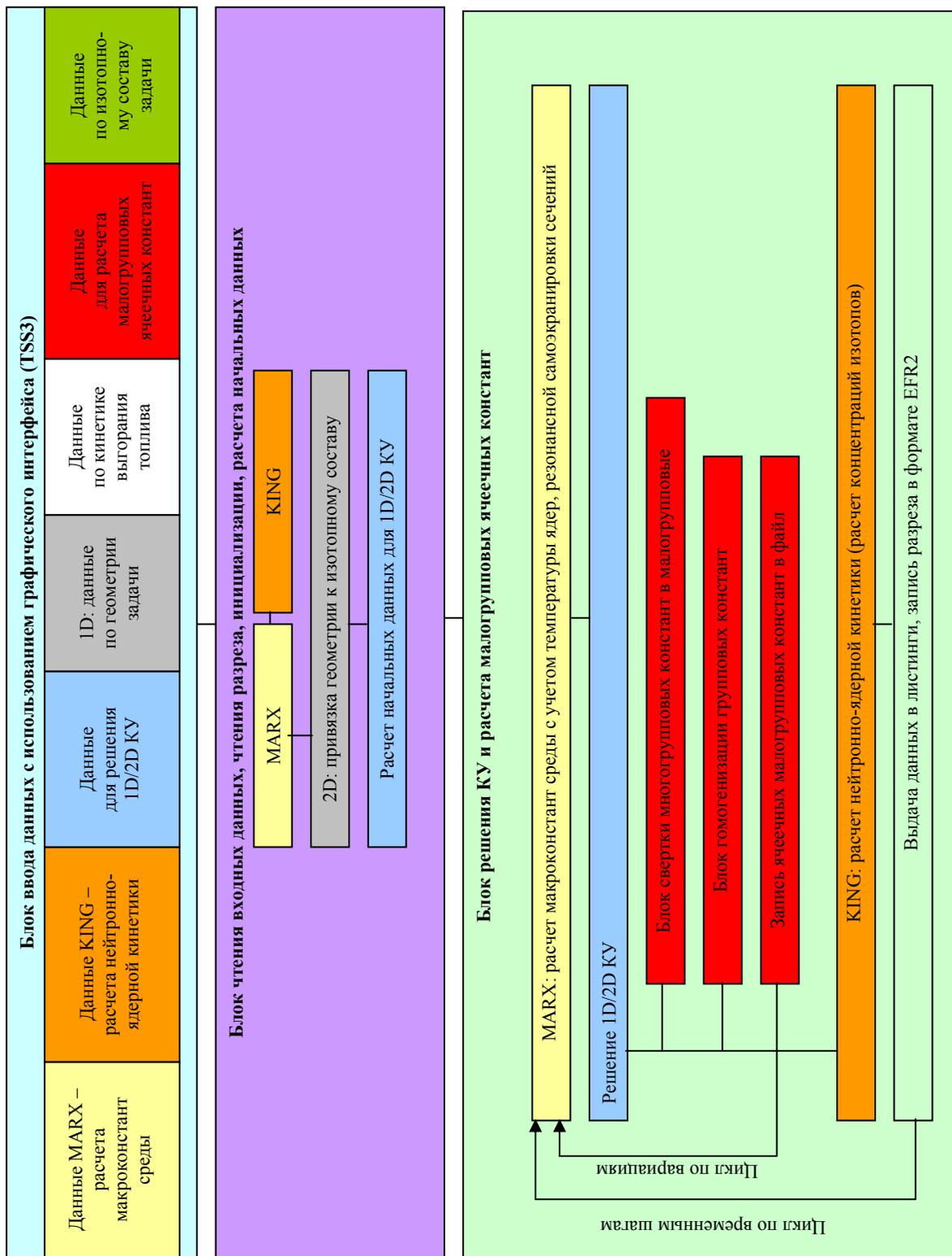


Рис. 1. Структура комплекса CONCORD

Далее происходит свертка многогрупповых констант в малогрупповые и их гомогенизация. Константы номинального состояния ячейки записываются в библиотеку.

Далее организуются циклы по всем параметрам интерполяции и по их вариациям. При этом: производится расчет уравнения переноса для данного состояния ячейки, свертка, гомогенизация и запись констант в библиотеку ячеечных констант. То есть все шаги, кроме расчета кинетики ядер и глубины выгорания топлива. Таким образом, рассчитываются гомогенные константы, отвечающие всем вариациям состояний ячейки. Далее переход к следующему шагу по времени.

Для получения малогрупповых гомогенных констант, описывающих все состояния для фиксированной глубины выгорания топлива отдельной ячейки, необходимо провести решение многогруппового уравнения переноса нейтронов для каждого из ее состояний.

Для проведения динамических 3D-расчетов реакторных установок необходимо получить библиотеку малогрупповых гомогенных констант, описывающую все состояния всех типов ячеек на все моменты времени по выгоранию. Сейчас данная процедура выполняется комплексом CONCORD на персональных компьютерах. И для нескольких десятков типов ячеек, для каждой из которых составлен список из 5–7 параметров интерполяции, имеющих по 2–3 вариации, процесс подготовки библиотеки ячеечных констант занимает несколько недель, при условии одновременного круглосуточного расчета на нескольких ПК. Это, вызвано тем, что решение многогруппового (порядка 100 групп, 10 направлений по угловой переменной и точностью сходимости итерации 1E-6) 2D-уравнения переноса нейтронов для описания конкретного состояния занимает до 2-х часов на современных персональных компьютерах.

Одним из способов ускорения этапа подготовки библиотеки ячеечных констант является применение технологий распараллеливания.

Разработка параллельной версии комплекса CONCORD

Как было сказано ранее, существует только последовательная Windows версия комплекса CONCORD. Входные данные для расчета задаются с использованием, с помощью универсальной программы ввода входных данных TSS3 [0], так же Windows зависимой. Сам процесс расчета занимает значительное время.

Таким образом, создание параллельной и кросс-платформенной версии комплекса CONCORD, для ускорения процесса получения ячеечных характеристик (ячеечных констант), представляется актуаль-

ной задачей. Так как запуски параллельной версии будут проводиться из командной строки OS Linux (посредством удаленного доступа), необходимо предварительно подготовить входные данные для старта комплекса. При этом требуется сохранить максимальную совместимость с существующей технологией ввода данных. Для этого необходимо реализовать возможность старта с разреза задачи полученного после этапа чтения входных данных и формирования ее начальных данных.

Менеджер задач параллельной версии комплекса CONCORD

Данную задачу призван решить менеджер задач комплекса CONCORD. Для его реализации выбран язык C#, платформа Windows Presentation Foundation (WPF), шаблон проектирования (паттерн) Model-View-ViewModel (MVVM) [0, 0], для связи с неуправляемым кодом (библиотекой EFR2 [**Ошибка! Источник ссылки не найден.**]) используется технология Marshaling.

Выбор обусловлен тем, что подготовка пакетов задач, состоящих из модифицированных, задач последовательной версии комплекса CONCORD, осуществляется исключительно под OS Windows. Кроме того, платформа WPF предоставляет богатую функциональность для построения графического пользовательского интерфейса, так как основана на технологии DirectX. Что снимает все ограничения на визуальное представление данных любой природы и сложности.

Шаблон проектирования MVVM был разработан специально для использования при с приложений на платформе WPF и позволяет создавать гибкие составные приложения.

Основная концепция шаблона – разделение приложения на 3 основных уровня (рис. 2): Представление – пользовательский интерфейс, отвечает только за отображение; Модель – данные, физические или иные (это могут быть файлы, информация из Баз Данных и т. д.); Модель Преставления – промежуточный слой между моделью и представлением, отвечает за подготовку данных к отображению, обработку действий пользователя и остальные операции. Благодаря такой иерархии, получается слабо связанное приложение, в котором можно без труда заменить любой из компонентов.

Как было сказано ранее, программа должна проверять список параметров интерполяции ячеечных констант, хранящийся в efr файле задачи. За работу с форматом efr отвечает C++ библиотека EFR2. Соответственно возникает необходимость обеспечить взаимодействие между C# (управляемый код) и C++ (неуправляемый код). И это возможно, благодаря

технологии Маршalling (Marshaling). Внешний вид менеджера задач показан на рис. 3.

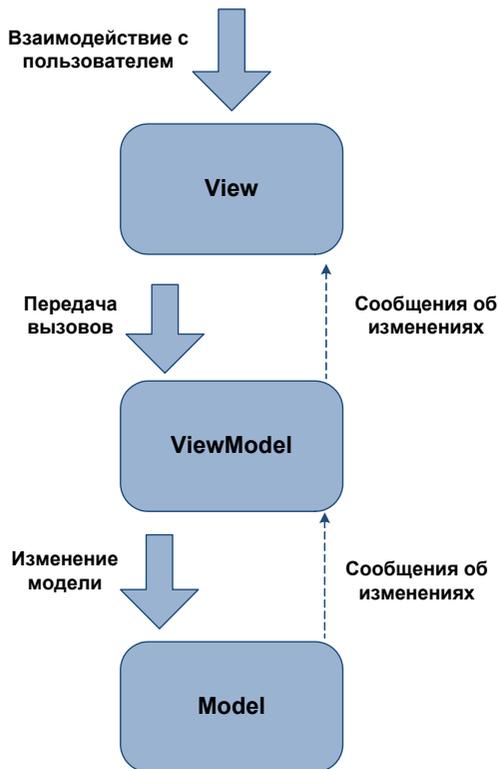


Рис. 2. MVVM

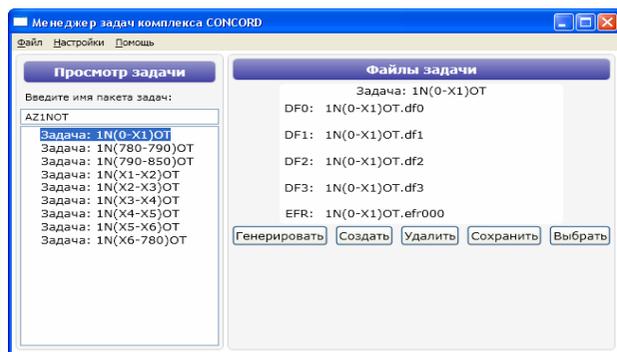


Рис. 3. Программа менеджер

Ключевым моментом является наличие в связываемой библиотеке P/Invoke функций, т.е. таких, к которым возможно сделать процедуру маршallingа. А именно, вся работа с памятью должна осуществляться с помощью CoTaskMemAlloc и CoTaskMemFree. Функции библиотеки EFR2 не обеспечивают такого механизма работы с памятью, вследствие чего возникла необходимость разработать P/Invoke функции, обеспечивающие требуемую программе функциональность. А именно: чтение и запись определенных данных efr файла. Так как все функции работают с efr файлом, то они были объединены в динамическую библиотеку.

Входными данными для программы являются файлы df наборов микроконстант задачи и efr файл-

разрез с информацией о задаче. Эти данные являются необходимыми и достаточными для ее запуска.

Рассмотрим формат файла описания пакета задач (табл. 1). Структура данного файла специально подбиралась наиболее простой, для того чтобы при необходимости можно было вручную переконфигурировать уже готовую задачу. Включение задач в существующий пакет нужно производить только в программе, потому что она автоматически производит проверку согласованности данных задач друг с другом.

Таблица 1

Формат файла описания пакета задач

Имя файла: <Имя пакета задач>.txt		
Номер строки	Тип	Описание
1	Целое	N – количество задач в пакете
2	Целое	Два числа: NTotalParam – общее число параметров; NParam – количество не нулевых (имеющих вариации)
3	Целое	NTotalParam чисел, каждое значение – число вариаций соответствующего параметра
4 – N	Строка	Имя задачи пакета задач

Стратегия работы параллельной версии комплекса CONCORD

Параллельная версия комплекса будет запускаться при помощи средств MPI. Отсюда возникает необходимость старта комплекса без графического пользовательского интерфейса, с файла разреза задачи. Для этого были внесены изменения в программный код комплекса.

Особое внимание уделить инициализации и планированию параллельных операций и обменов. Выбор стратегии распараллеливания комплекса с использованием библиотеки MPI обусловлен более широкими возможностями по сравнению с OpenMP.

Изначально планировалось распараллелить решение Кинетического уравнения (КУ) по группам с использованием средств OpenMP, так как процесс распараллеливания в этом случае сводится к правильной настройке доступа к переменным в общей памяти и расстановке соответствующих директив.

При дальнейшем анализе задачи, было решено отказаться от данного решения. Так как, возрастающее число параметров интерполяции и число их вариаций требует большего числа процессоров, поэтому использование MPI является более актуальным. Отметим, что вариации параметров интерполяции не зависят друг от друга, поэтому организовать распараллеливание можно довольно просто. Еще один плюс, в том, что на каждом временном шаге (шаге по выгоранию) для расчета вариаций нужны только

номинальные параметры, то есть необходимо выполнить всего один обмен за шаг.

Кроме того, использование MPI допускает дальнейшее использование гибридной с OpenMP методики распараллеливания. В принятой схеме распараллеливания, используется распараллеливание по вариациям параметров интерполяции.

Рассмотрим схему распараллеливания более подробно. Количество процессорных элементов, участвующих в расчете может быть любым в следующих границах: оно должно быть не меньше числа задач в пакете и не больше произведения числа параметров интерполяции на число задач в пакете.

Стратегия работы параллельной программы следующая. Сразу после входа в главную программу, выполняется инициализация параллельной программы. Каждый процесс определяет свой ранг и общее число процессов, участвующих в запуске. Глобальным главным или ведущим, считается процесс с рангом 0. Каждый процесс считывает информацию из файла конфигурации комплекса. Главный процесс выполняет чтение файла пакета задач. После чего выполняет широковещательную рассылку имени пакета задач. Далее, из исходного пула процессов, формируются группы, для работы над задачами пакета. Это осуществляется при помощи расщепления исходного коммутатора `MPI_COMM_WORLD` функцией `MPI_Split` на новые не пересекающиеся коммуникационные пространства (рис. 4).

Каждое полученное коммуникационное пространство соответствует одной задаче из пакета за-

дач. Каждый процесс определяет свой новый ранг и число процессов в новом коммутаторе, общим с ним. Главными процессами для каждого коммуникационного пространства задачи считаются те, чей новый ранг равен 0.

Далее ведущие задач запрашивают у глобально-го главного процесса задачи для расчета. После чего вся работа ведется внутри групп задач. Главный процесс задачи выполняет широковещательную рассылку имени задачи всем процессам в одной с ним группе. Выполняется создание необходимой инфраструктуре каталогов и вызов программы инициализации начальных данных.

В главной программе счета, для расчета нейтронных спектров по веществам и параметра $K_{эф}$ были организованы циклы по вариациям параметров интерполяции. За расчет одной задачи отвечает группа процессов. Их число может варьироваться в пределах от 1 до N , где N – число параметров интерполяции + 1 (для расчета номинального состояния). Поэтому возникает вопрос о распределении параметров вариаций между процессами задачи. При этом важно максимально сбалансировать расчет, т. е. максимально равномерно распределить параметры интерполяции между процессами. Стоит отметить, что головной процесс отвечает за расчет номинального состояния, поэтому должен получить минимальное число параметров для расчета. Так как от скорости его расчета и его данных, зависят все остальные процессы в группе.

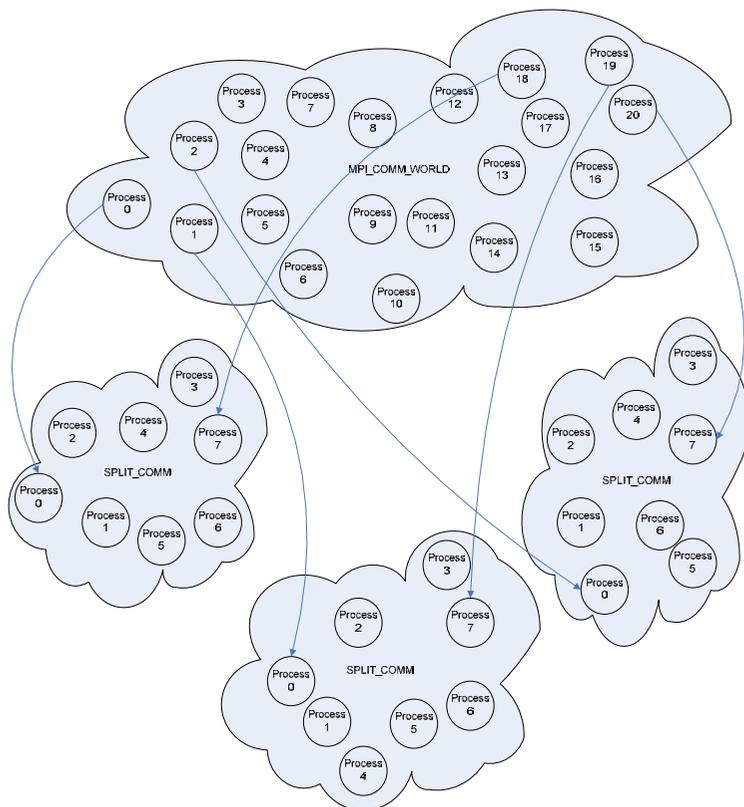


Рис. 4. Деление коммуникационного пространства

Например, пусть у нас есть задача с 6 параметрами интерполяции. В ее решении могут участвовать от 1 до $N = 6 + 1 = 7$ процессов. Соответственно, возможны следующие варианты распределений параметров, представленные в табл. 2.

Из таблицы видно, что чем меньше ранг – тем меньше вариаций он будет рассчитывать. Это распределение удовлетворяет требованию, что главный процесс, должен получить наименьшее из возможных число параметров интерполяции. Так в первом случае в расчете участвует только главный процесс, он выполняет весь расчет. Во втором – два процесса. И так далее. Наиболее интересным представляется случай для 4 процессов. Автором были предложены следующие формулы расчетов количества параметров интерполяции, для каждого процесса. Определим число параметров интерполяции, по формуле (1):

$$G_{size} = \left\lceil \frac{N_{par}}{N_{proc}} \right\rceil, \quad (1)$$

где G_{size} – будем называть grain size, минимальное количество параметров интерполяции, рассчитываемых одним процессом; N_{par} – число параметров интерполяции задачи; N_{proc} – число процессов участвующих в решении задачи; $\lceil \cdot \rceil$ – обозначает взятие целой части.

Так для 4-х процессов:

$$G_{size} = \left\lceil \frac{N_{par}}{N_{proc}} \right\rceil = \left\lceil \frac{7}{4} \right\rceil = \lceil 1,75 \rceil = 2.$$

Очевидно, что ни один процесс не может посчитать 1,75 параметра. Соответственно, применим условие минимума расчетов для ведущего процесса. И округлим 1,75 в меньшую сторону, т. е. к 1.

Осталось 6 не распределенных параметров интерполяции и 3 процесса. Для расчета количества параметров интерполяции для каждого процессора, включая нулевой, используются авторские рекуррентные формулы. Вычисления производятся каж-

дым процессором отдельно, исходя из своего ранга и числа процессоров, участвующих в решении задачи.

Формулы (2) учитывают ограничение не дублируемости параметров интерполяции (ни какие два процесса задачи не будут рассчитывать один и тот же параметр интерполяции) и принцип минимума расчетов ведущего процесса.

$$G_{size}^i = \left\lceil \frac{N_{par} - N_{par}^i}{N_{proc} - i} \right\rceil, \quad (2)$$

где N_{par}^i – число распределенных параметров интерполяции; G_{size}^i – текущий размер порции распределения; i – ранг процесса.

Заметим, что для процесса с рангом 0, формулы (2) вырождаются в (1).

$$\begin{aligned} i &= 0; \\ N_{par}^0 &= 0; \\ G_{size} &= \left\lceil \frac{N_{par}}{N_{proc}} \right\rceil. \end{aligned}$$

На основании, количества интерполяций, рассчитываемых каждым процессом, были получены величины начальных и конечных индексов для циклов главной программы счета. При этом в вырожденном случае, когда весь расчет ведется одним процессом, комплекс работает точно так же как и в последовательном режиме.

Еще одна важная модификация главной программы счета, касается, зависимостей дочерних процессов от ведущего по данным. Ведущий, рассчитывает номинальные состояния ячейки. Для продолжения счета каждый из ведомых процессов должен получить эти данные от ведущего. Поэтому, в начале каждого шага цикла по выгоранию производится широковещательная рассылка всех необходимых данных от ведущего процесса, всем процессам в группе.

Таблица 2

Распределение вариаций по процессам

Количество процессов	Число параметров интерполяции							Количество вариаций на процессе
	1	2	3	4	5	6	7	
1	■	■	■	■	■	■	■	7
2	■	■	■	■	■	■	■	3
3	■	■	■	■	■	■	■	2
4	■	■	■	■	■	■	■	1
5	■	■	■	■	■	■	■	1
6	■	■	■	■	■	■	■	1
7	■	■	■	■	■	■	■	1

В конце каждого счетного шага по выгоранию, каждый процесс, записывает рассчитанные данные ячеечных констант в выходной файл специального формата, уникальный для каждого процесса. Для получения результирующего файла, нужно произвести объединение файлов всех процессов в один.

Ранее, типичный процесс объединения состоял из следующих этапов:

- файлы ячеечных констант задачи конвертировались из двоичных файлов в текстовые с помощью программы `df_txt`;

- полученные в результате работы `df_txt` текстовые файлы объединялись `txt_txt` в один текстовый файл;

- итоговый текстовый файл с объединенными ячеечными константами конвертировался при помощи `txt_df` в двоичный файл специального формата.

Данный способ получения объединенных ячеечных констант весьма неудобен, а на больших объемах данных, просто неприемлем, так как требует множественных преобразований, а как следствие ресурсов. Кроме того, все программы являются отдельными, то есть не являются составной частью комплекса CONCORD и должны быть вызваны отдельно, после завершения работы комплекса, в строго определенном порядке.

Поэтому, для хранения и работы с ячеечными константами, были разработаны структуры данных, с использованием которых задачу создания объединенных файлов ячеечных констант задачи можно решить минуя множественные конвертации форматов, то есть напрямую. Автором был реализован полный набор подпрограмм и функций, обеспечивающих создание объединенных файлов ячеечных констант задачи.

Результаты тестирования параллельной версии комплекса CONCORD

Для тестирования программы менеджера задач и параллельной версии комплекса CONCORD, использующего подготовленные им данные, на многопроцессорной суперЭВМ была проведена серия расчетов ячеечных констант для ТВС 1-го типа.

Для этого с помощью разработанной программы менеджера был сформирован пакет задач, каждая из которых соответствовала расчету определенного слоя ТВС заданным набором параметров интерполяции со своими вариациями. Ниже представлено описание текстового файла для запуска пакета задач, полученного в программе менеджере.

```
9
4 3
2 2 2 0
1N(0-X1)OT
1N(780-790)OT
1N(790-850)OT
```

```
1N(X1-X2)OT
1N(X2-X3)OT
1N(X3-X4)OT
1N(X4-X5)OT
1N(X5-X6)OT
1N(X6-780)OT
```

Здесь 1N – идентификатор типа ТВС, значение в скобках соответствует определенному слою (по высоте) ТВС, идентификаторы OT – его заданному номинальному теплогидравлическому состоянию.

В качестве параметров интерполяции использовались следующие величины:

Параметр интерполяции	Вариации
Температура теплоносителя, К	293
	640
Плотность теплоносителя, г/см ³	0,3
	1
Температура топлива, К	293
	1200

Номинальное значение параметров интерполяции:

Параметр интерполяции	Вариации
Температура теплоносителя, К	293
Плотность теплоносителя, г/см ³	0,998
Температура топлива, К	293

Время расчета пакета задач 63 процессами (7 процессов на задачу, 1 процесс – 1 вариация) составило 1 час 23 минуты, 9-ю процессами – 8 часов 59 минут (1 процесс – 1 задача, 7 вариаций), а одним – 2 дня 17 часов 55 минут. Таким образом, ускорение:

$$S = \frac{T_{seq}}{T_{par}};$$

$$S_{1-63} = \frac{3955}{83} = 47,65;$$

$$S_{9-63} = \frac{4851}{83} = 58,44,$$

где S_{1-63} – ускорение (время расчета в минутах пакета задач в последовательном режиме к времени расчета в параллельном режиме 63-мя процессами); S_{9-63} – реальное ускорение (время расчета в минутах пакета задач 9-ю процессами к времени расчета в параллельном режиме 63-мя процессами).

А эффективность:

$$E = \left(\frac{S}{P} \right) \cdot 100\%;$$

$$E_{1-63} = \left(\frac{47,65}{63} \right) \cdot 100\% = 75,6\%;$$

$$E_{9-63} = \left(\frac{58,44}{63} \right) \cdot 100\% = 92,7\%;$$

где E_{1-63} – эффективность; E_{9-63} – реальная эффективность.

Также для иллюстрации получаемого выигрыша во времени, на РС с процессором Intel Core 2 Quad 2,4 ГГц был проведен в последовательном режиме расчет одной вариации, который занял 1 час 30 минут. Соответственно, расчет всего пакета задач в этом случае занимает: 5670 минут = 3 дня 22 часа 30 минут, что значительно больше.

Заключение

В результате работы по разработке параллельной кросс платформенной версии комплекса CONCORD были получены следующие результаты:

- создана параллельная кросс платформенная версия комплекса CONCORD для проведения массовых расчетов нейтронно-физических характеристик ячеек реакторов;

- с применением современных программных технологий разработан менеджер задач комплекса CONCORD для проведения массовых расчетов на суперЭВМ;

- с использованием разработанных программ проведены расчеты на суперЭВМ ячеечных констант одного из реальных типов ТВС модельной кассетной сборки, показавшие ускорение счета в 58 раз, а эффективность 92 %.

Применение методов распараллеливания и организации новой технологии получения библиотек ячеечных констант на многопроцессорных супер-

ЭВМ позволяют проводить массовые расчеты ячеечных констант для трехмерных динамических расчетов реакторов в кратчайшие сроки.

Литература

1. Звенигородская О. А., Шагалиев Р. М., Шемякина Т. В., Данилова Е. Н. и др. Сравнительные расчеты одной трехмерной нестационарной тестовой задачи для реакторной установки РБМК по программам KORAT-3D, DINA, STEPAN и Z3DAM (ACADEM) // ВАИТ. Сер. математическое моделирование физических процессов. 2001. Вып. 4. С. 3–10.

2. Крутько Н. А. Универсальный программный пакет TSS3 для построения пользовательского интерфейса программных комплексов на языке Фортран // ВАИТ. Сер. математическое моделирование физических процессов. В печати.

3. John Gossman «Introduction to Model/View/ViewModel pattern for building WPF apps» <http://blogs.msdn.com/johngossman/archive/2005/10/08/478683.aspx>.

4. Джош Смит (Josh Smith) «Шаблоны. Приложение WPF с шаблоном проектирования модель-представление-модель представления» // MSDN Magazine. Февраль 2009.

5. Волгин А. В., Красов А. В., Кузнецов М. Ю., Тарасов В. И. Библиотека ЕФР для универсального представления расчетных данных // Труды РФЯЦ-ВНИИЭФ. 2007. Вып. 11. С. 130–135.